

Arbitrary Lagrangian–Eulerian finite element method for lipid membranes

Amaresh Sahu^{id}

McKetta Department of Chemical Engineering, University of Texas, Austin, TX 78712, USA

Corresponding author: Amaresh Sahu, asahu@che.utexas.edu

(Received 12 February 2025; revised 25 June 2025; accepted 29 July 2025)

An arbitrary Lagrangian–Eulerian finite element method and numerical implementation for curved and deforming lipid membranes is presented here. The membrane surface is endowed with a mesh whose in-plane motion need not depend on the in-plane flow of lipids. Instead, in-plane mesh dynamics can be specified arbitrarily. A new class of mesh motions is introduced, where the mesh velocity satisfies the dynamical equations of a user-specified two-dimensional material. A Lagrange multiplier constrains the out-of-plane membrane and mesh velocities to be equal, such that the mesh and material always overlap. An associated numerical inf–sup instability ensues, and is removed by adapting established techniques in the finite element analysis of fluids. In our implementation, the aforementioned Lagrange multiplier is projected onto a discontinuous space of piecewise linear functions. The new mesh motion is compared to established Lagrangian and Eulerian formulations by investigating a pre-eminent numerical benchmark of biological significance: the pulling of a membrane tether from a flat patch and its subsequent lateral translation.

Key words: membranes, computational methods

1. Introduction

In this paper we present an arbitrary Lagrangian–Eulerian (ALE) finite element method and open-source Julia code (Sahu 2024) to simulate the dynamics of curved and deforming lipid membranes. Our developments build on ALE theories of biological membranes where the surface parametrisation is independent of the in-plane flow of lipids (Hu, Zhang & W. 2007; Torres-Sánchez *et al.* 2019; Sahu *et al.* 2020b). In our simulations, we arbitrarily specify the dynamics of the discretised surface and avoid highly distorted mesh

elements – all while not altering membrane dynamics. The utility of our implementation is demonstrated by analysing a biologically motivated scenario that is difficult to simulate with existing numerical methods: tether formation, elongation and subsequent translation.

Biological membranes are two-dimensional (2-D) materials, comprised of lipids and proteins, which make up the boundary of the cell and many of its internal organelles. The lipids and proteins flow in-plane as a 2-D fluid, while the membrane bends out-of-plane as an elastic shell. Such membranes play a fundamental role in many cellular processes, including endocytosis (Higgins & McMahon 2002; McMahon & Boucrot 2011), cell migration (Lauffenburger & Horwitz 1996) and tether network dynamics within the cell (Terasaki, Chen & Fujiwara 1986; Sciaky *et al.* 1997). Lipid membranes often undergo dramatic shape changes in which their in-plane and out-of-plane dynamics are coupled. Consequently, comprehensive models and advanced numerical techniques are needed to describe membrane behaviour. In the early 1970's, the seminal contributions of Canham (1970), Helfrich (1973) and Evans (1974) – all of which can be viewed as extensions of Naghdi's fundamental contributions to shell theory (Naghdi 1973) – laid the foundation for theoretical developments (Evans & Hochmuth 1978; Waxman 1984; Zhong-can & Helfrich 1989; Steigmann 1998, 1999; Capovilla & Guven 2002; Guven 2004; Pollard, Al-Izzi & Morris 2024) and analysis (Seifert, Berndl & Lipowsky 1991; Seifert & Langer 1993; Bar-Ziv *et al.* 1995; Fournier 1996; Goldstein *et al.* 1996; Seifert 1997; Powers, Huber & Goldstein 2002; Du, Liu & Wang 2004; Vlahovska & Gracia 2007; Agrawal & Steigmann 2009; Stone 2010; Agrawal & Steigmann 2011; Rahimi, DeSimone & Arroyo 2013; Maitra *et al.* 2014; Narsimhan, Spann & Shaqfeh 2015; Stone & Masoud 2015; Sabass & Stone 2016; Vlahovska 2016; Al-Izzi, Sens & Turner 2020; Fonda *et al.* 2020; Lin *et al.* 2021; Yu & Košmrlj 2023; Al-Izzi, Turner & Sens 2024; Dharmavaram & Hanna 2024; Faizi, Granek & Vlahovska 2024; Reboucas *et al.* 2024; Venkatesh & Narsimhan 2024; Venkatesh, Bhargava & Narsimhan 2025). However, the general, coupled nonlinear equations governing the in-plane and out-of-plane dynamics of a single-component membrane were not obtained until 2007 (Hu *et al.* 2007). These governing equations were subsequently obtained via other techniques (Arroyo & DeSimone 2009; Rangamani *et al.* 2012; Sahu, Sauer & Mandadapu 2017), and extended to describe the dynamics of multicomponent phase separation and chemical reactions with proteins in the surrounding fluid (Sahu *et al.* 2017). For our detailed perspective on the development of lipid membrane theories, see Chapter IV of Sahu (2022).

The equations governing membrane dynamics are highly nonlinear partial differential equations written on a surface that is itself arbitrarily curved and deforming over time. One cannot in general analytically solve for the time evolution of lipid flows and membrane shape changes, which are intricately coupled. However, it is also difficult to solve the full membrane equations numerically, as standard solution methods from fluid and solid mechanics struggle to capture the membrane's in-plane fluidity and out-of-plane elasticity. Many numerical studies accordingly investigated specific aspects of membrane behaviour. For example, several works captured the hydrodynamics of lipid flows on vesicles with a prescribed geometry, including the effects of the surrounding fluid as well as embedded proteins and other inclusions (Oppenheimer & Diamant 2010, 2011; Sigurdsson & Atzberger 2016; Gross & Atzberger 2018; Samanta & Oppenheimer 2021). These developments were recently extended with fluctuating hydrodynamics to incorporate phase separation and the discrete motion of proteins (Rower, Padidar & Atzberger 2022; Tran, Blanpied & Atzberger 2022). Orthogonal efforts described elastic membrane deformations either (i) in the limit of no in-plane viscosity (Feng & Klug 2006; Barrett *et al.* 2008a,b; Dziuk 2008; Ma & Klug 2008; Bonito, Nochetto & Pauletti

2010; Elliott & Stinner 2010; Mercker *et al.* 2013), possibly with the dynamics of the surrounding fluid (Narsimhan *et al.* 2015), or (ii) with the in-plane fluidity replaced by viscoelasticity (Rangarajan & Gao 2015; Zhu, Lee & Rangamani 2022). In both cases, the dynamic coupling between in-plane viscous flows and shape changes is not reflected (Sahu *et al.* 2020a). Still other works incorporated all of the aforementioned membrane complexities, but restricted their investigations to specific geometries (Rangamani *et al.* 2012, 2014; Walani, Torres & Agrawal 2015; Hassinger *et al.* 2017) – for which the membrane equations are simplified.

We reiterate that many challenges arise when developing a general numerical method that truly captures in-plane viscous lipid flows, out-of-plane membrane bending and their coupling – all on an arbitrarily curved and deforming surface. Several studies (Barrett, Garcke & Nürnberg 2015; Rodrigues *et al.* 2015; Sauer *et al.* 2017; Omar *et al.* 2020) took a Lagrangian approach, where the surface is discretised and the resulting mesh is convected with the physical, material velocity. Lagrangian implementations successfully capture membrane dynamics, but struggle to resolve in-plane flows as they lead to highly distorted elements. A remeshing procedure was used to maintain element aspect ratios, though it led to unphysical oscillations in the membrane curvature (Rodrigues *et al.* 2015). An alternative approach is for the mesh to only move normal to the surface, so that it is unaffected by in-plane flows. The mesh motion is then out-of-plane Lagrangian, as it tracks the material surface, and in-plane Eulerian. Such an approach, which we refer to as Eulerian, was implemented in Reuther, Nitschke & Voigt (2020) – though the membrane geometry was updated explicitly in a piecemeal manner. Given the importance of geometry to membrane dynamics (Sahu *et al.* 2020a), numerical methods relying on explicit mesh updates could suffer from issues known to affect explicit algorithms in the study of fluids (Zienkiewicz & Codina 1995; Zienkiewicz, Taylor & Zhu 2013). Moreover, even if a fully implicit Eulerian mesh motion was implemented as in our prior work (Sahu *et al.* 2020b), scenarios arise where the method will fail (Torres-Sánchez *et al.* 2019). We show one such example in § 4.2.

Since neither Lagrangian nor Eulerian approaches can capture commonly observed membrane behaviours, a more general mesh motion is required. The ALE theory underlying a general mesh motion was independently derived by both others (Torres-Sánchez *et al.* 2019) and ourselves (Sahu *et al.* 2020b). A new mesh motion was implemented in Torres-Sánchez *et al.* (2019), in which the mesh moved only in the direction normal to a known prior configuration of the membrane. In practice, such a choice was again not sufficiently general and required remeshing steps, which introduced errors in the numerical solution. Since the aforementioned approach resembles an Eulerian mesh motion, we hypothesise that it may not work in certain situations – such as the tether pulling scenario discussed in § 4.2. There is thus still a need for numerical implementations of more general mesh motions, which can be specified by the user when solving for membrane dynamics in a particular scenario.

The aforementioned limitations of numerical techniques motivate our development and implementation of a fully implicit ALE finite element method for lipid membranes. Rather than prescribing the mesh motion directly, we choose for the mesh velocity to satisfy a set of partial differential equations as if the mesh were itself another material. We then supply appropriate boundary conditions to the mesh velocity. In addition, the mesh and membrane are constrained to coincide via a Lagrange multiplier – which is understood as the mesh analogue of a normal force per unit area. The presence of a scalar Lagrange multiplier coupled to vector velocities leads to a numerical instability reminiscent of that identified by Ladyzhenskaya (1969), Babuška (1973) and Brezzi (1974), hereafter referred to as the LBB condition. We suppress the instability with the method of Dohrmann &

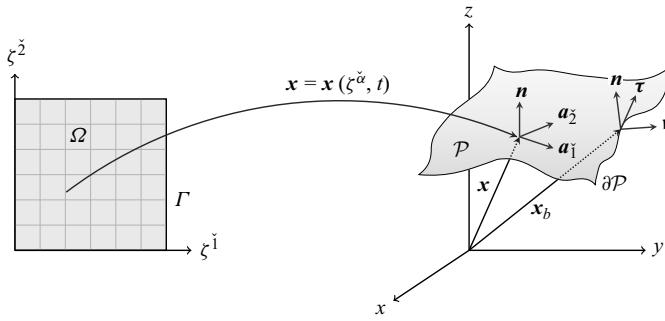


Figure 1. Surface geometry. A schematic of the mapping $\mathbf{x} = \mathbf{x}(\xi^{\checkmark}, t)$, at a single instant in time, between the parametric domain Ω and the membrane patch \mathcal{P} . The in-plane basis vectors a_{\checkmark} and unit normal vector \mathbf{n} are shown at a point \mathbf{x} on the patch, as are the in-plane unit normal \mathbf{v} and unit tangent τ at a point \mathbf{x}_b at the patch boundary $\partial\mathcal{P}$.

Bochev (2004), where the Lagrange multiplier is projected onto a space of discontinuous, piecewise linear functions.

The remainder of this paper is organised as follows. In § 2, we present the strong and weak formulations of all equations governing the membrane and mesh. A summary of the corresponding finite element formulation is provided in § 3; additional details can be found in Appendix A. All numerical results from Lagrangian, Eulerian and ALE simulations are presented in § 4. We close with conclusions and pathways for future work in § 5. Our source code is provided in the Julia package MembraneAleFem.jl (Sahu 2024).

2. The governing equations: Strong and weak formulations

The membrane is treated as a single 2-D differentiable manifold embedded in the Euclidean space \mathbb{R}^3 , for which we implicitly assume no slip between the two bilayer leaflets. In our ALE formulation the membrane surface is parametrised by coordinates that need not follow material flows, as detailed in Sahu *et al.* (2020b). Accordingly, the so-called mesh velocity \mathbf{v}^m resulting from the parametrisation will in general not equal the membrane velocity \mathbf{v} of material points. The dynamics of both the membrane and mesh are detailed in what follows; many of the results were derived in our prior work (Sahu *et al.* 2020b; Sahu 2022). The phospholipid bilayer is treated as an area-incompressible material – for which the areal density is constant and the surface tension, λ , is a Lagrange multiplier field to be solved for. Our goal is to determine the fundamental unknowns \mathbf{v} , \mathbf{v}^m and λ , as well as the membrane position \mathbf{x} , over time.

2.1. Surface parametrisation, geometry and kinematics

We begin by reviewing the framework with which lipid membranes are described. Only the most relevant features are presented here, as Sahu (2022) details our understanding of the membrane geometry and kinematics, while Sahu *et al.* (2020b) provides the ALE theory.

Consider an arbitrarily curved and deforming membrane surface \mathcal{S} , of which we examine a patch $\mathcal{P} \subset \mathcal{S}$. At any time t , the surface position \mathbf{x} is parametrised by two coordinates: ξ^1 and ξ^2 . Here, as in Sahu *et al.* (2020b), the ‘check’ accent (\checkmark) indicates the parametrisation can be arbitrarily specified. From now on, Greek indices span the set $\{1, 2\}$, such that $\mathbf{x}(\xi^{\checkmark}, t) \in \mathbb{R}^3$ denotes the position of a point on the membrane surface. As shown in figure 1, there is a mapping from the parametric domain Ω in the ξ^1 - ξ^2 plane

to the membrane patch \mathcal{P} . Area integrals over the patch are thus evaluated as

$$\int_{\mathcal{P}} (\dots) \, da = \int_{\Omega} (\dots) J_{\Omega} \, d\Omega, \quad (2.1)$$

where J_{Ω} is the Jacobian of the mapping. Here J_{Ω} has dimensions of area and its functional form is provided below. At any point $\mathbf{x}(\zeta^{\check{\alpha}}, t)$, the vectors $\mathbf{a}_{\check{\alpha}} := \partial \mathbf{x} / \partial \zeta^{\check{\alpha}}$ form a basis of the tangent plane to the surface, which has unit normal $\mathbf{n} = (\mathbf{a}_{\check{1}} \times \mathbf{a}_{\check{2}}) / |\mathbf{a}_{\check{1}} \times \mathbf{a}_{\check{2}}|$. Covariant components of the metric and curvature tensors are respectively $a_{\check{\alpha}\check{\beta}} := \mathbf{a}_{\check{\alpha}} \cdot \mathbf{a}_{\check{\beta}}$ and $b_{\check{\alpha}\check{\beta}} := \mathbf{n} \cdot \mathbf{x}_{;\check{\alpha}\check{\beta}} = \mathbf{n} \cdot \mathbf{x}_{;\check{\alpha}\check{\beta}}$. Here $(\cdot)_{,\check{\alpha}} := \partial(\cdot) / \partial \zeta^{\check{\alpha}}$ and $(\cdot)_{;\check{\alpha}}$ are respectively the partial and covariant derivatives with respect to $\zeta^{\check{\alpha}}$. The Jacobian J_{Ω} appearing in (2.1) is expressed as $J_{\Omega} = \sqrt{\det a_{\check{\alpha}\check{\beta}}}$. The mean curvature H is calculated as $H = (1/2) a^{\check{\alpha}\check{\beta}} b_{\check{\alpha}\check{\beta}}$, where $a^{\check{\alpha}\check{\beta}}$ is the contravariant metric and is calculated as the matrix inverse of $a_{\check{\alpha}\check{\beta}}$. In the present work, the Einstein summation convention is employed, in which repeated raised and lowered indices are summed over. The Gaussian curvature K is computed as $K = \det(b_{\check{\alpha}\check{\beta}}) / \det(a_{\check{\alpha}\check{\beta}})$. Finally, at points \mathbf{x}_b on the patch boundary $\partial\mathcal{P}$, additional in-plane basis vectors are defined: $\mathbf{v} = v^{\check{\alpha}} \mathbf{a}_{\check{\alpha}} = v_{\check{\alpha}} \mathbf{a}^{\check{\alpha}}$ is the in-plane unit normal and $\mathbf{r} = \tau^{\check{\alpha}} \mathbf{a}_{\check{\alpha}} = \tau_{\check{\alpha}} \mathbf{a}^{\check{\alpha}}$ is the in-plane unit tangent (see figure 1). Line integrals over the patch boundary are calculated as

$$\int_{\partial\mathcal{P}} (\dots) \, ds = \int_{\Gamma} (\dots) J_{\Gamma} \, d\Gamma, \quad (2.2)$$

where s is the arc length parametrisation of $\partial\mathcal{P}$ and $\Gamma := \partial\Omega$ is the boundary of the parametric domain. In (2.2), $J_{\Gamma} = [(\tau^1)^2 + (\tau^2)^2]^{-1/2}$ is the Jacobian of the mapping from Γ to $\partial\mathcal{P}$.

When solving for the membrane state at a particular time, we seek the surface tension, material velocity and mesh velocity fields on the surface. To this end, it is useful to define relevant function spaces over the parametric domain Ω . The space of square-integrable functions, $L^2(\Omega)$, is expressed as

$$L^2(\Omega) := \left\{ u(\zeta^{\check{\alpha}}) : \Omega \rightarrow \mathbb{R} \text{ such that } \left(\int_{\Omega} u^2 \, d\Omega \right)^{1/2} < \infty \right\}. \quad (2.3)$$

The Sobolev space of order k , $H^k(\Omega)$, consists of functions that are square-integrable and also have up to k partial derivatives that are square-integrable:

$$H^k(\Omega) := \left\{ u(\zeta^{\check{\alpha}}) : \Omega \rightarrow \mathbb{R} \text{ such that } \underbrace{u_{,\check{\alpha}\check{\beta}\dots\check{\gamma}}}_{m \text{ derivatives}} \in L^2(\Omega) \text{ for } 0 \leq m \leq k \right\}. \quad (2.4)$$

From the definitions in (2.3) and (2.4), it is evident that $H^0(\Omega) = L^2(\Omega)$. Moreover, since the material and mesh velocities are elements of \mathbb{R}^3 , we also define

$$\mathbf{H}^k(\Omega) := \left\{ \mathbf{u}(\zeta^{\check{\alpha}}) : \Omega \rightarrow \mathbb{R}^3 \text{ such that } u_j \in H^k(\Omega) \text{ for } j = 1, 2, 3 \right\} \quad (2.5)$$

as the space of functions in which each Cartesian component is an element of $H^k(\Omega)$. In (2.5) we denote $u_j := \mathbf{u} \cdot \mathbf{e}_j$, where $\{\mathbf{e}_j\}_{j=1,2,3}$ is the canonical Cartesian basis in \mathbb{R}^3 .

We close with a discussion of membrane kinematics. The membrane velocity $\mathbf{v} = d\mathbf{x}/dt = \dot{\mathbf{x}}$ is the material time derivative of the position, and is expanded in the $\{\mathbf{a}_{\check{\alpha}}, \mathbf{n}\}$ basis as $\mathbf{v} = v^{\check{\alpha}} \mathbf{a}_{\check{\alpha}} + v \mathbf{n}$. The mesh velocity, which is treated as a fundamental unknown independent of the material velocity, is defined to be the rate of change of position when

$\zeta^{\check{\alpha}}$ is held constant – expressed as $\mathbf{v}^m = (\partial \check{\mathbf{x}} / \partial t)|_{\zeta^{\check{\alpha}}}$. In order for the material and mesh velocities to correspond to the same surface, kinematics require

$$\mathbf{v}^m \cdot \mathbf{n} = \mathbf{v} \cdot \mathbf{n} \quad (2.6)$$

such that the mesh motion is always out-of-plane Lagrangian (Sahu *et al.* 2020*b*). Importantly, the in-plane material velocity components $v^{\check{\alpha}}$ and mesh velocity components $v_m^{\check{\alpha}} = \mathbf{v}^m \cdot \mathbf{a}^{\check{\alpha}}$ need not coincide, such that within our ALE framework one can dictate how the mesh evolves within the membrane surface. In this work, (2.6) is referred to as the ALE kinematic constraint.

2.2. The balance of mass: Material incompressibility

As lipid membranes only stretch 2 %–3 % before tearing (Evans & Skalak 1980; Nichol & Hutter 1996), we treat them as area-incompressible 2-D materials. The local form of the balance of mass, also referred to as the continuity equation and incompressibility constraint, is given by

$$\mathbf{a}^{\check{\alpha}}(\zeta^{\check{\gamma}}, t) \cdot \mathbf{v}_{,\check{\alpha}}(\zeta^{\check{\gamma}}, t) = 0 \quad \text{for all } \zeta^{\check{\gamma}} \in \Omega, \quad t \in [0, t_f], \quad (2.7)$$

where t_f is the end of the time interval under consideration. For notational simplicity, the functional dependence of all quantities will be suppressed, as will the domains of $\zeta^{\check{\gamma}}$ and t . The incompressibility constraint $\mathbf{a}^{\check{\alpha}} \cdot \mathbf{v}_{,\check{\alpha}} = 0$ is equivalently expressed as $v_{;\check{\alpha}}^{\check{\alpha}} - 2vH = 0$, and is enforced by the Lagrange multiplier field $\lambda = \lambda(\zeta^{\check{\alpha}}, t)$ – which has dimension of energy/area and acts as the membrane surface tension (Sahu 2022, Chapter V, § 6(a)–(c)).

The weak formulation of (2.7) is obtained by multiplying it with an arbitrary surface tension variation $\delta\lambda$ and integrating over the membrane area. The tension variation is assumed to be square-integrable, for which the weak form of the continuity equation is expressed as

$$\int_{\Omega} \delta\lambda (\mathbf{a}^{\check{\alpha}} \cdot \mathbf{v}_{,\check{\alpha}}) J_{\Omega} d\Omega = 0 \quad \text{for all } \delta\lambda \in L^2(\Omega). \quad (2.8)$$

Care must be taken when discretising (2.8) in the course of finite element analysis, as one could violate the LBB condition (Ladyzhenskaya 1969; Babuška 1973; Brezzi 1974) and observe spurious surface tension oscillations in the numerical solution. A variety of techniques were developed to prevent such oscillations and numerically stabilise the system (Malkus & Hughes 1978; Brezzi & Pitkäranta 1984; Zienkiewicz & Nakazawa 1984; Dörfler 1996; Dörfler & Bochev 2004). We choose to employ the method of Dohrmann & Bochev (2004), as it is based on an underlying theory and is straightforward to implement numerically (Zienkiewicz *et al.* 2013).

2.2.1. The Dohrmann–Bochev method: Numerical stabilisation

The Dohrmann–Bochev method prevents spurious surface tension oscillations by projecting the surface tension onto a space of discontinuous, piecewise linear functions. The function space, denoted \check{L} , is defined in (3.7) following a discussion of the surface discretisation. The L^2 projection of a given surface tension field $\lambda \in L^2(\Omega)$ onto \check{L} is denoted $\check{\lambda}$ and is defined through the relation (Dohrmann & Bochev 2004)

$$\int_{\Omega} \delta\check{\lambda} (\lambda - \check{\lambda}) d\Omega = 0 \quad \text{for all } \delta\check{\lambda} \in \check{L}. \quad (2.9)$$

In practice, the method of Dohrmann & Bochev (2004) is implemented by quadratically penalising surface tension deviations from the space \check{L} , for which the quantity

$$\frac{\alpha^{DB}}{\zeta} \int_{\Omega} (\delta\lambda - \delta\check{\lambda})(\lambda - \check{\lambda}) d\Omega \quad (2.10)$$

is subtracted from (2.8). Here, ζ is the 2-D intramembrane viscosity: a material property with dimensions of energy \times time/area. In addition, α^{DB} is a user-chosen parameter with dimensions of area, such that (2.8) and (2.10) have the same dimensions – though the value of α^{DB} is not observed to affect simulation results. The numerically stabilised weak formulation of the incompressibility constraint is written as

$$\mathcal{G}_\lambda = 0 \quad \text{for all } \delta\lambda \in L^2(\Omega), \quad (2.11)$$

where \mathcal{G}_λ is the surface tension contribution to the direct Galerkin expression (Zienkiewicz & Taylor 2014) given by

$$\mathcal{G}_\lambda := \int_{\Omega} \delta\lambda (\mathbf{a}^{\check{\alpha}} \cdot \mathbf{v}_{,\check{\alpha}}) J_{\Omega} d\Omega - \frac{\alpha^{DB}}{\zeta} \int_{\Omega} (\delta\lambda - \delta\check{\lambda})(\lambda - \check{\lambda}) d\Omega. \quad (2.12)$$

In (2.12), $\check{\lambda}$ and $\delta\check{\lambda}$ are understood to be the L^2 projections of their respective counterparts λ and $\delta\lambda$ onto \check{L} , according to (2.9).

2.3. The balance of linear momentum: Membrane dynamics

Consider a general, arbitrarily curved and deforming 2-D material for which inertial effects are negligible. The local form of the balance of linear momentum for such a material is given by

$$\mathbf{T}_{;\check{\alpha}}^{\check{\alpha}} + \mathbf{f} = \mathbf{0}, \quad (2.13)$$

where $\mathbf{T}^{\check{\alpha}}$ is the internal traction along a curve of constant $\zeta^{\check{\alpha}}$ on the surface and \mathbf{f} is the net body force per unit area on the material by its surroundings. The balance of angular momentum additionally requires that the stress vectors be expressed as (Sahu 2022)

$$\mathbf{T}^{\check{\alpha}} = \sigma^{\check{\alpha}\check{\beta}} \mathbf{a}_{\check{\beta}} - (M^{\check{\beta}\check{\alpha}} \mathbf{n})_{;\check{\beta}}. \quad (2.14)$$

Here, $\sigma^{\check{\alpha}\check{\beta}}$ contains the couple-free, in-plane stress components and $M^{\check{\alpha}\check{\beta}}$ contains the couple-stress components. If the constitutive form of $\sigma^{\check{\alpha}\check{\beta}}$ and $M^{\check{\alpha}\check{\beta}}$ are known, then (2.13) and (2.14) govern the dynamics of a general 2-D material.

In lipid bilayers, the constitutive relations for $\sigma^{\check{\alpha}\check{\beta}}$ and $M^{\check{\alpha}\check{\beta}}$ are well known, and depend on three material parameters. The first is the intramembrane viscosity ζ , which characterises the irreversibility of in-plane flows. The other parameters are the mean and Gaussian bending moduli, denoted by k_b and k_g , which have dimensions of energy and respectively penalise non-zero mean and Gaussian curvatures (Canham 1970; Helfrich 1973; Evans 1974). We previously determined the membrane stresses and couple stresses within the framework of irreversible thermodynamics, and found that (Sahu 2022)

$$M^{\check{\alpha}\check{\beta}} = k_b H a^{\check{\alpha}\check{\beta}} + k_g (2 H a^{\check{\alpha}\check{\beta}} - b^{\check{\alpha}\check{\beta}}) \quad (2.15)$$

and

$$\sigma^{\check{\alpha}\check{\beta}} = k_b (H^2 a^{\check{\alpha}\check{\beta}} - 2 H b^{\check{\alpha}\check{\beta}}) - k_g K a^{\check{\alpha}\check{\beta}} + \lambda a^{\check{\alpha}\check{\beta}} + \pi^{\check{\alpha}\check{\beta}}, \quad (2.16)$$

where

$$\pi^{\check{\alpha}\check{\beta}} = \zeta \mathbf{v}_{,\check{\mu}} \cdot (\mathbf{a}^{\check{\alpha}} a^{\check{\mu}\check{\beta}} + \mathbf{a}^{\check{\beta}} a^{\check{\mu}\check{\alpha}}) \quad (2.17)$$

are the in-plane viscous stress components. The couple stresses (2.15) involve only k_b and k_g , and thus are purely elastic, while $\sigma^{\check{\alpha}\check{\beta}}$ contains bending, tensile and viscous contributions (2.16).

The in-plane and out-of-plane equations governing lipid membrane dynamics are obtained by substituting (2.14)–(2.17) into (2.13) and contracting the result with $\mathbf{a}_{\check{\alpha}}$ and \mathbf{n} – which yields (Sahu 2022)

$$\zeta \left(\Delta_s v_{\check{\alpha}} + K v_{\check{\alpha}} + 2v_{,\check{\alpha}} H - 2v_{,\check{\beta}} b_{\check{\alpha}}^{\check{\beta}} - 2v H_{,\check{\alpha}} \right) + f_{\check{\alpha}} + \lambda_{,\check{\alpha}} = 0 \quad (2.18)$$

and

$$f + 2\lambda H + \zeta \left(2b^{\check{\alpha}\check{\beta}} v_{\check{\alpha};\check{\beta}} - 8v H^2 + 4v K \right) - k_b (2H^3 - 2HK + \Delta_s H) = 0. \quad (2.19)$$

Equations (2.18) and (2.19) are respectively referred to as the in-plane and shape equations, and were independently obtained via different approaches (Hu *et al.* 2007; Arroyo & DeSimone 2009; Rangamani *et al.* 2012; Sahu *et al.* 2017). Here, the body force per unit area f is decomposed as $f = \mathbf{a}^{\check{\alpha}} f_{\check{\alpha}} + f \mathbf{n}$, and the operator Δ_s acts on an arbitrary quantity (\cdot) as $\Delta_s(\cdot) := a^{\check{\mu}\check{\nu}}(\cdot)_{;\check{\mu}\check{\nu}}$. Note that while the membrane bends elastically, the in-plane viscosity ζ enters the shape equation (2.19) due to a coupling between in-plane stresses and curvature (Sahu *et al.* 2020a). The surface tension λ and surface curvatures H and K also enter both the in-plane and shape equations, leading to non-trivial couplings between in-plane and out-of-plane dynamics.

2.3.1. The boundary conditions

One cannot determine a well-posed set of boundary conditions to (2.18) and (2.19) by inspection. A series of systematic developments for elastic shells (Green & Naghdi 1968; Steigmann 1998, 1999) underlie the formulation of the general lipid membrane boundary conditions (Capovilla, Guven & Santiago 2002; Arroyo & DeSimone 2009; Rangamani *et al.* 2012; Sahu *et al.* 2017; Sauer & Duong 2017). In what follows, we highlight possible boundary conditions and provide their physical justification. Details of our own derivations, which reproduce earlier results, are provided in Chapter V, § 5(d) of Sahu (2022).

The in-plane equations governing lipid flows (2.18) are identical to those governing a 2-D Newtonian fluid (Scriven 1960). We thus expect the boundary conditions to be similar to those of a fluid, in which one specifies either the velocity \mathbf{v} or (for a surface) the force per length \mathbf{F} on the boundary. For general 2-D materials, the force per length on the patch boundary is given by

$$\mathbf{F} = \mathbf{T}^{\check{\alpha}} v_{\check{\alpha}} - \left(M^{\check{\alpha}\check{\beta}} v_{\check{\alpha}} \tau_{\check{\beta}} \mathbf{n} \right)_{,\check{\mu}} \tau^{\check{\mu}}, \quad (2.20)$$

which – for the case of lipid membranes – has bending, tensile and viscous contributions (recall \mathbf{v} and $\boldsymbol{\tau}$ are boundary basis vectors; see figure 1). In our numerical implementation, on each edge of the membrane patch we specify a component of either the velocity \mathbf{v} or the force per length \mathbf{F} in each of the three Cartesian directions. We denote Γ_F^j and Γ_v^j as the respective portions of the boundary where $F_j := \mathbf{F} \cdot \mathbf{e}_j$ and $v_j := \mathbf{v} \cdot \mathbf{e}_j$ are specified. For $j \in \{1, 2, 3\}$, the intersection $\Gamma_v^j \cap \Gamma_F^j = \emptyset$ and the closure of the union $\overline{\Gamma}_v^j \cup \overline{\Gamma}_F^j = \Gamma$. The boundary conditions are expressed as

$$v_j = \bar{v}_j \quad \text{on } \Gamma_v^j \quad \text{and} \quad F_j = \bar{F}_j \quad \text{on } \Gamma_F^j, \quad (2.21)$$

where \bar{v}_j and \bar{F}_j are known quantities that we prescribe.

The boundary conditions in (2.21) are necessary but not sufficient for a mathematically well-posed scenario. To see why, note that the membrane bending energy gives rise to the $\Delta_s H = a^{\check{\mu}\check{\nu}} H_{;\check{\mu}\check{\nu}}$ term in the shape equation (2.19). Since the mean curvature contains two spatial derivatives of the surface position through the curvature components $b_{\check{\alpha}\check{\beta}} = \mathbf{n} \cdot \mathbf{x}_{,\check{\alpha}\check{\beta}}$, the $\Delta_s H$ term contains four derivatives of the surface position. Following canonical developments in the theory of beam bending (Timoshenko 1921, 1922), we expect to specify two conditions along the entire boundary: either the out-of-plane velocity or force per length, as well as either the slope of the surface or the boundary moment

$$M := M^{\check{\alpha}\check{\beta}} v_{\check{\alpha}} v_{\check{\beta}}. \quad (2.22)$$

Since the velocity and force boundary conditions are already contained in (2.21), we need only specify one of the latter pair. More precisely, we partition the boundary into the disjoint sets Γ_n and Γ_M , with $\Gamma_n \cap \Gamma_M = \emptyset$ and $\Gamma_n \cup \Gamma_M = \Gamma$, and prescribe

$$\mathbf{n} \cdot \mathbf{v}_{,\check{\alpha}} v^{\check{\alpha}} = \bar{v}_v \quad \text{on } \Gamma_n \quad \text{and} \quad M = \bar{M} \quad \text{on } \Gamma_M, \quad (2.23)$$

where \bar{v}_v and \bar{M} are prescribed quantities. With (2.20)–(2.23), we have a mathematically well-posed set of boundary conditions for the governing equations.

2.3.2. The weak formulation

The weak formulation of the balance of linear momentum is obtained by first contracting (2.13) with an arbitrary velocity variation $\delta\mathbf{v}$. At this point, we recognise the four spatial derivatives contained in the shape equation (2.19) through the $\Delta_s H$ term will, in the subsequent weak form, yield two spatial derivatives of both the velocity variation and the surface position. Since \mathbf{v} and \mathbf{x} are assumed to lie in the same space of functions, both are elements of $\mathbf{H}^2(\Omega)$ – for which higher-order basis functions, with continuous first derivatives, are required in the numerical implementation. In accordance with the boundary conditions in (2.21) and (2.23), the space of admissible material velocity variations \mathcal{V}_0 is expressed as

$$\mathcal{V}_0 := \left\{ \mathbf{u}(\xi^{\check{\alpha}}) : \Omega \rightarrow \mathbb{R}^3 \text{ such that } \mathbf{u} \in \mathbf{H}^2(\Omega), u_j|_{\Gamma_v^j} = 0, (\mathbf{n} \cdot \mathbf{u}_{,\check{\alpha}} v^{\check{\alpha}})|_{\Gamma_n} = 0 \right\} \quad (2.24)$$

for $j \in \{1, 2, 3\}$. We integrate the result of the contraction over the membrane area, and substitute the general form of the stress vectors (2.14) to obtain

$$\int_{\Omega} \delta\mathbf{v} \cdot \left[\sigma^{\check{\alpha}\check{\beta}} \mathbf{a}_{\check{\beta}} - (M^{\check{\beta}\check{\alpha}} \mathbf{n})_{;\check{\beta}} \right]_{;\alpha} J_{\Omega} d\Omega + \int_{\Omega} \delta\mathbf{v} \cdot \mathbf{f} J_{\Omega} d\Omega = 0 \quad \text{for all } \delta\mathbf{v} \in \mathcal{V}_0. \quad (2.25)$$

Starting with (2.25), a series of algebraic manipulations are required to determine the weak form of the linear momentum balance. The calculations rely on developments by Green & Naghdi (1968) and Steigmann (1998), and can be found in Sauer *et al.* (2017), Sauer & Duong (2017). Following the notation and development of our prior efforts (Sahu 2022, Chapter V, § 5(d)), the weak formulation of the balance of linear momentum is expressed as

$$\mathcal{G}_v = 0 \quad \text{for all } \delta\mathbf{v} \in \mathcal{V}_0, \quad (2.26)$$

where

$$\begin{aligned} \mathcal{G}_v := & \int_{\Omega} \frac{1}{2} \left(\delta \mathbf{v}_{,\check{\alpha}} \cdot \mathbf{a}_{\check{\beta}} + \delta \mathbf{v}_{,\check{\beta}} \cdot \mathbf{a}_{\check{\alpha}} \right) \sigma^{\check{\alpha}\check{\beta}} J_{\Omega} d\Omega + \int_{\Omega} \frac{1}{2} \left(\delta \mathbf{v}_{;\check{\alpha}\check{\beta}} + \delta \mathbf{v}_{;\check{\beta}\check{\alpha}} \right) \cdot \mathbf{n} M^{\check{\alpha}\check{\beta}} J_{\Omega} d\Omega \\ & - \int_{\Omega} \delta \mathbf{v} \cdot \mathbf{f} J_{\Omega} d\Omega - \sum_{j=1}^3 \int_{\Gamma_F^j} \delta v_j \bar{F}_j J_{\Gamma} d\Gamma - \int_{\Gamma_M} \delta \mathbf{v}_{,\check{\alpha}} v^{\check{\alpha}} \cdot \mathbf{n} \bar{M} J_{\Gamma} d\Gamma. \end{aligned} \quad (2.27)$$

Note that jump forces can arise at corners of the patch and enter (2.27). They are assumed here to be zero. In (2.27), $\delta v_j := \delta \mathbf{v} \cdot \mathbf{e}_j$ is the j th Cartesian component of the arbitrary velocity variation. It is useful to recognise that (2.27) is general to any material whose stress vectors can be expressed as in (2.14), with the corresponding boundary conditions in (2.21) and (2.23).

2.4. The dynamics of the mesh

In our ALE formulation, the material velocity \mathbf{v} and the mesh velocity \mathbf{v}^m are independent quantities. The evolution of the position $\mathbf{x}(\zeta^{\check{\alpha}}, t)$ of the membrane surface is dictated by the mesh velocity via the relations

$$\mathbf{v}^m(\zeta^{\check{\alpha}}, t) = \frac{\partial}{\partial t}(\mathbf{x}(\zeta^{\check{\alpha}}, t)) \quad (2.28)$$

and

$$\mathbf{x}(\zeta^{\check{\alpha}}, t) = \mathbf{x}(\zeta^{\check{\alpha}}, 0) + \int_0^t \mathbf{v}^m(\zeta^{\check{\alpha}}, t') dt', \quad (2.29)$$

which both apply over all parametric coordinates $\zeta^{\check{\alpha}} \in \Omega$ and for all times $t \in [0, t_f]$ under consideration (see Sahu *et al.* (2020b, § 2) for additional details). As we previously recognised $\mathbf{x} \in \mathbf{H}^2(\Omega)$, (2.28) and (2.29) reveal $\mathbf{v}^m \in \mathbf{H}^2(\Omega)$ as well. In what follows, we discuss three mesh velocity schemes of increasing complexity, all of which are consistent with the kinematic constraint (2.6) such that the mesh and material always overlap. We begin with Lagrangian and Eulerian schemes, which were previously employed by both ourselves and others, and are known to suffer from limitations (Barrett *et al.* 2015; Rodrigues *et al.* 2015; Sauer *et al.* 2017; Omar *et al.* 2020; Reuther *et al.* 2020; Sahu *et al.* 2020b). We then discuss a new class of ALE schemes in which the mesh velocity itself satisfies dynamical equations similar to those that govern the material velocity.

2.5. The Lagrangian mesh motion

We begin with the simplest membrane motion, namely a Lagrangian scheme. In this case, the mesh velocity is prescribed to be equal to the material velocity:

$$\mathbf{v}^m = \mathbf{v}. \quad (2.30)$$

In our numerical implementation, mesh velocity degrees of freedom are mapped to their material velocity counterparts such that (2.30) is satisfied identically over the entire surface. The direct Galerkin expression in the case of a Lagrangian mesh motion is then given by

$$\mathcal{G}^L := \mathcal{G}_{\lambda} + \mathcal{G}_v = 0 \quad \text{for all } \delta \lambda \in L^2(\Omega), \delta \mathbf{v} \in \mathbf{V}_0. \quad (2.31)$$

In (2.31) the superscript ‘ L ’ denotes a Lagrangian scheme, where $\mathbf{v}^m = \mathbf{v}$ throughout and mesh positions are updated according to (2.29). In addition, \mathcal{G}_λ and \mathcal{G}_v are respectively provided in (2.12) and (2.27).

2.6. The Eulerian mesh motion

The second scheme we implement is in-plane Eulerian, with $\mathbf{v}^m \cdot \mathbf{a}_{\check{\alpha}} = 0$, and out-of-plane Lagrangian, for which $\mathbf{v}^m \cdot \mathbf{n} = \mathbf{v} \cdot \mathbf{n}$. The in-plane and out-of-plane conditions are expressed as the single equation

$$\mathbf{v}^m = (\mathbf{n} \otimes \mathbf{n}) \mathbf{v}, \quad (2.32)$$

where ‘ \otimes ’ denotes the dyadic product. As there are no spatial derivatives in (2.32), no mesh velocity boundary conditions are to be prescribed for an Eulerian mesh motion.

The weak form of (2.32) is obtained by contracting it with an arbitrary mesh velocity variation $\delta \mathbf{v}^m \in \mathbf{H}^2(\Omega)$, integrating over the membrane surface, and multiplying the result by a user-specified constant α_m^E . The weak formulation of the Eulerian mesh velocity equation is then given by (Sahu *et al.* 2020b)

$$\mathcal{G}_m^E := \alpha_m^E \int_{\Omega} \delta \mathbf{v}^m \cdot [\mathbf{v}^m - (\mathbf{n} \otimes \mathbf{n}) \mathbf{v}] J_{\Omega} d\Omega = 0 \quad \text{for all } \delta \mathbf{v}^m \in \mathbf{H}^2(\Omega). \quad (2.33)$$

In (2.33) the superscript ‘ E ’ signifies the scheme is Eulerian, and the constant α_m^E has dimensions of energy \times time/length⁴ such that \mathcal{G}_m^E has the same dimensions as \mathcal{G}_λ and \mathcal{G}_v . In this study, $\alpha_m^E = \zeta / \ell^2$ for all results presented, as varying α_m^E was not observed to affect membrane behaviour (ℓ is a chosen characteristic length). The direct Galerkin expression for a scenario with an Eulerian mesh motion is expressed as

$$\mathcal{G}^E := \mathcal{G}_\lambda + \mathcal{G}_v + \mathcal{G}_m^E = 0 \quad \text{for all } \delta \lambda \in L^2(\Omega), \delta \mathbf{v} \in \mathbf{V}_0, \delta \mathbf{v}^m \in \mathbf{H}^2(\Omega). \quad (2.34)$$

2.7. The ALE mesh motion

The final scheme we consider is neither in-plane Lagrangian nor in-plane Eulerian. Instead, the mesh is modelled as a separate 2-D material with its own constitutive relations and associated dynamical equations. The mesh analogues of the stress vectors are then expressed as (cf. (2.14))

$$\mathbf{T}_m^{\check{\alpha}} = \sigma_m^{\check{\alpha}\check{\beta}} \mathbf{a}_{\check{\beta}} - (M_m^{\check{\beta}\check{\alpha}} \mathbf{n})_{;\check{\beta}}, \quad (2.35)$$

where $\sigma_m^{\check{\alpha}\check{\beta}}$ and $M_m^{\check{\alpha}\check{\beta}}$ are the mesh analogues of $\sigma^{\check{\alpha}\check{\beta}}$ and $M^{\check{\alpha}\check{\beta}}$. Here $\sigma_m^{\check{\alpha}\check{\beta}}$ and $M_m^{\check{\alpha}\check{\beta}}$ can be specified arbitrarily without altering the dynamics of the membrane, and so the mesh dynamics can be that of an elastic, viscous or viscoelastic material. When the mesh velocity \mathbf{v}^m is determined by solving an arbitrary set of governing equations, it will no longer satisfy the kinematic constraint (2.6) across the membrane surface. We enforce (2.6) with an additional Lagrange multiplier field, which physically acts on the mesh as an external body force per unit area in the normal direction – hereafter referred to as the mesh pressure p^m . Following the developments of § 2.3, the dynamical equation governing the mesh dynamics is given by

$$\mathbf{T}_{m;\check{\alpha}}^{\check{\alpha}} + p^m \mathbf{n} = \mathbf{0}. \quad (2.36)$$

Upon specification of appropriate boundary conditions, (2.6) and the three components of (2.36) uniquely determine the mesh pressure p^m and the three components of the mesh velocity \mathbf{v}^m . The ability to specify mesh boundary conditions can prevent undesirable

mesh distortions, and is an advantage of our ALE method. Two choices of the mesh motion, for which $\sigma_m^{\check{\alpha}\check{\beta}}$ and $M_m^{\check{\alpha}\check{\beta}}$ are prescribed, are discussed subsequently. In the first case, the mesh is area-compressible and purely viscous; in the second case it also resists bending. In both cases, the simplest boundary conditions for the mesh velocity are chosen. The investigation of more complex boundary conditions, as well as more involved mesh behaviours, is left to a future study.

2.7.1. The weak formulation of the kinematic constraint

The mesh pressure is an unknown Lagrange multiplier field enforcing the ALE kinematic constraint in (2.6). The weak form of (2.6) is obtained by multiplying it with an arbitrary variation $\delta p^m \in L^2(\Omega)$, and integrating the result over the membrane surface to obtain

$$\int_{\Omega} \delta p^m \mathbf{n} \cdot (\mathbf{v}^m - \mathbf{v}) J_{\Omega} d\Omega = 0 \quad \text{for all } \delta p^m \in L^2(\Omega). \quad (2.37)$$

Equation (2.37) bears some resemblance to (2.8), which was obtained from the incompressibility constraint. In both equations, the variation of a scalar Lagrange multiplier interacts with a vector velocity. Moreover, when (2.37) is discretised, our numerical simulations exhibit an instability reminiscent of the checkerboarding that arises when the LBB condition is violated. We thus hypothesise that both numerical instabilities are similar, and attempt to stabilise the mesh pressure with the Dohrmann–Bochev method. Following the results of § 2.2.1, we express the weak formulation of the kinematic constraint (2.6) as

$$\mathcal{G}_p = 0 \quad \text{for all } \delta p^m \in L^2(\Omega), \quad (2.38)$$

where the numerically stabilised mesh pressure contribution to the direct Galerkin expression is given by (cf. (2.12), (2.37))

$$\mathcal{G}_p := - \int_{\Omega} \delta p^m \mathbf{n} \cdot (\mathbf{v}^m - \mathbf{v}) J_{\Omega} d\Omega - \frac{\alpha^{DB} \ell^2}{\zeta} \int_{\Omega} (\delta p^m - \delta \check{p}^m)(p^m - \check{p}^m) d\Omega. \quad (2.39)$$

In (2.39), \check{p}^m and $\delta \check{p}^m$ are the L^2 projections of p^m and δp^m onto the space \check{L} , as defined through (2.9). The factor of ℓ^2 , for a characteristic length ℓ , is required for dimensional consistency. Our choice to employ the Dohrmann–Bochev method to stabilise p^m does not yet sit on a firm theoretical footing, and currently can only be justified with empirical success. We hope to investigate the mathematical nature of the stabilisation of p^m in a future paper.

2.7.2. The case where the mesh dynamics is purely viscous

For the case of purely viscous mesh dynamics, quantities are denoted with a superscript or subscript ‘ v ’. The scheme is referred to as ‘ALE-viscous’, with the shorthand ‘Av’ or ‘ALE-v’. Since the mesh velocity field is area-compressible, the mesh motion resists both shearing and dilatation.

Strong formulation: we prescribe for the mesh analogue of the in-plane stresses and couple stresses to be (cf. (2.15)–(2.17))

$$\sigma_{m,v}^{\check{\alpha}\check{\beta}} = \pi_m^{\check{\alpha}\check{\beta}} := \zeta^m \mathbf{v}_{,\check{\mu}}^m \cdot (\mathbf{a}^{\check{\alpha}} a^{\check{\mu}\check{\beta}} + \mathbf{a}^{\check{\beta}} a^{\check{\mu}\check{\alpha}}) \quad \text{and} \quad M_{m,v}^{\check{\alpha}\check{\beta}} = 0, \quad (2.40)$$

where ζ^m is a user-specified parameter that we refer to as the mesh viscosity. Importantly, since the mesh velocity is area-compressible, our choice in (2.40) resists both shear and dilatation of the mesh. By substituting (2.35) and (2.40) into (2.36), the dynamical

equations governing the mesh are found to be (Sahu 2022, Chapter V, § 5(c))

$$\zeta^m \left(\Delta_s v_{\check{\alpha}}^m + K v_{\check{\alpha}}^m + 2(v^m)_{,\check{\alpha}} H - 2(v^m)_{,\check{\beta}} b_{\check{\alpha}}^{\check{\beta}} - 2v^m H_{,\check{\alpha}} + (a^{\check{\beta}} (v_{\check{\gamma}}^m)_{,\check{\beta}} - 2v^m H)_{,\check{\alpha}} \right) = 0 \quad (2.41)$$

and

$$p^m + \zeta^m \left(2b^{\check{\alpha}\check{\beta}} (v_{\check{\alpha}}^m)_{,\check{\beta}} - 8v^m H^2 + 4v^m K \right) = 0, \quad (2.42)$$

where $\mathbf{v}^m = v_{\check{\alpha}}^m \mathbf{a}^{\check{\alpha}} + v^m \mathbf{n}$. The solution to (2.41) is independent of ζ^m , and the mesh pressure in (2.42) will adjust so as to satisfy the kinematic constraint (2.6). The mesh dynamics accordingly do not depend on the choice of ζ^m . In all numerical results presented, we choose $\zeta^m = \zeta$ for simplicity.

Boundary conditions: since the mesh velocity satisfies differential equations (see (2.41) and (2.42)) rather than an algebraic equation (cf. (2.30), (2.32)), boundary conditions for the mesh velocity are required. To this end, we introduce the mesh analogue of the force per length on the boundary:

$$\mathbf{F}_v^m = \mathbf{T}_{m,v}^{\check{\alpha}} v_{\check{\alpha}} = \sigma_{m,v}^{\check{\alpha}\check{\beta}} v_{\check{\alpha}} \mathbf{a}_{\check{\beta}}. \quad (2.43)$$

Recalling $M_{m,v}^{\check{\alpha}\check{\beta}} = 0$ according to (2.40), the purely viscous mesh dynamics cannot sustain moments – as is the case for a fluid film (Sahu *et al.* 2020b). Consequently, neither slope nor moment boundary conditions are required on the boundary, since the mesh analogue of the moment $M^m = 0$ identically throughout. With this understanding, the parametric boundary Γ is partitioned into the portions Γ_v^{mj} and Γ_F^{mj} , on which $v_j^m := \mathbf{v}^m \cdot \mathbf{e}_j$ and $F_j^m := \mathbf{F}^m \cdot \mathbf{e}_j$ are respectively specified. For simplicity in our numerical formulation, the mesh is assumed to be either static or force-free on its entire boundary, for which (cf. (2.21))

$$v_j^m = 0 \quad \text{on} \quad \Gamma_v^{mj} \quad \text{and} \quad F_j^m = 0 \quad \text{on} \quad \Gamma_F^{mj}. \quad (2.44)$$

Weak formulation: the weak formulation of the purely viscous ALE mesh behaviour is obtained by drawing analogy to the developments in § 2.3.2. First, the space of arbitrary mesh velocity variations is defined as

$$\mathcal{V}_{0,v}^m := \left\{ \mathbf{u}(\zeta^{\check{\alpha}}) : \Omega \rightarrow \mathbb{R}^3 \text{ such that } \mathbf{u} \in \mathbf{H}^2(\Omega), u_j|_{\Gamma_v^{mj}} = 0 \right\}. \quad (2.45)$$

By contracting (2.36) with an arbitrary mesh velocity variation $\delta \mathbf{v}^m \in \mathcal{V}_{0,v}^m$ and repeating the steps of § 2.3.2, the weak formulation of the ALE-viscous mesh motion is found to be (cf. (2.26) and (2.27))

$$\mathcal{G}_m^{Av} = 0 \quad \text{for all} \quad \delta \mathbf{v}^m \in \mathcal{V}_{0,v}^m, \quad (2.46)$$

where

$$\mathcal{G}_m^{Av} := \int_{\Omega} \frac{1}{2} \left(\delta \mathbf{v}_{,\check{\alpha}}^m \cdot \mathbf{a}_{\check{\beta}} + \delta \mathbf{v}_{,\check{\beta}}^m \cdot \mathbf{a}_{\check{\alpha}} \right) \sigma_{m,v}^{\check{\alpha}\check{\beta}} J_{\Omega}. \quad (2.47)$$

In (2.46) and (2.47), the superscript ‘Av’ refers to the choice of a purely viscous ALE mesh motion. The direct Galerkin expression for this motion is then written as

$$\mathcal{G}^{Av} := \mathcal{G}_{\lambda} + \mathcal{G}_v + \mathcal{G}_m^{Av} + \mathcal{G}_p = 0$$

$$\text{for all} \quad \delta \lambda \in L^2(\Omega), \delta \mathbf{v} \in \mathcal{V}_0, \delta \mathbf{v}^m \in \mathcal{V}_{0,v}^m, \delta p^m \in L^2(\Omega). \quad (2.48)$$

2.7.3. The case where the mesh dynamics is viscous and resist bending

We now consider the case where the mesh dynamics is viscous in response to in-plane shears and area dilatations, and additionally resist bending. The mesh motion is referred to as ‘ALE-viscous-bending’, with shorthand ‘Avb’ or ‘ALE-vb’. Corresponding quantities are denoted with a subscript or superscript ‘vb’.

Strong formulation: we choose for the bending resistance of the mesh to arise in the same manner as the membrane itself, with the viscous resistance identical to that of the ALE-viscous scenario. To this end, we choose (cf. (2.15)–(2.17) and (2.40))

$$M_{m,vb}^{\check{\alpha}\check{\beta}} = k_b^m H a^{\check{\alpha}\check{\beta}} + k_g^m (2 H a^{\check{\alpha}\check{\beta}} - b^{\check{\alpha}\check{\beta}}) \quad (2.49)$$

and

$$\sigma_{m,vb}^{\check{\alpha}\check{\beta}} = k_b^m (H^2 a^{\check{\alpha}\check{\beta}} - 2 H b^{\check{\alpha}\check{\beta}}) - k_g^m K a^{\check{\alpha}\check{\beta}} + \pi_m^{\check{\alpha}\check{\beta}}, \quad (2.50)$$

where $\pi_m^{\check{\alpha}\check{\beta}}$ is defined in (2.40). In (2.49) and (2.50), k_b^m and k_g^m are user-specified parameters; we always choose $k_b^m = k_b$ and $k_g^m = k_g$ for simplicity. It is well known that bending terms from the Helfrich free energy do not enter the in-plane equations (Rangamani *et al.* 2012), and so the in-plane dynamical mesh equations in the ALE-vb case are given by (2.41). The out-of-plane dynamical equation is expressed as (cf. (2.19) and (2.42))

$$p^m + \zeta^m (2 b^{\check{\alpha}\check{\beta}} v_{\check{\alpha};\check{\beta}} - 8 v H^2 + 4 v K) - k_b^m (2 H^3 - 2 H K + \Delta_s H) = 0. \quad (2.51)$$

Contrary to the ALE-v scenario (2.42), here the relative magnitude of ζ^m and k_b^m do affect the resultant mesh pressure p^m . Investigating the relationship between p^m , ζ^m and k_b^m – as well as the ensuing mesh dynamics – is left to a later study.

Boundary conditions: we once again introduce the mesh analogue of the force per length on the boundary, which in this case is written as (cf. (2.20) and (2.43))

$$\mathbf{F}_{vb}^m = \mathbf{T}_{m,vb}^{\check{\alpha}} v_{\check{\alpha}} - \left(M_{m,vb}^{\check{\alpha}\check{\beta}} v_{\check{\alpha}} \tau_{\check{\beta}} \mathbf{n} \right)_{,\check{\mu}} \tau^{\check{\mu}}. \quad (2.52)$$

In principle, one could specify boundary forces for the mesh dynamics. However, as in the ALE-viscous case, we choose simple boundary conditions where the mesh is either static or force-free on the entire boundary (2.44). In addition, since the mesh couple stresses are non-zero, either slope or moment boundary conditions are required for \mathbf{v}^m – where the analogue of the boundary moment is calculated as $M^m = M_m^{\check{\alpha}\check{\mu}} v_{\check{\alpha}} v_{\check{\mu}}$ (cf. (2.22)). At present, we take the simpler approach and specify zero-slope conditions on the entire boundary (cf. (2.23)):

$$\mathbf{n} \cdot \mathbf{v}_{,\check{\alpha}}^m v^{\check{\alpha}} = 0 \quad \text{on } \Gamma. \quad (2.53)$$

The ability to specify the slope of the mesh at a boundary will be relevant in the tether pulling scenario of § 4.2.

Weak formulation: at this point, it is straightforward to write the weak formulation of the mesh dynamics. The space of arbitrary mesh variations is given by

$$\mathcal{V}_{0,vb}^m := \left\{ \mathbf{u}(\zeta^{\check{\alpha}}) : \Omega \rightarrow \mathbb{R}^3 \text{ such that } \mathbf{u} \in \mathbf{H}^2(\Omega), u_j|_{\Gamma_v^{mj}} = 0, (\mathbf{n} \cdot \mathbf{u}_{,\check{\alpha}} v^{\check{\alpha}})|_{\Gamma} = 0 \right\}. \quad (2.54)$$

The weak formulation is then expressed as

$$\mathcal{G}_m^{Avb} = 0 \quad \text{for all } \delta \mathbf{v}^m \in \mathcal{V}_{0,vb}^m, \quad (2.55)$$

where we define (cf. (2.27))

$$\begin{aligned}\mathcal{G}_m^{Avb} := & \int_{\Omega} \frac{1}{2} \left(\delta \mathbf{v}_{,\check{\alpha}}^m \cdot \mathbf{a}_{\check{\beta}} + \delta \mathbf{v}_{,\check{\beta}}^m \cdot \mathbf{a}_{\check{\alpha}} \right) \sigma_{m,vb}^{\check{\alpha}\check{\beta}} J_{\Omega} d\Omega \\ & + \int_{\Omega} \frac{1}{2} \left(\delta \mathbf{v}_{,\check{\alpha}\check{\beta}}^m + \delta \mathbf{v}_{,\check{\beta}\check{\alpha}}^m \right) \cdot \mathbf{n} M_{m,vb}^{\check{\alpha}\check{\beta}} J_{\Omega} d\Omega - \int_{\Omega} (\delta \mathbf{v}^m \cdot p^m \mathbf{n}) J_{\Omega} d\Omega.\end{aligned}\quad (2.56)$$

The direct Galerkin expression for the scenario with the ALE-vb mesh motion is written as

$$\begin{aligned}\mathcal{G}^{Avb} := & \mathcal{G}_{\lambda} + \mathcal{G}_v + \mathcal{G}_m^{Avb} + \mathcal{G}_p = 0 \\ \text{for all } & \delta \lambda \in L^2(\Omega), \delta \mathbf{v} \in \mathcal{V}_0, \delta \mathbf{v}^m \in \mathcal{V}_{0,vb}^m, \delta p^m \in L^2(\Omega).\end{aligned}\quad (2.57)$$

3. The finite element formulation

With the Lagrangian, Eulerian and ALE weak formulations, our goal is to solve for the state of the membrane over time. We seek to determine (i) the membrane velocity \mathbf{v} , (ii) the mesh velocity \mathbf{v}^m , and (iii) the membrane tension λ – for all parametric points $\zeta^{\check{\alpha}} \in \Omega$ and times $t \in [0, t_f]$. In doing so, the surface position is obtained from the mesh velocity through (2.29). We cannot solve for the highly nonlinear membrane behaviour exactly, and turn to the finite element method to numerically calculate the approximate solutions \mathbf{v}_h , \mathbf{v}_h^m and λ_h . An overview of our solution method is presented in what follows. Further specifics can be found in [Appendix A](#) and in the MembraneAleFem.jl documentation ([Sahu 2024](#)).

3.1. The surface discretisation and space of solutions

Let us begin by assuming there is a discretisation \mathcal{T}_h of the parametric domain Ω into nel (number of elements) non-overlapping finite elements Ω^e of characteristic length h :

$$\mathcal{T}_h := \{\Omega^1, \Omega^2, \dots, \Omega^{\text{nel}}\}. \quad (3.1)$$

Our task is to choose the set of basis functions over each element $\Omega^e \in \mathcal{T}_h$ – denoted $\{N_k^e(\zeta^{\check{\alpha}})\}$, where the index k ranges from 1 to nen (number of elemental nodes) – such that the resultant solution spaces are consistent with their infinite-dimensional counterparts. A complexity arises because mesh and membrane velocities belong to the space $H^2(\Omega)$. Accordingly, \mathbf{v}_h and \mathbf{v}_h^m are required to be continuous and have continuous first derivatives across elements. We satisfy the continuity criteria with a so-called tensor product of quadratic B-spline basis functions in each of the ζ^1 and ζ^2 directions ([Piegl & Tiller 1997](#); [Cottrell, Hughes & Bazilevs 2009](#)). For our purposes, the parametric domain Ω is partitioned into a rectangular grid of finite elements, though more advanced discretisations are now established ([Toshniwal, Speleers & Hughes 2017](#); [Wei *et al.* 2018](#); [Paul *et al.* 2020](#); [Koh, Toshniwal & Cirak 2022](#)). The resultant scalar and vector finite-dimensional spaces \mathcal{U}_h and \mathcal{U}_h^m are respectively given by

$$\mathcal{U}_h := \left\{ u(\zeta^{\check{\alpha}}) : \Omega \rightarrow \mathbb{R} \text{ such that } u \in C^1(\Omega) \cap H^2(\Omega), u|_{\Omega^e} \in \mathbb{Q}_2(\Omega^e) \text{ for all } \Omega^e \in \mathcal{T}_h \right\} \quad (3.2)$$

and

$$\mathcal{U}_h^m := \left\{ \mathbf{u}(\zeta^{\check{\alpha}}) : \Omega \rightarrow \mathbb{R}^3 \text{ such that } u_j \in \mathcal{U}_h \right\}, \quad (3.3)$$

for which

$$\mathbf{v}_h \in \mathcal{U}_h \quad \text{and} \quad \mathbf{v}_h^m \in \mathcal{U}_h \quad (3.4)$$

by construction. In (3.2), $C^m(\Omega)$ denotes the space of scalar functions on Ω with m continuous derivatives, and $\mathbb{Q}_n(\Omega^e)$ denotes the tensor-product space of n th-order polynomials in the ξ^1 and ξ^2 directions on the finite element Ω^e . The arbitrary variations $\delta\mathbf{v}_h$ and $\delta\mathbf{v}_h^m$ are also assumed to lie in \mathcal{U}_h , and are respectively elements of the spaces

$$\mathcal{V}_{0,h} := \mathcal{V}_0 \cap \mathcal{U}_h \quad \text{and} \quad \mathcal{V}_{0,h}^m := \mathcal{V}_0^m \cap \mathcal{U}_h. \quad (3.5)$$

3.1.1. The membrane tension, mesh pressure and Dohrmann–Bochev projection

In general, the Lagrange multipliers λ and p^m are elements of $L^2(\Omega)$, and their finite-dimensional counterparts λ_h and p_h^m need not be restricted to lie in \mathcal{U}_h . However, for convenience in the numerical implementation, we do in fact choose

$$\lambda_h \in \mathcal{U}_h \quad \text{and} \quad p_h^m \in \mathcal{U}_h. \quad (3.6)$$

Since the same basis functions are used for the velocity and surface tension, the well-known inf–sup condition is violated and the LBB instability arises (Ladyzhenskaya 1969; Babuška 1973; Brezzi 1974). We apply the method of Dohrmann & Bochev (2004) to stabilise unphysical oscillations in both the surface tension and mesh pressure. To do so, λ_h and p_h^m are projected onto the space

$$\check{\mathcal{L}} := \left\{ u(\xi^\alpha) : \Omega \rightarrow \mathbb{R} \text{ such that } u|_{\Omega^e} \in \mathbb{P}_1(\Omega^e) \text{ for all } \Omega^e \in \mathcal{T}_h \right\}, \quad (3.7)$$

where $\mathbb{P}_n(\Omega^e)$ denotes the space of polynomials of order n on Ω^e . Since functions in $\check{\mathcal{L}}$ form a plane over each rectangular element Ω^e , they are discontinuous across finite elements.

3.2. The method of numerical solution

An overview of our numerical implementation is presented below. Additional details are provided in Appendix A and the MembraneAleFem.jl package repository (Sahu 2024).

Once the parametric domain is discretised into finite elements, all membrane unknowns are expressed as the sum of B-spline basis functions multiplied by membrane degrees of freedom. The degrees of freedom are collected into a column vector and are generically denoted $[\mathbf{u}]$; they are also referred to as nodal values and their dependence on time is implied. Variations of membrane unknowns are similarly decomposed, with degree-of-freedom variations compatible with all boundary conditions collected into the column vector $[\delta\mathbf{u}]$. Since the direct Galerkin expression \mathcal{G} is linear in the arbitrary variations, its discretised counterpart \mathcal{G}_h is expressed as

$$\mathcal{G}_h([\delta\mathbf{u}], [\mathbf{u}]) = [\delta\mathbf{u}]^T [\mathbf{r}([\mathbf{u}])] = 0 \quad \text{for all } [\delta\mathbf{u}]. \quad (3.8)$$

In (3.8) the residual vector $[\mathbf{r}]$ is defined to be

$$[\mathbf{r}([\mathbf{u}])] := \frac{\partial \mathcal{G}_h([\delta\mathbf{u}], [\mathbf{u}])}{\partial [\delta\mathbf{u}]} \quad (3.9)$$

Since the variations $[\delta\mathbf{u}]$ in (3.9) are arbitrary, the membrane dynamics at any time $t \in [0, t_f]$ satisfy the nonlinear equation

$$[\mathbf{r}([\mathbf{u}])] = [\mathbf{0}]. \quad (3.10)$$

Equation (3.10) is solved with the Newton–Raphson method, where a sequence of progressively better estimates of the membrane state $[\mathbf{u}]$ is generated. Given the j th estimate $[\mathbf{u}]_j$, the $(j+1)$ th estimate is calculated according to

$$[\mathbf{u}]_{j+1} = [\mathbf{u}]_j - [\mathbf{K}]_j^{-1} [\mathbf{r}([\mathbf{u}]_j)], \quad (3.11)$$

where the global tangent diffusion matrix at the j th iteration is defined as

$$[\mathbf{K}]_j := \left. \frac{\partial [\mathbf{r}([\mathbf{u}])]}{\partial [\mathbf{u}]} \right|_{[\mathbf{u}]_j}. \quad (3.12)$$

In prior studies (Sauer *et al.* 2017; Sahu *et al.* 2020b), the tangent diffusion matrix was calculated analytically: an involved task. In the present work, however, $[\mathbf{K}]$ is calculated by numerically differentiating $[\mathbf{r}]$ with respect to each entry of $[\mathbf{u}]$. In particular, the tangent matrix (3.12) is calculated to machine precision by extending $[\mathbf{r}]$ and $[\mathbf{u}]$ into the complex plane, following the general developments of Lyness & Moler (1967), Lyness (1968) (see also, e.g. Tanaka *et al.* 2014). We note that when calculating derivatives with respect to mesh velocity degrees of freedom, one must also perturb the surface positions, as the two are related through (2.28) and (2.29).

4. The results of numerical simulations

We now present results from numerical simulations, where our finite element implementation – including the various mesh motions discussed previously – was used to simulate the dynamics of lipid membranes. We first validate our code against a standard numerical benchmark, namely pure bending of a lipid membrane patch. We then simulate the drawing of a tube from a membrane sheet: a process known to be important in various biological phenomena, including dynamic rearrangement of the endoplasmic reticulum (Terasaki *et al.* 1986, 2013) and tether formation by proteins travelling along microtubules (Leduc *et al.* 2004). The well-known quasi-static tether pulling behaviour (Powers *et al.* 2002) is confirmed, and the effect of pull speed on pull force is quantified. We close by laterally translating the tether across the membrane surface with our ALE scheme – a result which cannot be obtained with established Lagrangian or Eulerian methods.

4.1. The pure bending of a flat patch

We begin with a simple scenario: starting with a flat membrane patch, boundary moments are applied to the left and right edges to bend the membrane, as shown schematically in figure 2(a). For a given applied moment, the equilibrium configuration is known analytically to be a portion of a cylinder (Sauer *et al.* 2017; Sauer & Duong 2017). The case of pure bending is thus used to validate our numerical implementation, including the three mesh motions. For a given observable, errors in the numerical result u_h with respect to the known analytical solution u are calculated as

$$E_h^u := \left(\int_{\Omega} |u - u_h|^2 d\Omega \right)^{1/2}, \quad (4.1)$$

where h denotes the characteristic length of a finite element in the parametric domain.

4.1.1. The problem set-up

In our code, the membrane patch is initially a square of side length ℓ in the x – y plane. The initial velocity and position are respectively given by

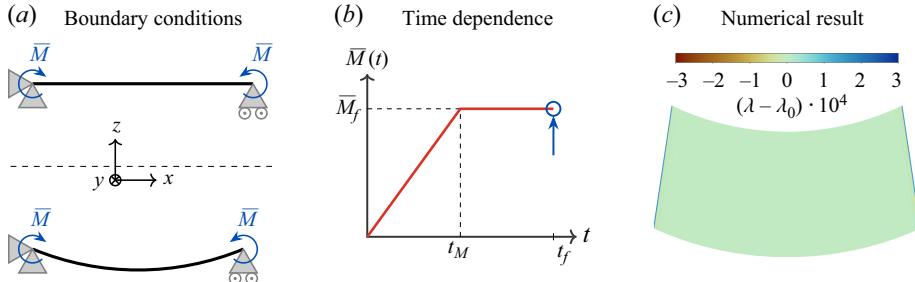


Figure 2. Pure bending scenario. (a) Boundary moments are applied to an initially flat patch in the x – y plane (top). The right edge is free to move in the x and y directions, but is constrained in the z direction. At equilibrium, the membrane forms a portion of a cylinder (bottom). (b) Time dependence of the prescribed boundary moment, according to (4.4). Errors are calculated at the final time t_f , as indicated by the blue arrow and circle. (c) Result of a numerical simulation on a 128×128 mesh. The calculated surface tension differs from the analytical solution by less than three parts in 10 000, as shown by the colour bar.

$$\mathbf{v}(\zeta^{\check{\alpha}}, t \leq 0) = \mathbf{0} \quad \text{and} \quad \mathbf{x}(\zeta^{\check{\alpha}}, t \leq 0) = x \mathbf{e}_x + y \mathbf{e}_y, \quad (4.2)$$

where $\zeta^{\check{\alpha}} \in \Omega := [0, 1] \times [0, 1]$ and the real-space coordinates x and y are parametrised as

$$x(\zeta^{\check{\alpha}}, t \leq 0) = \ell \zeta^{\check{1}} \quad \text{and} \quad y(\zeta^{\check{\alpha}}, t \leq 0) = \ell \zeta^{\check{2}}. \quad (4.3)$$

The magnitude of the applied moment, denoted $\bar{M}(t)$, is linearly ramped up over time until $t = t_M$ – after which the boundary moment is held constant at the final value \bar{M}_f (see figure 2b):

$$\bar{M}(t) = \begin{cases} 0, & t \leq 0, \\ (t/t_M) \bar{M}_f, & 0 < t < t_M, \\ \bar{M}_f, & t \geq t_M. \end{cases} \quad (4.4)$$

For times $t > t_M$, the applied boundary moment is constant in time and the membrane is bent into a portion of a cylinder, as shown in figure 2(c). The boundary forces and moments of this stationary solution are calculated in [Appendix B](#) and summarised in [table 1](#). Here, r_c and λ_0 are respectively the cylinder radius and surface tension, the latter of which is constant in the equilibrium configuration. We choose the Gaussian bending modulus $k_g = -k_b/2$ such that $M = 0$ on the top and bottom edges, and no body force ($f = \mathbf{0}$) such that $\lambda_0 = k_b/(4r_c^2)$ and $\mathbf{F} = \mathbf{0}$ on the left and right edges (see [table 1](#)). In this case, the cylinder radius r_c is related to the final boundary moment \bar{M}_f according to

$$r_c = \frac{k_b}{2\bar{M}_f}. \quad (4.5)$$

All of the boundary conditions prescribed in our numerical implementation for Lagrangian and Eulerian mesh motions are provided in [table 2](#). For the ALE simulations, we choose the ALE-v mesh motion discussed in § 2.7.2. The first three rows of [table 2](#) are then repeated for the mesh counterparts \bar{v}_j^m and \bar{F}_j^m . Recall that $M^m = 0$ by construction, so there is no mesh analogue of the moment boundary condition. As a result, the membrane deforms only due to the physically applied moment $\bar{M}(t)$ – which is also the case for the Lagrangian and Eulerian simulations.

	Top	Bottom	Left	Right
M	$\frac{k_b}{2r_c} + \frac{k_g}{r_c}$	$\frac{k_b}{2r_c} + \frac{k_g}{r_c}$	$\frac{k_b}{2r_c}$	$\frac{k_b}{2r_c}$
F	$\left[\frac{k_b}{4r_c^2} + \lambda_0 \right] \mathbf{e}_y$	$- \left[\frac{k_b}{4r_c^2} + \lambda_0 \right] \mathbf{e}_y$	$\left[\lambda_0 - \frac{k_b}{4r_c^2} \right] \mathbf{e}_\theta$	$\left[\frac{k_b}{4r_c^2} - \lambda_0 \right] \mathbf{e}_\theta$

Table 1. Moments and forces on the boundary of a portion of a cylindrical membrane with radius r_c , as shown in figure 2 and calculated in Appendix B. The constant surface tension λ_0 is set by the net body force per area in the normal direction, $f := f \cdot \mathbf{n}$. In figure 2(a) the cylindrical basis vector \mathbf{e}_θ is tangent to the black curve and points to the left.

	Top	Bottom	Left	Right
$\bar{F}_x = 0$	$\bar{F}_x = 0$	$\bar{v}_x = 0$	$\bar{F}_x = 0$	
$\bar{v}_y = 0$	$\bar{v}_y = 0$	$\bar{v}_y = 0$	$\bar{F}_y = 0$	
$\bar{F}_z = 0$	$\bar{F}_z = 0$	$\bar{v}_z = 0$	$\bar{v}_z = 0$	
$\bar{M} = 0$	$\bar{M} = 0$	$\bar{M} = \bar{M}(t)$	$\bar{M} = \bar{M}(t)$	

Table 2. Boundary conditions prescribed in the numerical implementation. We choose to set $k_g = -k_b/2$ and $f = \mathbf{0}$; the latter yields $\lambda_0 = k_b/(4r_c^2)$. The prescribed moment $\bar{M}(t)$ is given by (4.4). The first three rows are repeated for \bar{v}_j^m and \bar{F}_j^m in the case of a viscous ALE mesh motion.

4.1.2. Non-dimensionalisation

For the case of pure bending, there are five membrane parameters: the bending modulus k_b , final bending moment \bar{M}_f , ramp-up time t_M , patch length ℓ and intramembrane viscosity ζ . These parameters all dimensionally consist of mass, length and time. The pure bending scenario is thus completely described by two dimensionless quantities. The first is the Föppl–von Kármán number, which compares surface tension to bending forces (Sahu *et al.* 2020a). For a cylinder with $f = \mathbf{0}$, the membrane surface tension is constant: $\lambda = \lambda_0 := k_b/(4r_c^2) = \bar{M}_f^2/k_b$, where (4.5) was substituted in the last equality (Evans & Yeung 1994). The Föppl–von Kármán number Γ is then given by

$$\Gamma = \frac{\lambda_0 \ell^2}{k_b} = \left(\frac{\bar{M}_f \ell}{k_b} \right)^2. \quad (4.6)$$

The second dimensionless quantity is the Scriven–Love number S_L , which captures dynamical effects and in this scenario is set by t_M (Sahu *et al.* 2020a). As the characteristic velocity scale of membrane deformations is given by ℓ/t_M , the Scriven–Love number is expressed as

$$S_L = \frac{\ell^2 \zeta}{k_b t_M}, \quad (4.7)$$

which can be understood as a ratio between the fundamental membrane time scale $\ell^2 \zeta / k_b$ and the ramp-up time t_M . For the results presented, we run simulations with $\zeta = 1.0$, $k_b = 1.0$ and $\ell = 1.0$. The Föppl–von Kármán and Scriven–Love numbers are then set by choosing \bar{M}_f and t_M , respectively.

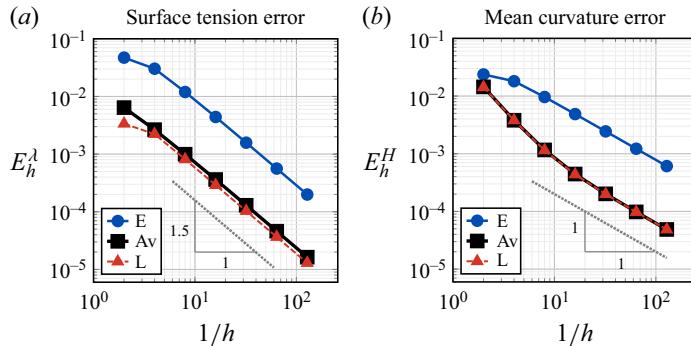


Figure 3. Errors in the (a) surface tension and (b) mean curvature at the final time t_f , according to (4.1), when the membrane patch is subjected to pure bending boundary conditions. The mesh consists of $nel = 1/h^2$ parametric area elements, each of which is a square with side length h . Here, $1/h$ ranges from 2^1 to 2^7 in powers of two. The labels ‘E’, ‘Av’ and ‘L’ refer to Eulerian, ALE-viscous and Lagrangian mesh motions. In all cases, the convergence of the error confirms our numerical implementation is working as expected. Relevant parameters are specified in § 4.1.3; we also choose $\zeta = 1.0$, $k_b = 1.0$ and $\ell = 1.0$.

4.1.3. Results

We simulate pure bending scenarios where $\Gamma = 0.25$ and $SL = 0.5$, for which $\bar{M}_f = k_b/(2\ell)$ and $t_M = 2\ell^2\zeta/k_b$ (see (4.6) and (4.7)). Lagrangian, Eulerian and ALE-viscous simulations are carried out on meshes ranging from 2×2 to 128×128 . Other relevant parameters are $\alpha^{DB} = \ell^2$, $\alpha_m^E = \zeta/\ell^2$, $\Delta t = 0.1\ell^2\zeta/k_b$ and $t_f = 4t_M$. The final configuration from one such simulation, at time t_f , is shown in figure 2(c). The L^2 error in surface tension and mean curvature, relative to the analytical solution and calculated at time t_f according to (4.1), is plotted as a function of mesh size in figure 3. Note that the analytical solution is valid only for a static patch; we thus deform the membrane slowly relative to the intrinsic time scale $\ell^2\zeta/k_b$ and calculate errors after it has relaxed sufficiently. All three mesh motions converge towards the analytical solution upon mesh refinement, indicating our numerical implementation is working as expected. We believe the error from the Eulerian mesh motion is larger than that of the other two schemes because boundary conditions are not prescribed for the mesh velocity. Instead, material velocity boundary conditions are prescribed and weakly communicated to the mesh velocity through (2.33); edges are also the location where errors are the largest (see figure 2c).

4.2. The pulling of a tether from a flat sheet

When a single point on the membrane is displaced in the direction \mathbf{n} normal to the surface, a tent-like shape forms when deformations are small. As the point continues to be displaced, the bilayer undergoes a non-trivial morphological transition and forms a cylindrical tether (Evans & Yeung 1994; Dai & Sheetz 1995; Fygenson, Marko & Libchaber 1997; Raucher & Sheetz 1999). Tethers are known to arise in biological settings, including in the endoplasmic reticulum (Terasaki *et al.* 1986; Scott *et al.* 2024) and at the junction between cells (Sowinski *et al.* 2008; Dubey *et al.* 2016; Imachi *et al.* 2020). In addition, tethers form in *in vitro* settings with optical tweezers (Evans & Yeung 1994; Cuvelier *et al.* 2005) and through the polymerisation of microtubules (Fygenson *et al.* 1997) – possibly with the use of molecular motors (Roux *et al.* 2002; Koster *et al.* 2003). Tether pulling is also commonly used in assays to probe both static and dynamic membrane properties, as one can measure the force \mathcal{F}_p required to pull the tether (Cuvelier *et al.* 2005;

	Bdry	Inner	Pull	Pin
LAG	$v_z = 0$ $\bar{\mathbf{F}}_{\parallel} = \lambda_0 \mathbf{v}$	$v_z = 0$	$\mathbf{v} = \mathbf{v}_p(t)$	$v_x = v_y = 0$
EUL	$v_z = 0$ $\bar{\mathbf{F}}_{\parallel} = \lambda_0 \mathbf{v}$	$v_z = 0$	$\mathbf{v} = \mathbf{v}_p(t)$	$v_x = v_y = 0$ $v_x^m = v_y^m = 0$
ALE-vb	$v_z = 0$ $\mathbf{v}^m = \mathbf{0}$ $\bar{\mathbf{F}}_{\parallel} = \lambda_0 \mathbf{v}$	$v_z = 0$ $v_z^m = 0$	$\mathbf{v} = \mathbf{v}_p(t)$ $\mathbf{v}^m = \mathbf{v}_p(t)$	$v_x = v_y = 0$ $v_x^m = v_y^m = 0$

Table 3. Boundary conditions prescribed in our numerical implementation to pull a tether, with Lagrangian (LAG), Eulerian (EUL) and ALE-viscous-bending (ALE-vb) mesh motions. Here, ‘Bdry’ refers to nodes on the boundary, ‘Inner’ refers to inner nodes adjacent to the boundary, ‘Pull’ corresponds to nodes associated with the central element Ω_p^e and ‘Pin’ refers to nodes at the centre of each edge. The latter conditions are required to prevent rigid body rotations and translations. The ‘Inner’ column enforces zero-slope boundary conditions.

Koster *et al.* 2005; Shi *et al.* 2018). An analysis of membrane tether pulling is thus relevant to both biological and *in vitro* scenarios. In what follows, we focus on the simplest case, in which a tether is pulled from an initially flat sheet.

4.2.1. The problem set-up

In numerical simulations the membrane patch is initially a square of side length ℓ in the x - y plane, centred at the origin. We set $v_z = 0$ on the entire patch boundary, and also pin the centre node of each edge ($\mathbf{v} = \mathbf{0}$) to remove rigid body rotations and translations. In addition, zero-slope boundary conditions are enforced by constraining $v_z = 0$ for nodes adjacent to the boundary – hereafter referred to as inner boundary nodes. Finally, an in-plane force/length $\bar{\mathbf{F}}_{\parallel} = \lambda_0 \mathbf{v}$ is applied on the boundary, where \mathbf{v} is the in-plane unit normal at the patch boundary and λ_0 is the static surface tension – which we choose. All boundary conditions are summarised in table 3, including those for the Eulerian and ALE schemes where \mathbf{v}^m is also a fundamental unknown.

The initial membrane state is given by

$$\mathbf{v}(\zeta^{\check{\alpha}}, t \leq 0) = \mathbf{0} \quad \text{and} \quad \lambda(\zeta^{\check{\alpha}}, t \leq 0) = \lambda_0, \quad (4.8)$$

where $\zeta^{\check{\alpha}} \in \Omega := [0, 1] \times [0, 1]$. The corresponding membrane position is expressed as

$$\mathbf{x}(\zeta^{\check{\alpha}}, t \leq 0) = x \mathbf{e}_x + y \mathbf{e}_y, \quad (4.9)$$

with

$$x(\zeta^{\check{\alpha}}, t \leq 0) = (\zeta^1 - 0.5) \ell \quad \text{and} \quad y(\zeta^{\check{\alpha}}, t \leq 0) = (\zeta^2 - 0.5) \ell. \quad (4.10)$$

At time $t = 0$, we seek to vertically displace the centre of the membrane patch, where $x = 0$ and $y = 0$. However, as shown in Appendix C, there is no unique way to specify the velocity at a single point given our use of B-spline basis functions, as they are not interpolatory. Instead, we vertically displace the portion of the membrane corresponding to the entire parametric element Ω_p^e containing the point $\zeta_p^{\check{\alpha}} := (0.5, 0.5)$ at the centre of the parametric domain:

$$\mathbf{v}(\zeta^{\check{\alpha}}, t \geq 0) = \mathbf{v}_p(t) \quad \text{for all} \quad \zeta^{\check{\alpha}} \in \Omega_p^e. \quad (4.11)$$

In (4.11), $\mathbf{v}_p(t)$ is a known function of time. It is given by $v_p \mathbf{e}_z$, for constant v_p , unless otherwise specified. Following the derivations in Appendix C, (4.11) is enforced by setting

all nodal velocity degrees of freedom associated with Ω_p^e to $\mathbf{v}_p(t)$. The resultant pull force $\mathcal{F}_p(t)$ is calculated as the sum of the corresponding components of the residual vector.

4.2.2. Non-dimensionalisation

In the scenario under consideration, there are five membrane parameters: the bending modulus k_b , equilibrium surface tension λ_0 , 2-D intramembrane viscosity ζ , patch length ℓ and speed of tube drawing v_p . We hope to include the hydrodynamics of the surrounding fluid, including membrane permeability (Alkadri & Mandadapu 2024), and the effects of cytoskeletal contacts (Shi *et al.* 2018) in a future effort – and better compare our results with experiments (Cuvelier *et al.* 2005; Brochard-Wyart *et al.* 2006). Since the five quantities dimensionally involve only mass, length and time, two dimensionless numbers once again determine the evolution of the system. The Föppl–von Kármán number Γ is given by (Sahu *et al.* 2020a)

$$\Gamma := \frac{\lambda_0 \ell^2}{k_b}, \quad (4.12)$$

and can be interpreted in two ways given the morphological changes that occur when pulling a tether. First, when the membrane is nearly planar and height deflections are small, gradients in membrane shape occur over the length scale ℓ ; Γ then quantifies the relative magnitude of tensile and bending forces. In our simulations, Γ^{-1} is small, and consequently a boundary layer of characteristic width $\sqrt{k_b/\lambda_0} \ll \ell$ develops at the point of application of the pull force (Powers *et al.* 2002). As the membrane is pulled further, a tether grows from the boundary layer region. The tether is close in shape to a cylinder, and the radius of a cylindrical membrane is well known to be (Zhong-can & Helfrich 1989)

$$r_c := \sqrt{\frac{k_b}{4\lambda_0}}. \quad (4.13)$$

Equation (4.13) can be obtained from energetic arguments alone, or equivalently, from a balance of bending and tensile forces. Note that in situations where there is a pressure jump across the membrane due to the surrounding fluid and $f = f \cdot \mathbf{n} \neq 0$, (4.13) and (4.17) are not valid. See the discussion in Chapter IX, § 1(a) of Sahu (2022). With (4.13), the Föppl–von Kármán number (4.12) is understood as the ratio

$$\Gamma = \left(\frac{\ell}{2r_c} \right)^2. \quad (4.14)$$

Equations (4.12)–(4.14) confirm that the Föppl–von Kármán number describes membrane energetics, as only lengths and the parameters k_b and λ_0 – which respectively have dimensions of energy and energy per area – are involved.

The second dimensionless quantity is the Scriven–Love number S_L , which compares the magnitude of viscous and bending forces in shaping the membrane (Sahu *et al.* 2020a). In this scenario, we define the Scriven–Love number as

$$S_L := \frac{r_c \zeta v_p}{k_b}. \quad (4.15)$$

We chose to include r_c – rather than ℓ – in (4.15) because we are primarily concerned with the behaviour of the tether, rather than the entire patch. Upon substituting (4.13) into (4.15), rearranging terms and recognising ζ/λ_0 is the fundamental time scale of lipid flows

Mesh motion	# DOFs	Wall clock time
LAG	17 401	271 mins
EUL	30 833	525 mins ^a
ALE-vb	34 282	636 mins

Table 4. Number of degrees of freedom (DOFs) and wall clock run time for the three different mesh motions, corresponding to the results presented in figure 4. All computations were carried out on a single node of the Perlmutter system at the National Energy Research Scientific Computing Center, with area element calculations distributed across 32 cores.

^aScaled to the total number of time steps, as the simulation failed.

(Sahu *et al.* 2020a), we find the Scriven–Love number can be equivalently expressed as

$$SL := \frac{v_p}{4r_c(\zeta/\lambda_0)^{-1}}, \quad (4.16)$$

i.e. a ratio of the speed at which the tether is pulled to the natural velocity scale of lipid flows in the tube (Sahu *et al.* 2020a). Thus, as SL tends to zero, we approach the quasi-static limit.

In our source code, when running tether pulling simulations, we choose $\zeta = 1.0$, $k_b = 1.0$ and $\lambda_0 = 0.25$. Our choice yields a tube radius $r_c = 1.0$ according to (4.13). The Föppl–von Kármán number Γ is altered by varying the patch size ℓ , while the Scriven–Love number SL is modified by changing the pull speed v_p .

4.2.3. The comparison of different mesh motions

We begin by pulling a tether in a scenario with Föppl–von Kármán number $\Gamma = 1024$ and Scriven–Love number $SL = 0.1$. Given these dimensionless numbers, $\ell = 64r_c$ and $v_p = 0.1k_b/(\zeta r_c)$ according to (4.14) and (4.15). Snapshots from tether pulling simulations with each mesh motion are shown in figure 4, with corresponding videos in the MembraneAleFem.jl package repository (Sahu 2024). A zoomed-in view of the late-time snapshots, with and without the underlying mesh, are shown in figure 5 to emphasise the advantages of the ALE scheme. In addition, figure 6 shows how the z component of the pull force, $\mathcal{F}_{p,z} := \mathcal{F}_p \cdot \mathbf{e}_z$, varies as a function of the vertical displacement z_p . We note that in the quasi-static limit ($SL \rightarrow 0$), the steady-state pull force of a perfect cylinder is given by Evans & Yeung (1994)

$$\mathcal{F}_{eq} := \frac{\pi k_b}{r_c} = 2\pi\sqrt{k_b\lambda_0}. \quad (4.17)$$

Note that some studies define $\kappa := k_b/2$ to be the bending modulus, in which case $\mathcal{F}_{eq} = 2\pi\sqrt{2\kappa\lambda_0}$. Since a tether pulled from a flat patch deviates slightly from a cylinder (Powers *et al.* 2002), (4.17) serves as an excellent approximation for $\mathcal{F}_{p,z}$ when $SL = 0$.

In what follows, we comment on the efficacy of the three mesh motions: Lagrangian (LAG), Eulerian (EUL) and ALE-viscous-bending (ALE-vb). Note that additional variables are used to solve for membrane behaviour in the latter two schemes: the fundamental variables are \mathbf{v} and λ (LAG); \mathbf{v} , \mathbf{v}^m and λ (EUL); and \mathbf{v} , \mathbf{v}^m , λ and p^m (ALE-vb). We present the number of degrees of freedom for each motion, as well as the wall clock run time, in table 4. When appropriate, our results are compared to those of prior theoretical and numerical developments.

Lagrangian scheme: we first consider results from simulations with a Lagrangian mesh motion, as shown in the left column of figures 4 and 5 (see supplementary movie 1

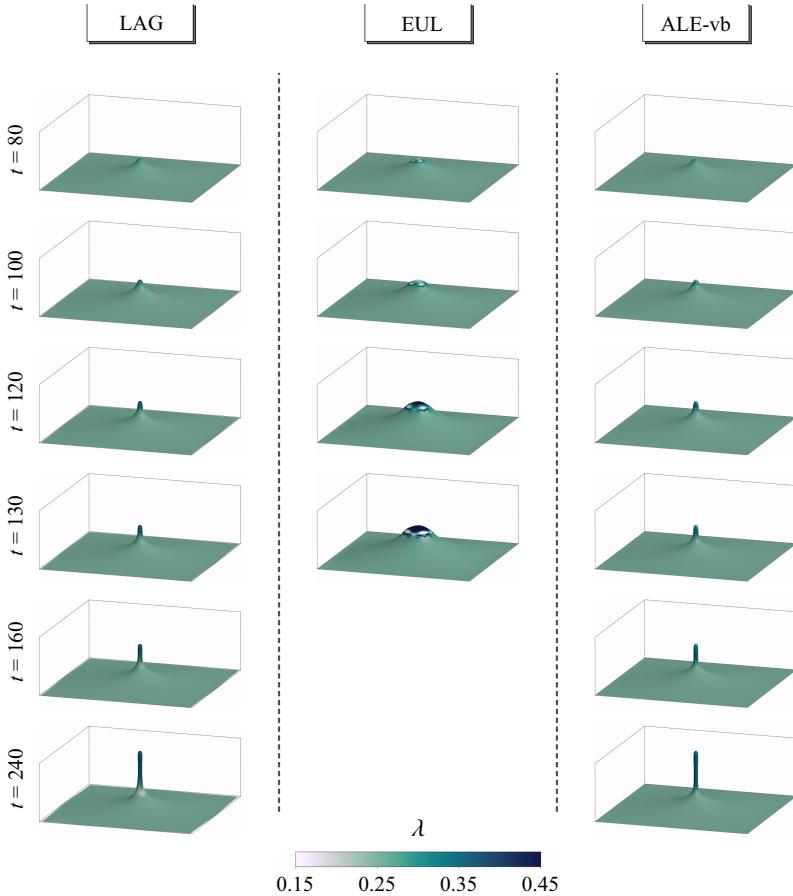


Figure 4. Snapshots from tether pulling simulations with a 65×65 mesh, in which three different mesh motions are employed. The colour bar indicates the surface tension, in units of k_b/r_c^2 , for all snapshots. Times are measured in units of $\zeta r_c^2/k_b$. (Left) Lagrangian simulations (LAG) successfully generate a tether. The morphological shape change from a tent to a tube occurs around time $t = 120$. Since the membrane is area incompressible, the patch boundary must be pulled inwards to accommodate the tether surface area. (Centre) The Eulerian mesh motion (EUL) is not able to form a tube. Rather, the tent morphology persists and bulges outward until the method fails around time $t = 143$. The material velocity degrees of freedom of the central element are constrained to move upwards; no such constraint is placed on the mesh velocity degrees of freedom. (Right) An ALE mesh motion that is viscous and resists bending (ALE-vb) successfully forms a tether. Both material and mesh velocities of the central element are constrained to move upwards. Since the mesh is area compressible, the patch boundary can be constrained to remain stationary as the tether develops. For all three mesh motions, $\Delta t = 0.5$, $SL = 0.1$ and $\Gamma = 1024$, for which $\ell = 64 r_c$ and $v_p = 0.1 k_b/(\zeta r_c)$ according to (4.14) and (4.15). See also supplementary movies 1–3.

available at <https://doi.org/10.1017/jfm.2025.10553>). A tent-like structure develops and grows until the vertical displacement $z_p := v_p t \approx 8r_c$, for which $t \approx 80 \zeta r_c^2/k_b$. Around this point, the linear relationship between $\mathcal{F}_{p,z}$ and z_p breaks down, as shown in figure 6. The membrane undergoes a morphological transition between $t \approx 80 \zeta r_c^2/k_b$ and $t \approx 120 \zeta r_c^2/k_b$, during which $\mathcal{F}_{p,z}$ reaches its largest value. After the shape transition, a tether continues to be drawn at an approximately constant force, which we refer to as the steady-state force \mathcal{F}_{ss} (see figure 6). However, a pronounced area of low surface tension develops

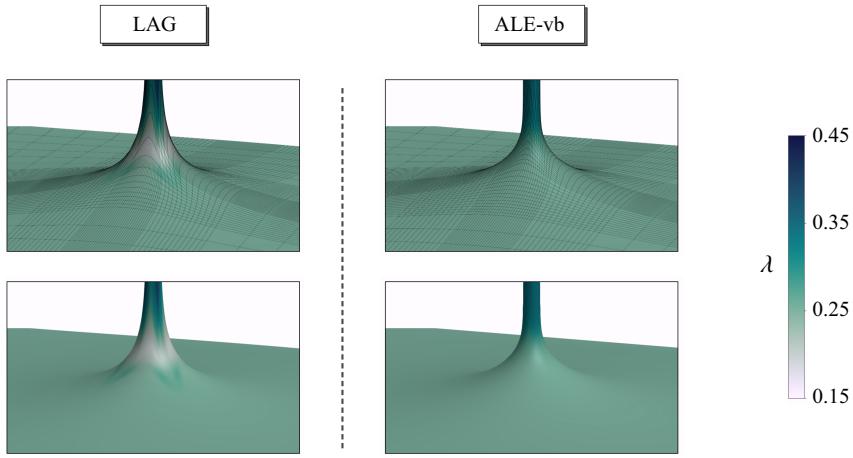


Figure 5. Zoomed-in views of the $t = 240 \zeta r_c^2 / k_b$ snapshots from figure 4, with the underlying mesh shown (top) and hidden (bottom). In the Lagrangian simulations (left), the mesh is drawn into the tube along with the lipids due to the areal incompressibility of the membrane. Mesh elements close to the diagonal of the square patch, where $y = \pm x$, become highly distorted – which leads to an artificially low surface tension in the transition region between the tether and surrounding membrane. Numerical artefacts are visible in the striation of the tension. The ALE-viscous-bending result (right), in contrast, shows a less distorted mesh because the choice of mesh constitution resists both shear and dilatation. No surface tension artefacts are visible, and a smooth tension gradient is observed from the flat patch into the tether.

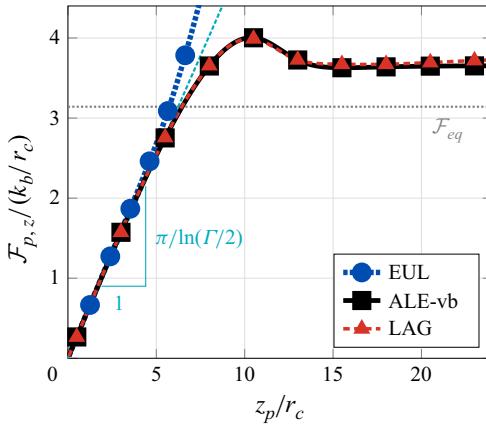


Figure 6. The z component of the pull force ($\mathcal{F}_{p,z}$) as a function of the z displacement. The dashed cyan line is the result from the linear theory, as presented in (4.20). The Eulerian simulation (blue circles, dotted line) is unable to form a tether, and is unphysical. The Lagrangian (red triangles, dashed line) and ALE (black squares, solid line) simulations capture the tent-to-tube transition, after which the Lagrangian steady-state pull force \mathcal{F}_{ss} is slightly larger. Both overshoot the equilibrium pull force \mathcal{F}_{eq} (4.17) due to dynamical effects from tether pulling (see § 4.2.4). Numerical parameters are those specified in figure 4.

in the region where the tether meets the flat patch (see figures 4 and 5). We comment on the unphysical nature of this development following a presentation of our ALE results.

The majority of our findings from Lagrangian tether pulling simulations, as well as the Lagrangian implementation itself, are not new. Powers *et al.* (2002) provided a detailed account of the axisymmetric, quasi-static membrane shape in response to vertical displacements, and along with Derényi *et al.* (2002) numerically calculated the

force versus displacement curve in the quasi-static limit. More recently, rate effects were included in axisymmetric simulations that focused primarily on the slip between two monolayer leaflets (Rahimi & Arroyo 2012) – an effect not considered in the present work. General non-axisymmetric Lagrangian formulations were developed since then. The implementation in Rodrigues *et al.* (2015) included rate effects when pulling a tether, but the force was not provided as a function of displacement and it is unclear if the membrane underwent the morphological transition from a tent to a tube in simulations. In contrast, Sauer *et al.* (2017) pulled a tether from the centre of an initially axisymmetric mesh and illustrated the tent-to-tube transition. While this implementation seems to be capable of capturing rate effects, the reported tether pulling results employed a numerical stabilisation scheme that did not capture the coupling between in-plane lipid flows and out-of-plane forces. Thus, to the best of our knowledge, we present the first non-axisymmetric Lagrangian simulation capturing the dynamics of a tether pulled from a membrane patch – though we do not consider the results to be novel, as Rodrigues *et al.* (2015) and Sauer *et al.* (2017) may have been able to do the same.

Eulerian scheme: let us next examine results from an Eulerian mesh motion. As shown in supplementary movie 2 and the centre column of figure 4, the numerical implementation fails to form a tether. In addition, unphysical gradients develop in the surface tension, and the code fails at time $t = 143 \zeta r_c^2 / k_b$. In what follows, we discuss why such a failure is to be expected, as a tether cannot form when the mesh motion is Eulerian.

We begin by introducing J and J^m as the relative area dilatations of the membrane and mesh, respectively. These dilatations are related to the corresponding flow fields according to (Sahu 2022, Chapter V, § 1(c))

$$\frac{\dot{J}}{J} = v_{;\check{\alpha}} - 2vH \quad \text{and} \quad \frac{\dot{J}^m}{J^m} = (v_m^{\check{\alpha}})_{;\check{\alpha}} - 2v^m H. \quad (4.18)$$

Since the membrane is incompressible, $\dot{J}/J = 0$ and $v_{;\check{\alpha}}^{\check{\alpha}} = 2vH$. To understand how the mesh dilates or compresses, we recognise that $v^m = v$ and $v_m^{\check{\alpha}} = 0$ according to (2.32). We thus find that

$$\frac{\dot{J}^m}{J^m} = -2vH \neq 0. \quad (4.19)$$

At this point, we make three observations regarding the geometry and dynamics of the initial tent formation: (i) $v \geq 0$ everywhere, (ii) $H < 0$ in a central region where the surface is concave down, and (iii) $H \geq 0$ elsewhere. Accordingly, $\dot{J}^m/J^m > 0$ in the centre of the tent, and the mesh continuously dilates. In this manner, the mesh expands laterally and under-resolves the region where the morphological transition would occur.

Mesh dilatation at the patch centre, where $\zeta^1 = 0.5$ and $\zeta^2 = 0.5$ (see (4.10)), is quantified in figure 7. Here, the relative area change of the mesh J^m is plotted as a function of the membrane displacement z_p . The dilatation by two orders of magnitude suggests an Eulerian simulation with a similar increase in the number of elements. Such a patch, even after the large dilatation, may be sufficiently resolved to undergo the morphological transition to a tube. However, even if a tether was formed, it likely could not be extended with an Eulerian mesh motion: lipid flows are primarily in-plane during tether elongation, so the Eulerian mesh would remain stationary and the tether would be poorly resolved. We thus believe the inability to pull a tether is a general failure of Eulerian methods, including those implemented previously (Reuther *et al.* 2020; Sahu *et al.* 2020b). Moreover, since the ALE implementation of Torres-Sánchez *et al.* (2019) employed a mesh motion that was close to Eulerian, we are unsure if it would be able to successfully pull a tether.

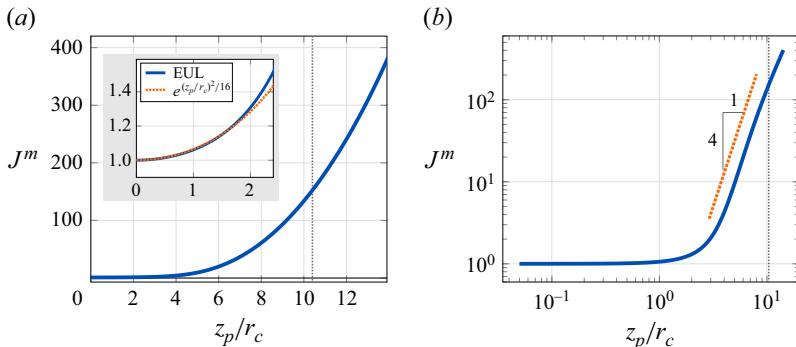


Figure 7. Mesh dilatation of the Eulerian simulation shown in figure 4. (a) The relative area change at the patch centre, J^m , is plotted as a function of the displacement z_p . The dashed vertical line is the displacement at which a tent is expected to transition to a tether, according to Lagrangian and ALE data in figure 6. (Inset) At small displacements, simulation data agrees with an approximate solution to (4.19). Here $v = \dot{z}_p$; if the mean curvature is linear in z_p then $H \sim -z_p/r_c^2$ and integrating (4.19) yields $\ln J^m \sim (z_p/r_c)^2$. The number 16 in the analytical expression is a fitting parameter. (b) The same data in a log–log plot suggests a power-law growth of the dilatation at large displacements, prior to the expected morphological transition (dotted vertical line) – though the data does not span even a single decade.

ALE scheme: we close by discussing the results of our ALE mesh motion. As shown in figures 4–6 and supplementary movie 3, a tether was successfully pulled with the ALE-vb scheme, and the surface tension in the region where the tether meets the flat patch was approximately constant. This latter point is quantified by considering the range of surface tension values at time $t = 240 \zeta r_c^2/k_b$. In the ALE simulation, the lowest (dimensionless) value of the surface tension is 0.249 – approximately equal to the magnitude of the in-plane force per length $|\bar{\mathbf{F}}_\parallel| = \lambda_0 = 0.25$ maintained on the boundary. Larger tension values, in this case up to 0.345, arise in the tether to draw in lipids during the dynamic process of pulling. In contrast, the minimum value of the tension in Lagrangian simulations is 0.191: well below $|\bar{\mathbf{F}}_\parallel|$ and also less than the larger values (up to 0.446) attained on the tether. Since lipids flow from regions of low to high tension, there does not seem to be a smooth flow of lipids from the flat patch into the tether. We thus find the ALE-viscous-bending mesh motion to be superior to its Lagrangian counterpart, and report only ALE-vb results for the remainder of our tether pulling analysis.

It is important to note that while the ALE-vb scheme successfully pulls a tether, the purely viscous ALE mesh motion – employed in § 4.1 – does not: as the centre of the patch is translated upwards, the tether tapers to a point. We believe an angular shape arises in the ALE-v simulation because of incompatible boundary conditions for the material and mesh. More specifically, since all nodes over a single element are translated upwards, a zero-slope condition is implicitly prescribed at the element boundary. Both the membrane and ALE-vb mesh can sustain such a requirement due to their inherent bending stiffness. In contrast, the ALE-v mesh has no bending rigidity and so cannot maintain zero slope on the element boundary. The tether resulting from the ALE-v scheme is consequently unphysical.

4.2.4. The geometry and dynamics of tether pulling

In comparing different mesh motions in § 4.2.3, all simulations were carried out at a single choice of Γ and S_L . We now investigate the effects of altering the patch size relative to the tether radius, as well as changing the speed of tether pulling – which respectively modify the Föppl–von Kármán and Scriven–Love numbers. We confirm that Γ dictates the initial

slope of the force versus displacement curve, as previously observed (Derényi *et al.* 2002; Powers *et al.* 2002; Sauer *et al.* 2017), while SL captures the overshoot of the tether pull force relative to the equilibrium (or quasi-static) result.

We first investigate the dependence of the pull force on the patch geometry. Figure 8(a) plots $\mathcal{F}_{p,z}$ as a function of z_p for different values of Γ , at fixed $SL = 0.1$. We observe that Γ alters the initial slope of the force versus displacement curve, but does not affect the steady-state pull force after the tether is formed – the latter of which is expected to be independent of the patch size (see (4.17) and Powers *et al.* 2002; Derényi *et al.* 2002). To calculate the initial dependence of $\mathcal{F}_{p,z}$ on z_p , we analyse the tent-like membrane shape when deformations are small. In this limit, the shape equation (2.19) is decoupled from in-plane lipid flows, and identical to its quasi-static counterpart (Sahu *et al.* 2020a). We thus take the small-deformation, quasi-static membrane tent result from Powers *et al.* (2002) and calculate the pull force to be given by (cf. (4.17))

$$\frac{\mathcal{F}_{p,z}}{k_b/r_c} = \frac{\pi}{\ln(\Gamma/2)} \cdot \frac{z_p}{r_c}, \quad \text{or equivalently} \quad \frac{\mathcal{F}_{p,z}}{\mathcal{F}_{eq}} = \frac{1}{\ln(\Gamma/2)} \cdot \frac{z_p}{r_c}, \quad (4.20)$$

which is shown as the dashed cyan line in figure 6. Our calculation of (4.20) is provided in Appendix D, and the collapse of force versus displacement curves is shown in figure 8(b).

The dependence of the pull force on the speed of tether pulling is investigated next. Results from simulations with variable SL and fixed $\Gamma = 1024$ are plotted in figure 8(c). The observed increase in pull force with increasing Scriven–Love number can be justified as follows. During tether pulling, lipids in the surrounding region flow inwards towards the tether. A mass balance, with the assumption of axisymmetry, indicates that the flow speed is approximately $v_p r_c / r$, where r is the distance from the z axis. Since the in-plane velocity grows as we approach the tether from its surroundings, a surface tension gradient is required to sustain the flow of lipids. As the tether is pulled more quickly, larger velocities and, thus, larger tension gradients ensue. A greater surface tension in the tether results, and leads to the larger pull force observed in simulations.

At present, we are unable to determine a general functional form for the long-time, steady-state pull force as a function of SL due to the high degree of nonlinearity in the governing equations. The main difficulty is that the surface tension and membrane geometry enter both the in-plane and shape equations. Moreover, the viscous–curvature coupling forces – which arise due to the flow of lipids – lead to $O(SL)$ changes of the membrane shape in the tent-like region. Since we are unable to approximate how the steady-state pull force depends on the Scriven–Love number, we choose not to collapse the data contained in figure 8(c). Instead, we plot the steady-state pull force \mathcal{F}_{ss} as a function of the Scriven–Love number in figure 8(d). We expect $\mathcal{F}_{ss} \approx \mathcal{F}_{eq}$ when $SL \rightarrow 0$. In the limit where $SL \ll 1$, we approximate dynamical effects by calculating the change in surface tension under the assumption of no membrane shape changes. We find that

$$\mathcal{F}_{ss}(SL) = (1 + 2SL) \mathcal{F}_{eq} \quad \text{for } SL \ll 1, \quad (4.21)$$

which is shown as the dotted grey line in figure 8(d). Evidently, (4.21) is a reasonable approximation when $SL < 0.1$.

4.2.5. The lateral translation of a membrane tether

Once a tube is pulled from a patch of membrane, a lateral force applied at the end of the tether causes it to translate relative to the surrounding region. Lipids quickly flow and readjust to accommodate the translating tether due to the in-plane fluidity of the membrane. While tether translation via a lateral force is observed in *in vitro* studies (Datar *et al.* 2015; Gomis Perez *et al.* 2022; Shi, Innes-Gold & Cohen 2022), the physics of tether translation

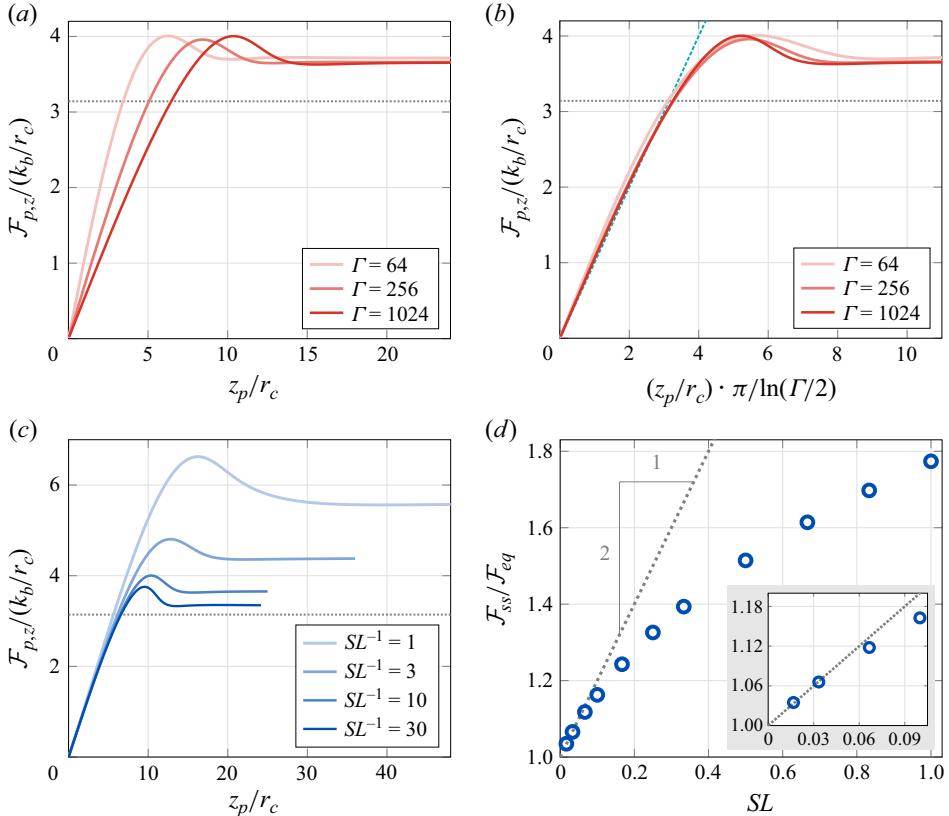


Figure 8. The z component of the pull force ($\mathcal{F}_{p,z}$) as a function of z_p for different values of Γ and SL . (a) When $SL = 0.1$ and Γ is varied, the initial slope of the force vs displacement curve is altered according to the linear theory (see (4.20)). (b) By scaling the z displacement appropriately, the data collapses – with the steady-state pull force independent of Γ after tether formation. The cyan line has slope one. (c) When $\Gamma = 1024$ and SL is varied, the initial slope of the force vs displacement curve is unchanged. The tent-to-tube transition occurs at larger displacements, and the long-time pull force increases with SL . (d) Long-time pull force, \mathcal{F}_{ss} , as a function of SL for $\Gamma = 1024$. The dotted grey line is the linear prediction from (4.21), which agrees with simulation results when $SL \ll 1$ – as highlighted by the zoomed-in inset. The nonlinear dependence of \mathcal{F}_{ss} on SL arises from the coupling between in-plane flows and out-of-plane shape deformations.

remains poorly understood. A theoretical description of tether translation is difficult because the system is no longer axisymmetric, and the greatly simplified axisymmetric equations (Derényi *et al.* 2002; Powers *et al.* 2002; Agrawal & Steigmann 2009, 2011; Omar *et al.* 2020) no longer apply. Tether translation also cannot be captured with general, non-axisymmetric Lagrangian numerical methods – as laterally translating the tether induces a rigid body translation of the entire patch (see Appendix E). Arbitrary Lagrangian–Eulerian finite element methods are thus required to simulate tether translation.

In figure 9 we present snapshots from a simulation of tether translation, which employed the ALE-vb mesh motion. Starting with the $t = 240 \zeta r_c^2/k_b$ configuration shown in figures 4 and 5, we prescribe a lateral velocity in the x direction given by

$$\mathbf{v}_p(t) = \begin{cases} +v_p \mathbf{e}_x & \text{for } 240 < t \cdot k_b / (\zeta r_c^2) < 300, \\ -v_p \mathbf{e}_x & \text{for } 300 < t \cdot k_b / (\zeta r_c^2) < 420, \end{cases} \quad (4.22)$$

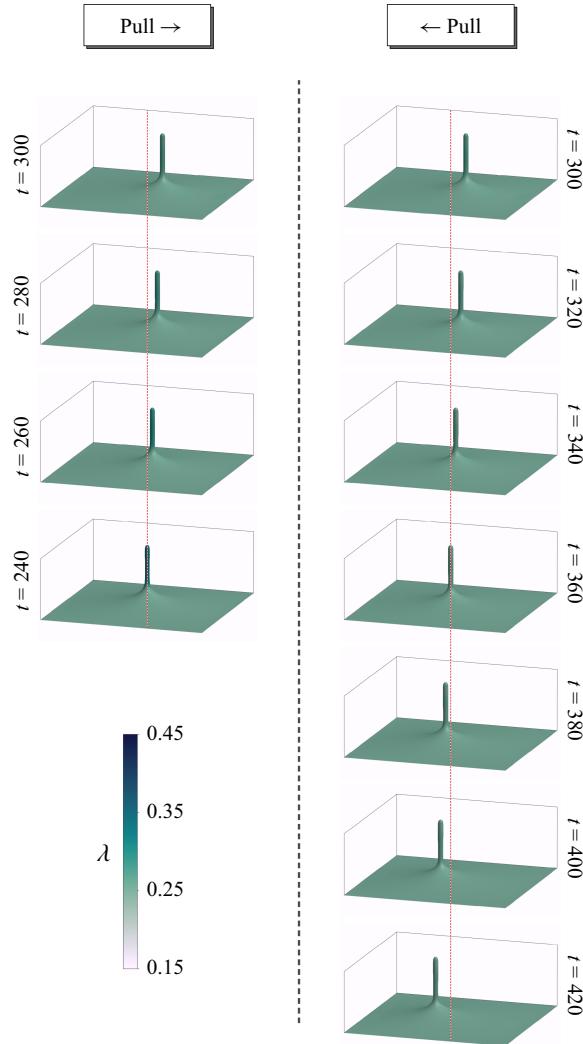


Figure 9. Tether translation in the $+e_x$ (left column) and then $-e_x$ (right column) directions. Times are measured in units of $\zeta r_c^2/k_b$, the pull velocity is specified in (4.22), and all other parameters are identical to those in figure 4. The vertical red lines are a visual aid to highlight the lateral translation of the tether. For a video of the simulation results, see supplementary movie 4.

where in both cases $v_p = 0.1k_b/(\zeta r_c)$ to be consistent with our choice $SL = 0.1$. In this simulation, $\Gamma = 1024$ is unchanged. We observe that the surface tension does not appreciably change, as the tether is translated slowly relative to the fundamental time scale of lipid rearrangements. Our results demonstrate that ALE methods can be used in scenarios where Lagrangian methods fail, and set the stage for future investigations of the forces, geometry and dynamics of tether translation. A video of the laterally pulled tether simulation is provided in the software repository (Sahu 2024).

5. Conclusions and future work

In the present work we (i) developed a robust ALE numerical method for lipid membranes and (ii) applied the method to a scenario where established Lagrangian and Eulerian

schemes fail. In our development, the mesh is treated as a material that is independent from the membrane – with the mesh equations of motion and corresponding boundary conditions arbitrarily prescribed by the user. Mesh and material surfaces are constrained to overlap with a Lagrange multiplier field, which enters the mesh dynamics as a force per area in the normal direction. By choosing for the mesh to resist shear and dilatation (through the mesh viscosity ζ^m) and bending (through the mesh bending moduli k_b^m and k_g^m), we successfully pulled a tether from a flat patch and then translated it laterally across the membrane surface. In contrast, Lagrangian and Eulerian simulations are respectively unable to translate and pull a tether. Our results thus mark the first numerical demonstration of lateral tether translation. We also analysed the dynamics of tether pulling by determining how the pull force increases monotonically with increasing pull speed. Our findings were presented in terms of the Föppl–von Kármán number Γ and Scriven–Love number S_L , which define the tether pulling scenario.

We close by highlighting that our numerical implementation (Sahu 2024) is structured such that one can easily choose different constitutive relations and boundary conditions for the mesh motion. In this manner, the mesh itself can resist shear, dilatation or their rates of change – irrespective of the material behaviour. Only purely viscous and viscous-with-bending mesh motions were considered in the present work; we aim to investigate different choices of mesh motion in a future study. As a consequence of the modular structure of the numerical implementation, it is straightforward to adapt the code to simulate 2-D materials with different constitution. We hope to support the open-source community in doing so. We also intend to extend our method to describe additional phenomena governing biological membranes – including the coupling between lipid flows and the hydrodynamics of the surrounding fluid (Narsimhan *et al.* 2015; Gross & Atzberger 2018; Faizi *et al.* 2024), membrane permeability (Alkadri & Mandadapu 2024), the effects of embedded particles (Stone 2010; Stone & Masoud 2015; Sabass & Stone 2016) and in-plane diffusion and phase transitions in multicomponent membrane systems (Subramaniam *et al.* 2013; Yu & Košmrlj 2023; Venkatesh & Narsimhan 2024; Venkatesh *et al.* 2025).

Supplementary movies. Supplementary movies are available at <https://doi.org/10.1017/jfm.2025.10553>.

Acknowledgements. We are indebted to Professor Kranthi Mandadapu for many scientific discussions on membrane dynamics and finite element methods within the setting of differential geometry. We are also grateful to Professor Panos Papadopoulos and Dr. Yannick Omar for bringing important details of finite element analysis to our attention, including the method of Dohrmann & Bochev (2004) and numerical differentiation of the residual vector (Tanaka *et al.* 2014). It is a pleasure to thank Dr. Joël Tchoufag regarding conversations on fluid mechanics and membrane dynamics in different geometries. Simulations were carried out on the Perlmutter high-performance computing system at the National Energy Research Scientific Computing Center. Our open-source package is written in the Julia programming language (Bezanson *et al.* 2017). Figures and videos of the simulation results were generated with the open-source Julia package Makie.jl (Danisch & Krumbiegel 2021). This work was partially supported by the Welch Foundation via Grant No. F-2208-20240404.

Declaration of interests. The authors report no conflict of interest.

Appendix A. The finite element implementation

The main novelty of the present work is the treatment of the mesh as an independent material with its own constitution, as discussed in § 2.7. However, none of the finite element techniques used here are new. All fundamental unknowns, their arbitrary variations and the surface position itself are discretised with the same basis functions – as is standard in isoparametric finite element methods. Over any element $\Omega^e \in \mathcal{T}_h$, we have

$$\mathbf{x}_h(\zeta^{\check{\alpha}}, t) = \sum_{k=1}^{nen} N_k^e(\zeta^{\check{\alpha}}) \mathbf{x}_k^e(t) = [\mathbf{N}^e(\zeta^{\check{\alpha}})] [\mathbf{x}^e(t)] \quad \text{for all } \zeta^{\check{\alpha}} \in \Omega^e. \quad (\text{A1})$$

In (A1) the matrix $[N^e(\zeta^{\check{\alpha}})]$ contains the non-zero elemental basis functions (see § 3.1) and the column vector $[\mathbf{x}^e(t)]$ collects the corresponding local degrees of freedom (or nodal positions) in the usual manner of finite element analysis:

$$[N^e(\zeta^{\check{\alpha}})] := \begin{bmatrix} N_1^e(\zeta^{\check{\alpha}}) [\mathbf{1}] & N_2^e(\zeta^{\check{\alpha}}) [\mathbf{1}] \dots N_{nen}^e(\zeta^{\check{\alpha}}) [\mathbf{1}] \end{bmatrix}. \quad (\text{A2})$$

Here $[\mathbf{1}]$ is the 3×3 identity matrix in Cartesian coordinates and

$$[\mathbf{x}^e(t)] := \begin{bmatrix} \mathbf{x}_1^e(t) \\ \vdots \\ \mathbf{x}_{nen}^e(t) \end{bmatrix}. \quad (\text{A3})$$

The velocity and mesh velocity are similarly decomposed as

$$\mathbf{v}_h(\zeta^{\check{\alpha}}, t) = [N^e(\zeta^{\check{\alpha}})] [\mathbf{v}^e(t)] \quad (\text{A4})$$

and

$$\mathbf{v}_h^m(\zeta^{\check{\alpha}}, t) = [N^e(\zeta^{\check{\alpha}})] [\mathbf{v}^{m,e}(t)] \quad (\text{A5})$$

for all $\zeta^{\check{\alpha}} \in \Omega^e$. Since the membrane tension and mesh pressure are scalar quantities, they are respectively discretised as

$$\lambda_h(\zeta^{\check{\alpha}}, t) = [N^e(\zeta^{\check{\alpha}})] [\lambda^e(t)] \quad (\text{A6})$$

and

$$p_h^m(\zeta^{\check{\alpha}}, t) = [N^e(\zeta^{\check{\alpha}})] [\mathbf{p}^{m,e}(t)] \quad (\text{A7})$$

for all $\zeta^{\check{\alpha}} \in \Omega^e$, where $[N^e(\zeta^{\check{\alpha}})]$ is the row vector of non-zero elemental basis functions given by (cf. (A2))

$$[N^e(\zeta^{\check{\alpha}})] := \begin{bmatrix} N_1^e(\zeta^{\check{\alpha}}) & N_2^e(\zeta^{\check{\alpha}}) \dots N_{nen}^e(\zeta^{\check{\alpha}}) \end{bmatrix}. \quad (\text{A8})$$

Assuming a known solution at time t , the temporal evolution of the membrane is obtained via the backward Euler method. In particular, the membrane surface is updated according to (cf. (2.28), (2.29))

$$\mathbf{x}(\zeta^{\check{\alpha}}, t + \Delta t) = \mathbf{x}(\zeta^{\check{\alpha}}, t) + \Delta t \mathbf{v}^m(\zeta^{\check{\alpha}}, t + \Delta t), \quad (\text{A9})$$

or equivalently,

$$[\mathbf{x}^e(t + \Delta t)] = [\mathbf{x}^e(t)] + \Delta t [\mathbf{v}^{m,e}(t + \Delta t)]. \quad (\text{A10})$$

As discussed in § 3.2, the mesh velocity and all other unknowns satisfy the residual vector equation

$$[\mathbf{r}([\mathbf{u}(t + \Delta t)])] = [\mathbf{0}], \quad (\text{A11})$$

which is solved via the Newton–Raphson method – in which $[\mathbf{u}(t)]$ is used as the initial guess. We thus close our discussion of the finite element implementation by presenting the contributions to the residual vector; all details can be found in the software documentation (Sahu 2024).

A.1. The contributions to the residual vector

Our finite element implementation allows one to choose whether the mesh motion is Lagrangian, Eulerian or ALE – and, in the latter case, whether the mesh dynamics is

purely viscous or viscous with a bending resistance. The direct Galerkin expressions in (2.31), (2.34), (2.48) and (2.57) dictate the residual vector for each mesh motion. In what follows, we present the local residual vector contributions – corresponding to a single finite element $\Omega^e \in \mathcal{T}_h$ – for each fundamental unknown.

A.1.1. The surface tension contribution

As detailed in Sahu *et al.* (2020b), the residual vector corresponding to (2.12) is given by

$$[\mathbf{r}_\lambda^e] = \int_{\Omega^e} [\mathbf{N}^e]^T (\mathbf{a}^\alpha \cdot \mathbf{v}_{,\alpha}) J_\Omega d\Omega - \frac{\alpha^{DB}}{\zeta} \left\{ \int_{\Omega^e} [\mathbf{N}^e]^T \lambda d\Omega - [\mathbf{G}^e]^T [\mathbf{H}^e]^{-1} [\mathbf{G}^e] [\lambda^e] \right\}, \quad (\text{A12})$$

where all quantities in curly braces arise from the method of Dohrmann & Bochev (2004). The matrices $[\mathbf{G}^e]$ and $[\mathbf{H}^e]$ in (A12) are constructed from the basis functions $\check{N}_k^e(\zeta^\alpha)$ to the space \check{L} onto which surface tensions are projected. Since functions in \check{L} are linear and form a plane over Ω^e (see (3.7)), they can be expressed as $a + b\zeta^1 + c\zeta^2 = 0$ for some constants $a, b, c \in \mathbb{R}$. In our numerical implementation, the planar basis functions over a single element are chosen to be $\check{N}_1^e(\zeta^\alpha) = 1$, $\check{N}_2^e(\zeta^\alpha) = \xi(\zeta^1)$ and $\check{N}_3^e(\zeta^\alpha) = \eta(\zeta^2)$, where ξ and η parametrise the reference square $[-1, 1] \times [-1, 1]$ onto which Ω^e is mapped. Accordingly, over any element $\Omega^e \in \mathcal{T}_h$ we define the row vector

$$[\check{\mathbf{N}}^e(\zeta^\alpha)] := [\check{N}_1^e(\zeta^\alpha) \quad \check{N}_2^e(\zeta^\alpha) \quad \check{N}_3^e(\zeta^\alpha)], \quad (\text{A13})$$

with which the matrices $[\mathbf{G}^e]$ and $[\mathbf{H}^e]$ in (A12) are expressed as

$$[\mathbf{G}^e] := \int_{\Omega^e} [\check{\mathbf{N}}^e]^T [\mathbf{N}^e] d\Omega \quad \text{and} \quad [\mathbf{H}^e] := \int_{\Omega^e} [\check{\mathbf{N}}^e]^T [\check{\mathbf{N}}^e] d\Omega. \quad (\text{A14})$$

A.1.2. The material velocity contribution

It is straightforward to determine the contributions to the residual vector from (2.27). We take advantage of the symmetry of $\sigma^{\alpha\beta}$ and $M^{\alpha\beta}$ to obtain

$$\begin{aligned} [\mathbf{r}_v^e] &:= \int_{\Omega^e} [\mathbf{N}^e]_{,\alpha}^T \mathbf{a}_\beta \sigma^{\alpha\beta} J_\Omega d\Omega + \int_{\Omega^e} [\mathbf{N}^e]_{;\alpha\beta}^T \mathbf{n} M^{\alpha\beta} J_\Omega d\Omega - \int_{\Omega^e} [\mathbf{N}^e]^T \mathbf{f} J_\Omega d\Omega \\ &\quad - \sum_{j=1}^3 \int_{\partial\Omega^e \cap \Gamma_F^j} [\mathbf{N}^e]^T \bar{F}_j J_\Gamma d\Gamma - \int_{\partial\Omega^e \cap \Gamma_M} [\mathbf{N}^e]_{,\alpha}^T \mathbf{v}^\alpha \bar{M} J_\Gamma d\Gamma. \end{aligned} \quad (\text{A15})$$

In (A15), it is understood that the x , y and z components of the force terms are written to appropriate entries of the residual vector.

A.1.3. The Eulerian mesh velocity contribution

For the choice of an Eulerian mesh motion, the residual vector contribution corresponding to (2.33) is given by

$$[\mathbf{r}_m^{E,e}] := \alpha_m^E \int_{\Omega^e} [\mathbf{N}^e]^T (\mathbf{v}^m - (\mathbf{n} \otimes \mathbf{n}) \mathbf{v}) J_\Omega d\Omega \quad (\text{A16})$$

A.1.4. The ALE mesh pressure contribution

When an ALE mesh motion is employed, the mesh pressure ensures the kinematic constraint (2.6) is satisfied. With the Dohrmann–Bochev method applied once again, the

residual vector resulting from (2.39) is given by (cf. (A12))

$$\begin{aligned} [\mathbf{r}_p^e] = & - \int_{\Omega^e} [\mathbf{N}^e]^T (\mathbf{n} \cdot (\mathbf{v}^m - \mathbf{v})) J_\Omega d\Omega \\ & - \frac{\alpha^{DB}}{\zeta \ell^2} \left\{ \int_{\Omega^e} [\mathbf{N}^e]^T \mathbf{p}^m d\Omega - [\mathbf{G}^e]^T [\mathbf{H}^e]^{-1} [\mathbf{G}^e] [\mathbf{p}^{m,e}] \right\}. \end{aligned} \quad (\text{A17})$$

A.1.5. The ALE mesh velocity contribution

When an ALE mesh motion is employed, the residual vector contribution from the mesh velocity looks similar to that from the material velocity (cf. (A15)):

$$[\mathbf{r}_m^{A,e}] := \int_{\Omega^e} [\mathbf{N}^e]_{,\check{\alpha}}^T \mathbf{a}_{\check{\beta}} \sigma_m^{\check{\alpha}\check{\beta}} J_\Omega d\Omega + \int_{\Omega^e} [\mathbf{N}^e]_{,\check{\alpha}\check{\beta}}^T \mathbf{n} \mathbf{M}_m^{\check{\alpha}\check{\beta}} J_\Omega d\Omega - \int_{\Omega^e} [\mathbf{N}^e]^T \mathbf{p}^m \mathbf{n} J_\Omega d\Omega \quad (\text{A18})$$

In (A18), $\sigma_{m,v}^{\check{\alpha}\check{\beta}}$ and $M_{m,v}^{\check{\alpha}\check{\beta}}$ (respectively $\sigma_{m,vb}^{\check{\alpha}\check{\beta}}$ and $M_{m,vb}^{\check{\alpha}\check{\beta}}$) are substituted when the motion is ALE-viscous (respectively ALE-viscous-bending).

Appendix B. The static portion of a cylinder

Here we consider a static membrane patch that is a portion of a cylinder – for which

$$\mathbf{x}(\theta, z) = r_c \mathbf{e}_r(\theta) + z \mathbf{e}_z. \quad (\text{B1})$$

In (B1), θ and z are the canonical cylindrical coordinates and r_c is the cylinder radius. With our differential geometric formulation, we arbitrarily choose to parametrise the surface as

$$\zeta^1 := \theta \in [0, \alpha] \quad \text{and} \quad \zeta^2 := \frac{z}{\ell} \in [0, 1], \quad (\text{B2})$$

such that

$$\mathbf{x}(\zeta^{\check{\alpha}}) = r_c \mathbf{e}_r(\zeta^1) + \ell \zeta^2 \mathbf{e}_z. \quad (\text{B3})$$

It is then straightforward to determine (Sahu 2022, Chapter IX, § 1)

$$\begin{aligned} \mathbf{a}_1 &= r_c \mathbf{e}_\theta(\zeta^1), & \mathbf{a}_2 &= \ell \mathbf{e}_z, & \mathbf{n} &= \mathbf{e}_r, & a_{\check{\alpha}\check{\beta}} &= \text{diag}(r_c^2, \ell^2), \\ a^{\check{\alpha}\check{\beta}} &= \text{diag}(r_c^{-2}, \ell^{-2}), & b_{\check{\alpha}\check{\beta}} &= \text{diag}(-r_c, 0), & b^{\check{\alpha}\check{\beta}} &= \text{diag}(-r_c^{-3}, 0), \end{aligned} \quad (\text{B4})$$

for which the mean and Gaussian curvatures are respectively given by

$$H = \frac{-1}{2r_c} \quad \text{and} \quad K = 0. \quad (\text{B5})$$

The couple-stress components $M^{\check{\alpha}\check{\beta}}$ and couple-free in-plane stress components $\sigma^{\check{\alpha}\check{\beta}}$ are calculated via (2.15) and (2.16) as

$$M^{\check{\alpha}\check{\beta}} = \begin{pmatrix} -k_b & 0 \\ \frac{2r_c^3}{2r_c} & \frac{-k_b}{2r_c\ell^2} - \frac{k_g}{r_c\ell^2} \end{pmatrix} \quad (\text{B6})$$

and

$$\sigma^{\check{\alpha}\check{\beta}} = \begin{pmatrix} \frac{\lambda_0}{r_c^2} - \frac{3k_b}{4r_c^4} & 0 \\ 0 & \frac{\lambda_0}{\ell^2} + \frac{k_b}{4r_c^2\ell^2} \end{pmatrix}, \quad (\text{B7})$$

where, for a static patch, there are no viscous stresses – for which $\pi^{\check{\alpha}\check{\beta}} = 0$. Here λ_0 denotes the constant surface tension, which is set by the balance of forces in the out-of-plane direction. With some additional calculations, we find the stress vectors $\mathbf{T}^{\check{\alpha}}$ to be given by

$$\mathbf{T}^{\check{1}} = \left(\frac{\lambda_0}{r_c} - \frac{k_b}{4r_c^3} \right) \mathbf{e}_\theta \quad \text{and} \quad \mathbf{T}^{\check{2}} = \left(\frac{\lambda_0}{\ell} + \frac{k_b}{4r_c^2\ell} \right) \mathbf{e}_r. \quad (\text{B8})$$

We now separately consider the top and bottom edges, where $\zeta^{\check{2}} = 1$ is fixed, and the left and right edges, where $\zeta^{\check{1}} = 0$ is fixed.

The top and bottom surfaces: at the top ($\zeta^{\check{2}} = 1$) and bottom ($\zeta^{\check{2}} = 0$) edges, $\mathbf{v} = \pm \mathbf{e}_z$, $v_{\check{1}} = 0$, $v_{\check{2}} = \pm \ell$, $\boldsymbol{\tau} = \mp \mathbf{e}_\theta$, $\tau_{\check{1}} = \mp r_c$ and $\tau_{\check{2}} = 0$. We thus determine (see § 2.3.1)

$$M = M^{\check{\alpha}\check{\beta}} v_{\check{\alpha}} v_{\check{\beta}} = M^{\check{2}\check{2}} v_{\check{2}} v_{\check{2}} = -\frac{k_b}{2r_c} - \frac{k_g}{r_c} \quad (\text{B9})$$

and

$$\mathbf{F} = \mathbf{T}^{\check{\alpha}} v_{\check{\alpha}} - \left(M^{\check{\alpha}\check{\beta}} v_{\check{\alpha}} \tau_{\check{\beta}} \mathbf{n} \right)_{,\check{\mu}} \tau^{\check{\mu}} = \mathbf{T}^{\check{2}} v_{\check{2}} = \pm \left(\lambda_0 + \frac{k_b}{4r_c^2} \right) \mathbf{e}_z. \quad (\text{B10})$$

The sign of the moment in table 1 is opposite that of (B9) due to the difference in orientation of the unit normal with respect to the surface.

The left and right surfaces: at the left ($\zeta^{\check{1}} = \alpha$) and right ($\zeta^{\check{1}} = 0$) edges, $\mathbf{v} = \pm \mathbf{e}_\theta$, $v_{\check{1}} = \pm r_c$, $v_{\check{2}} = 0$, $\boldsymbol{\tau} = \pm \mathbf{e}_z$, $\tau_{\check{1}} = 0$ and $\tau_{\check{2}} = \pm \ell$. In the same manner, we calculate

$$M = M^{\check{\alpha}\check{\beta}} v_{\check{\alpha}} v_{\check{\beta}} = M^{\check{1}\check{1}} v_{\check{1}} v_{\check{1}} = -\frac{k_b}{2r_c} \quad (\text{B11})$$

and

$$\mathbf{F} = \mathbf{T}^{\check{\alpha}} v_{\check{\alpha}} - \left(M^{\check{\alpha}\check{\beta}} v_{\check{\alpha}} \tau_{\check{\beta}} \mathbf{n} \right)_{,\check{\mu}} \tau^{\check{\mu}} = \mathbf{T}^{\check{1}} v_{\check{1}} = \pm \left(\lambda_0 - \frac{k_b}{4r_c^2} \right) \mathbf{e}_\theta. \quad (\text{B12})$$

The moment in (B11) again differs from the moment reported in table 1 due to the choice of normal vector.

Appendix C. The numerical calculation of the pull force

In this section we determine how to set the membrane velocity \mathbf{v} to a desired value \mathbf{v}_p at the centre of the membrane patch. Given the use of non-interpolatory basis functions, there is no unique way to do so. We thus choose to set the membrane velocity over the entire finite element containing the point of interest. The pull force $\mathcal{F}_p(t)$ resulting from the imposed displacement is calculated via variational arguments.

C.1. The inability to uniquely displace a single point on the membrane

To begin, at a chosen point (ξ_p^1, ξ_p^2) we intend for

$$\mathbf{v}(\xi_p^\alpha, t) = \mathbf{v}_p(t), \quad (\text{C1})$$

where $\mathbf{v}_p(t)$ is a known function. With the velocity discretisation in (A4), (C1) can be expressed as

$$\sum_{k=1}^{nen} N_k^e(\xi_p^\alpha) \mathbf{v}_k^e(t) = \mathbf{v}_p(t), \quad (\text{C2})$$

where $\{N_k^e(\xi_p^\alpha)\}$ and $\{\mathbf{v}_k^e\}$ are respectively the local basis functions and nodal velocities of the finite element Ω_p^e containing ξ_p^α . Importantly, the basis function values are set by the choice of ξ_p^α , and they are non-interpolatory (Piegl & Tiller 1997) – and thus, all nen nodal velocities contribute to (C2). Since $\mathbf{v}_p \in \mathbb{R}^3$ and $\mathbf{v}_k^e \in \mathbb{R}^3$ for all k , (C2) can be understood as a system of three scalar equations involving $3 \cdot nen$ scalar unknowns. As $nen = (\text{poly} + 1)^2$ for B-splines of polynomial order poly (Piegl & Tiller 1997), there is no unique way to specify the $\{\mathbf{v}_k^e\}$ in (C2). In practice, we set

$$\mathbf{v}_k^e(t) = \mathbf{v}_p(t) \quad \text{for } k \in \{1, 2, \dots, nen\}. \quad (\text{C3})$$

Given the properties of B-spline functions (Piegl & Tiller 1997), (C3) results in a uniform prescribed velocity over the entire parametric element Ω_p^e – expressed as

$$\mathbf{v}(\xi^\alpha, t) = \mathbf{v}_p(t) \quad \text{for all } \xi^\alpha \in \Omega_p^e. \quad (\text{C4})$$

C.2. The pull force when a finite element is uniformly displaced

In this section we calculate the force \mathcal{F}_p required to pull an entire finite element at a prescribed velocity as in (C4). From the strong form of the governing membrane equations (2.18), (2.19) we recognise the pull force is the area integral of the net body force per area f on the membrane – expressed as

$$\mathcal{F}_p(t) = \int_{\Omega_p^e} f(\xi^\alpha, t) J_\Omega(\xi^\alpha, t) d\Omega. \quad (\text{C5})$$

Since the prescribed velocity in (C4) can be viewed as a constraint, f is then understood as the associated Lagrange multiplier field over the element Ω_p^e .

Our task now is to determine how to calculate \mathcal{F}_p numerically, which is accomplished by considering how the weak form would be modified if (C4) was not satisfied directly. The principle of virtual power would then necessitate the quantity

$$\begin{aligned} & \delta \left\{ \int_{\Omega_p^e} f(\xi^\alpha, t) \cdot (\mathbf{v}(\xi^\alpha, t) - \mathbf{v}_p(t)) J_\Omega(\xi^\alpha, t) d\Omega \right\} \\ &= \int_{\Omega_p^e} \delta \mathbf{v} \cdot f J_\Omega d\Omega + \int_{\Omega_p^e} \delta f \cdot (\mathbf{v} - \mathbf{v}_p) J_\Omega d\Omega + \int_{\Omega_p^e} f \cdot (\mathbf{v} - \mathbf{v}_p) \delta J_\Omega d\Omega \end{aligned} \quad (\text{C6})$$

be subtracted from the direct Galerkin expression, where $\delta J_\Omega = \Delta t J_\Omega \mathbf{a}^\alpha \cdot \delta \mathbf{v}_{,\alpha}^m$ (see Appendix C.2.1 of Sahu *et al.* 2020b). At this point, the fundamental unknowns and

arbitrary variations are discretised as

$$\begin{aligned} \mathbf{v}(\zeta^{\check{\alpha}}, t) &= [\mathbf{N}_p^e] [\mathbf{v}_p^e(t)], & \mathbf{v}^m(\zeta^{\check{\alpha}}, t) &= [\mathbf{N}_p^e] [\mathbf{v}_p^{m,e}(t)], & \mathbf{f}(\zeta^{\check{\alpha}}, t) &= [\mathbf{N}_p^e] [\mathbf{f}_p^e(t)], \\ \delta \mathbf{v}(\zeta^{\check{\alpha}}) &= [\mathbf{N}_p^e] [\delta \mathbf{v}_p^e], & \delta \mathbf{v}^m(\zeta^{\check{\alpha}}) &= [\mathbf{N}_p^e] [\delta \mathbf{v}_p^{m,e}], & \delta \mathbf{f}(\zeta^{\check{\alpha}}) &= [\mathbf{N}_p^e] [\delta \mathbf{f}_p^e]. \end{aligned} \quad (\text{C7})$$

In (C7) we introduced the shorthand $[\mathbf{N}_p^e] := [\mathbf{N}^e(\zeta^{\check{\alpha}})]$ for all $\zeta^{\check{\alpha}} \in \Omega_p^e$. Upon substituting (C7) into the second line of (C6) and defining the elemental mass matrix $[\mathbf{M}_p^e]$ as

$$[\mathbf{M}_p^e] := \int_{\Omega_p^e} [\mathbf{N}_p^e]^T [\mathbf{N}_p^e] J_\Omega d\Omega \quad (\text{C8})$$

we find the quantity

$$\begin{aligned} & [\delta \mathbf{v}_p^e]^T [\mathbf{M}_p^e] [\mathbf{f}_p^e(t)] + [\delta \mathbf{f}_p^e]^T \left\{ [\mathbf{M}_p^e] [\mathbf{v}_p^e(t)] - \left(\int_{\Omega_p^e} [\mathbf{N}_p^e]^T J_\Omega d\Omega \right) \mathbf{v}_p(t) \right\} \\ & + [\delta \mathbf{v}_p^{m,e}]^T \int_{\Omega_p^e} [\mathbf{N}_p^e]_{,\check{\alpha}}^T \cdot (\mathbf{a}^{\check{\alpha}} \otimes \mathbf{f})(\mathbf{v} - \mathbf{v}_p) \Delta t J_\Omega d\Omega \end{aligned} \quad (\text{C9})$$

is to be subtracted from the discretised weak Galerkin expression \mathcal{G}_h . We now separately consider the portions of (C9) arising from variations in the pull force, material velocity and mesh velocity.

C.2.1. The pull force contribution

The portion of the direct Galerkin expression associated with the net force per unit area \mathbf{f} can be expressed as

$$\mathcal{G}_f = [\delta \mathbf{f}_p^e]^T \left(\int_{\Omega_p^e} [\mathbf{N}_p^e]^T J_\Omega d\Omega \right) \mathbf{v}_p(t) + [\delta \mathbf{f}_p^e]^T [\mathbf{M}_p^e] [\mathbf{v}_p^e(t)] = 0 \quad \forall \quad [\delta \mathbf{f}_p^e] \quad (\text{C10})$$

Importantly, the mass matrix $[\mathbf{M}_p^e]$ is invertible and the variation $[\delta \mathbf{f}_p^e]$ is arbitrary, so the nodal velocity degrees of freedom are found to be given by

$$[\mathbf{v}_p^e(t)] = [\mathbf{M}_p^e]^{-1} \left(\int_{\Omega_p^e} [\mathbf{N}_p^e]^T J_\Omega d\Omega \right) \mathbf{v}_p(t) \quad (\text{C11})$$

It is a well-known property of B-splines (Piegl & Tiller 1997) that if the constraint $\mathbf{v}(\zeta^{\check{\alpha}}, t) = \mathbf{v}_p(t)$ is enforced over the entire element, then the unique solution for the nodal degrees of freedom is given by

$$[\mathbf{v}_p^e(t)] = \begin{bmatrix} \mathbf{v}_p(t) \\ \vdots \\ \mathbf{v}_p(t) \end{bmatrix} = \begin{bmatrix} [1] \\ \vdots \\ [1] \end{bmatrix} \mathbf{v}_p(t). \quad (\text{C12})$$

By comparing (C11) and (C12), we recognise that

$$[\mathbf{M}_p^e]^{-1} \int_{\Omega_p^e} [\mathbf{N}_p^e]^T J_\Omega d\Omega = \begin{bmatrix} [1] \\ \vdots \\ [1] \end{bmatrix}, \quad (\text{C13})$$

which will be useful in our subsequent developments.

C.2.2. The material velocity contribution

The material velocity portion of the discretised direct Galerkin expression \mathcal{G}_h is given by

$$\mathcal{G}_v = [\delta \mathbf{v}]^T [\mathbf{r}^v] - [\delta \mathbf{v}_p^e]^T [\mathbf{M}_p^e] [f_p^e(t)] = 0 \quad \text{for all } [\delta \mathbf{v}], \quad (\text{C14})$$

where $[\delta \mathbf{v}]$ is the global variation of the nodal velocities, $[\delta \mathbf{v}_p^e]$ is the velocity variation of the nen nodes associated with Ω_p^e and $[\mathbf{r}^v]$ is the velocity portion of the global residual vector in the absence of a pull force. Let us imagine reordering the global velocity degrees of freedom such that those associated with Ω_p^e appear last, and those not associated with the pull force appear first. The global velocity unknowns, their arbitrary variation and residual vector in the absence of a pull force are respectively expressed as

$$[\mathbf{v}(t)] = \begin{bmatrix} [\mathbf{v}_{\bar{p}}(t)] \\ [\mathbf{v}_p^e(t)] \end{bmatrix}, \quad [\delta \mathbf{v}] = \begin{bmatrix} [\delta \mathbf{v}_{\bar{p}}] \\ [\delta \mathbf{v}_p^e] \end{bmatrix}, \quad \text{and} \quad [\mathbf{r}^v] = \begin{bmatrix} [\mathbf{r}_{\bar{p}}^v] \\ [\mathbf{r}_p^{v,e}] \end{bmatrix}. \quad (\text{C15})$$

Here, the subscript ‘ \bar{p} ’ is used to signify ‘not p,’ i.e. degrees of freedom not associated with the pulled element Ω_p^e . Substituting (C15) into (C14) yields

$$[\delta \mathbf{v}_{\bar{p}}]^T [\mathbf{r}_{\bar{p}}^v] + [\delta \mathbf{v}_p^e]^T [\mathbf{r}_p^{v,e}] - [\delta \mathbf{v}_p^e]^T [\mathbf{M}_p^e] [f_p^e(t)] = 0 \quad \text{for all } [\delta \mathbf{v}_{\bar{p}}], [\delta \mathbf{v}_p^e]. \quad (\text{C16})$$

Since the arbitrary variations $[\delta \mathbf{v}_{\bar{p}}]$ and $[\delta \mathbf{v}_p^e]$ are independent of one another, (C16) requires that $[\mathbf{r}_{\bar{p}}^v] = [\mathbf{0}]$ – which is the equation one obtains when the nodal velocities over Ω_p^e are set directly. Additionally, the nodal body force degrees of freedom are found to be given by

$$[f_p^e(t)] = [\mathbf{M}_p^e]^{-1} [\mathbf{r}_p^{v,e}]. \quad (\text{C17})$$

By substituting (C7) and then (C17) into (C5), we calculate the pull force as

$$\mathcal{F}_p(t) = \left(\int_{\Omega_p^e} [\mathbf{N}_p^e]^T J_\Omega d\Omega \right) [f_p^e(t)] = \left(\int_{\Omega_p^e} [\mathbf{N}_p^e]^T J_\Omega d\Omega \right) [\mathbf{M}_p^e]^{-1} [\mathbf{r}_p^{v,e}]. \quad (\text{C18})$$

Finally, recognising the mass matrix is symmetric and substituting the transpose of (C13) into (C18) yields

$$\mathcal{F}_p(t) = [[1] \ [1] \dots [1]] [\mathbf{r}_p^{v,e}], \quad (\text{C19})$$

which is straightforward to calculate within finite element subroutines. Equation (C19) is the main result of this section.

C.2.3. The mesh velocity contribution

The mesh velocity portion of \mathcal{G}_h can be written as

$$\mathcal{G}_m = [\delta \mathbf{v}^m]^T [\mathbf{r}^m] - [\delta \mathbf{v}_p^{m,e}]^T \int_{\Omega_p^e} [\mathbf{N}_p^e]_{,\check{\alpha}}^T \cdot (\mathbf{a}^{\check{\alpha}} \otimes \mathbf{f}) (\mathbf{v} - \mathbf{v}_p) \Delta t J_\Omega \quad (\text{C20})$$

for any arbitrary variation $[\delta \mathbf{v}^m]$. By reordering global velocity degrees of freedom in the same manner as (C15), we find that

$$[\delta \mathbf{v}_{\bar{p}}^m]^T [\mathbf{r}_{\bar{p}}^m] + [\delta \mathbf{v}_p^{m,e}]^T [\mathbf{r}_p^{m,e}] - [\delta \mathbf{v}_p^{m,e}]^T \int_{\Omega_p^e} [\mathbf{N}_p^e]_{,\check{\alpha}}^T \cdot (\mathbf{a}^{\check{\alpha}} \otimes \mathbf{f}) (\mathbf{v} - \mathbf{v}_p) \Delta t J_\Omega \quad (\text{C21})$$

for all independent variations $[\delta \mathbf{v}_{\bar{p}}]$ and $[\delta \mathbf{v}_p^e]$. Equation (C21) requires that $[\mathbf{r}_{\bar{p}}^m] = [\mathbf{0}]$, which solves for all mesh velocity degrees of freedom not associated with the finite

element Ω_p^e . In addition, we find that

$$[\mathbf{r}_p^{m,e}] = \int_{\Omega_p^e} [\mathbf{N}_p^e]_{,\check{\alpha}}^T \cdot (\mathbf{a}^{\check{\alpha}} \otimes \mathbf{f}) (\mathbf{v} - \mathbf{v}_p) \Delta t J_\Omega, \quad (\text{C22})$$

where $[\mathbf{r}_p^{m,e}]$ is the mesh velocity portion of the residual vector corresponding to Ω_p^e in the absence of a pull force. The following discussion explains why (C22) is not used in our code.

C.3. The numerical implementation

In our numerical implementation we calculate the pull force $\mathcal{F}_p(t)$ directly – rather than with a Lagrange multiplier field. To do so, we set the *nen* nodal material velocity values $\{\mathbf{v}_k^e(t)\}$ associated with Ω_p^e directly according to (C12). Though these nodes are removed from the degree-of-freedom list, the residual $[\mathbf{r}_p^{v,e}]$ is still calculated, from which the pull force is determined according to (C19). We also directly set the *nen* mesh velocity degrees of freedom $\{\mathbf{v}_k^{m,e}(t)\}$ associated with Ω_p^e to be $\mathbf{v}_p(t)$ in a similar fashion. Since the nodal mesh velocities on Ω_p^e are known, they are removed from the degree-of-freedom list, and so the integral term in (C22) is not evaluated in practice.

Appendix D. The pull force at small deformations

Consider a membrane patch which, prior to any deformation, is at the constant surface tension λ_0 and spans the region between two concentric circles in the x - y plane. We denote ℓ as the diameter of the outer circle and $2r_p$ as the diameter of the inner circle. Eventually, we will take the limit as $r_p \rightarrow 0$. When a deformation is applied quasi-statically and the membrane height $h = h(r, \theta)$ above the x - y plane is small (i.e. $h \ll \ell$), the membrane shape is known to satisfy (Sahu 2022, Chapter VII)

$$\lambda_0 \nabla^2 h - \frac{1}{2} k_b \nabla^4 h = 0, \quad (\text{D1})$$

where

$$\nabla^2(\cdot) = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial(\cdot)}{\partial r} \right) + \frac{1}{r^2} \frac{\partial(\cdot)}{\partial \theta} \quad (\text{D2})$$

is the 2-D Laplacian expressed in terms of the canonical cylindrical coordinates r and θ . In what follows, all lengths are non-dimensionalised by $\ell/2$, for which

$$h^* := \frac{2h}{\ell}, \quad r^* := \frac{2r}{\ell}, \quad \text{and} \quad \hat{\nabla}^2 := \frac{\ell^2}{4} \nabla^2. \quad (\text{D3})$$

With (4.12) and (D3), the shape equation (D1) is presented in dimensionless form as

$$\hat{\nabla}^2 h^* - \frac{\Gamma}{2} \hat{\nabla}^4 h^* = 0. \quad (\text{D4})$$

In comparing (D4) with the description in Powers *et al.* (2002), we recognise the small parameter ϵ in the latter is given by $2/\Gamma$ in our notation. Thus, for an axisymmetric membrane with (i) prescribed displacement z_p at $r = r_p$, (ii) zero slope at $r = r_p$, (iii) no displacement at $r = \ell/2$, and (iv) no moment at $r = \ell/2$, we reproduce the Powers *et al.* (2002) solution as

$$h^*(r^*) = z_p^* \cdot \frac{\alpha K_1(\alpha) \ln r^* + K_0(\alpha r^*/r_p^*)}{\alpha K_1(\alpha) \ln r_p^* + K_0(\alpha)}, \quad (\text{D5})$$

where

$$\alpha := r_p^* \sqrt{\frac{\Gamma}{2}} \quad (\text{D6})$$

is a constant parameter defined for notational convenience. Note that in our numerics, we require zero slope rather than zero moment on the outer boundary. However, when membrane deformations are small, we do not expect this choice of boundary condition to affect the pull force calculation. In (D5) and (D6), K_0 and K_1 are modified Bessel functions of the second kind, and $z_p^* := 2z_p/\ell$ is the dimensionless displacement at the inner membrane boundary – which is located at $r^* = r_p^* := 2r_p/\ell$.

With the solution for the membrane shape at a given displacement in (D5), we seek to determine the magnitude of the pull force in the e_z direction: $\mathcal{F}_{p,z}$. The pull force is related to F_z , the force per length in the e_z direction at the inner boundary, via

$$\mathcal{F}_{p,z} = \lim_{r_p \rightarrow 0} 2\pi r_p F_z. \quad (\text{D7})$$

Since the membrane patch is parallel to the $x-y$ plane at the inner boundary, we recognise that (Sahu 2022, Chapter V, § 6(e))

$$F_z = k_b \left. \frac{\partial H}{\partial r} \right|_{r_p} = \frac{k_b}{2} \left. \left(\frac{\partial}{\partial r} (\nabla^2 h) \right) \right|_{r_p}, \quad (\text{D8})$$

where the mean curvature $H = (1/2)\nabla^2 h$ in the limit of small deformations. Moreover, since the membrane height is axisymmetric and has zero slope at the inner boundary, we apply (D2) and find that

$$\mathcal{F}_{p,z} = \lim_{r_p^* \rightarrow 0} \frac{2\pi k_b}{\ell} \left(r_p^* \left. \frac{d^3 h^*}{d(r^*)^3} \right|_{r_p^*} + \left. \frac{d^2 h^*}{d(r^*)^2} \right|_{r_p^*} \right). \quad (\text{D9})$$

With the height solution in (D5) and properties of Bessel functions (Abramowitz & Stegun 1964), the terms in parenthesis in (D9) are found to be

$$\left. \frac{d^2 h^*}{d(r^*)^2} \right|_{r_p^*} = \frac{z_p^* (\Gamma/2) K_0(\alpha)}{\alpha K_1(\alpha) \ln r_p^* + K_0(\alpha)} \quad (\text{D10})$$

and

$$r_p^* \left. \frac{d^3 h^*}{d(r^*)^3} \right|_{r_p^*} = \frac{-z_p^* (\Gamma/2) [K_0(\alpha) + \alpha K_1(\alpha)]}{\alpha K_1(\alpha) \ln r_p^* + K_0(\alpha)}. \quad (\text{D11})$$

Substituting (D10) and (D11) into (D9) yields

$$\mathcal{F}_{p,z} = \lim_{r_p^* \rightarrow 0} \frac{z_p^* \pi k_b \Gamma}{\ell} \left(\frac{-\alpha K_1(\alpha)}{\alpha K_1(\alpha) \ln r_p^* + K_0(\alpha)} \right). \quad (\text{D12})$$

At this point, we evaluate the limit by recognising $\alpha \rightarrow 0$ as $r_p^* \rightarrow 0$. In addition, we use the Bessel function relations (Abramowitz & Stegun 1964)

$$\lim_{\alpha \rightarrow 0} \alpha K_1(\alpha) = 1 \quad \text{and} \quad \lim_{\alpha \rightarrow 0} K_0(\alpha) = -\ln(\alpha). \quad (\text{D13})$$

By substituting (4.14), (D6) and (D13) into (D12) and rearranging terms, we obtain the small-deformation pull force expression in (4.20) – presented here as

$$\mathcal{F}_{p,z} = \frac{\pi k_b z_p}{r_c^2 \ln(\Gamma/2)}. \quad (\text{D14})$$

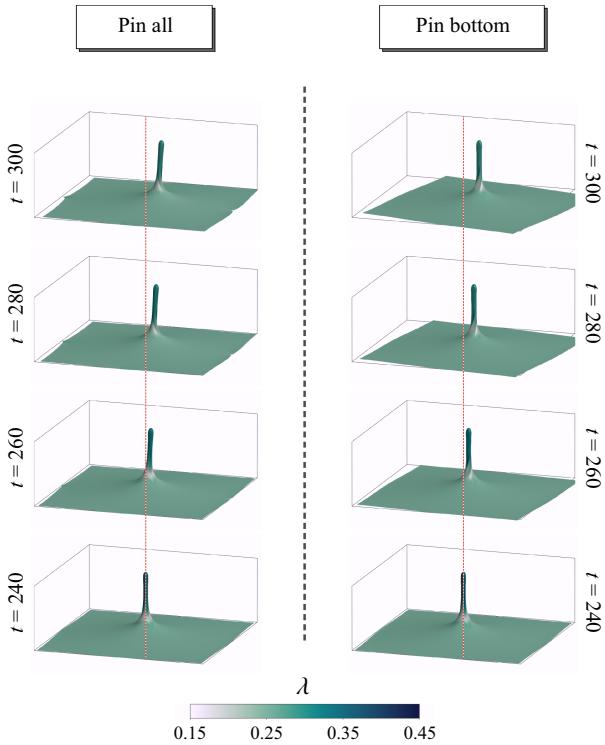


Figure 10. Tether translation in the e_x direction with a Lagrangian mesh motion. In the left column, simulation boundary conditions are those specified in the first two rows of table 3. Twisting and tilting of the tether, along with further striation of the surface tension, ensue. In the right column, an attempt is made to remove unphysical constraints on the membrane by only pinning the centre of the bottom boundary. Upon lateral pulling, the mesh rotates about the centre of the bottom edge; twisting and tension striation are once again observed. Both simulations fail to capture the expected behaviour shown in the left column of figure 9. As in the main text, times are measured in units of $\zeta r_c^2/k_b$, the pull velocity is specified in the first row of (4.22), and all other parameters are identical to those in figure 4. The vertical red lines are a visual aid to highlight the motion of the tether.

Appendix E. The translation of a tether with a Lagrangian mesh motion

When a pulled tether is translated laterally across a membrane surface, lipids in the surrounding patch flow in-plane to accommodate the large out-of-plane shape deformations. For a given prescribed lateral tether velocity as in (4.22), we do not know the corresponding velocity field $v(\zeta^\alpha, t)$ over the membrane patch *a priori*. Thus, any boundary conditions that pin certain nodes yield unphysical results, as shown in figure 10. If no nodes are pinned, however, a rigid body translation results – and the dynamics of lateral tether motion are not obtained. Lagrangian simulations are thus unable to capture the physics of tether translation.

REFERENCES

- ABRAMOWITZ, M. & STEGUN, I.A. 1964 *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, vol. 55. Dover.
 AGRAWAL, A. & STEIGMANN, D.J. 2009 Modeling protein-mediated morphology in biomembranes. *Biomech. Model. Mechanobiol.* **8**, 371–379.
 AGRAWAL, A. & STEIGMANN, D.J. 2011 A model for surface diffusion of trans-membrane proteins on lipid bilayers. *Z. Angew. Math. Phys.* **62**, 549–563.

- AL-IZZI, S.C., SENS, P. & TURNER, M.S. 2020 Shear-driven instabilities of membrane tubes and dynamin-induced scission. *Phys. Rev. Lett.* **125**, 018101.
- AL-IZZI, S.C., TURNER, M.S. & SENS, P. 2024 Membrane tubes with active pumping: water transport, vacuole formation and osmoregulation. [arXiv:2409.14835](https://arxiv.org/abs/2409.14835)
- ALKADRI, A.M. & MANDADAPU, K.K. 2024 The irreversible thermodynamics of curved lipid membranes II: permeability and osmosis. [arXiv:2412.19300](https://arxiv.org/abs/2412.19300)
- ARROYO, M. & DESIMONE, A. 2009 Relaxation dynamics of fluid membranes. *Phys. Rev. E* **79**, 31915–31931.
- BABUŠKA, I. 1973 The finite element method with Lagrangian multipliers. *Numer. Math.* **20**, 179–192.
- BAR-ZIV, R., MENES, R., MOSES, E. & SAFRAN, S.A. 1995 Local unbinding of pinched membranes. *Phys. Rev. Lett.* **75**, 3356–3359.
- BARRETT, J.W., GARCKE, H. & NÜRNBERG, R. 2008a On the parametric finite element approximation of evolving hypersurfaces in \mathbb{R}^3 . *J. Comput. Phys.* **227**, 4281–4307.
- BARRETT, J.W., GARCKE, H. & NÜRNBERG, R. 2008b Parametric approximation of Willmore flow and related geometric evolution equations. *SIAM J. Sci. Comput.* **31**, 225–253.
- BARRETT, J.W., GARCKE, H. & NÜRNBERG, R. 2015 Numerical computations of the dynamics of fluidic membranes and vesicles. *Phys. Rev. E* **92**, 052704.
- BEZANSON, J., EDELMAN, A., KARPINSKI, S. & SHAH, V.B. 2017 A fresh approach to numerical computing. *SIAM Rev.* **59**, 65–98.
- BONITO, A., NOCHETTO, R.H. & PAULETTI, M.S. 2010 Parametric FEM for geometric biomembranes. *J. Comput. Phys.* **229**, 3171–3188.
- BREZZI, F. 1974 On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Ref. Fr. Automat. Infor.* **8**, 129–151.
- BREZZI, F. & PITKÄRANTA, J. 1984 On the stabilization of finite element approximations of the Stokes equations. In *Efficient Solutions of Elliptic Systems* (ed. W. Hackbusch), pp. 11–19, Vieweg+Teubner Verlag.
- BROCHARD-WYART, F., BORGHI, N., CUVELIER, D. & NASSOY, P. 2006 Hydrodynamic narrowing of tubes extruded from cells. *Proc. Natl. Acad. Sci. USA* **103**, 7660–7663.
- CANHAM, P.B. 1970 The minimum energy of bending as a possible explanation of the biconcave shape of the human red blood cell. *J. Theor. Biol.* **26**, 61–81.
- CAPOVILLA, R. & GUVEN, J. 2002 Stresses in lipid membranes. *J. Phys. A: Math. General* **35**, 6233–6247.
- CAPOVILLA, R., GUVEN, J. & SANTIAGO, J.A. 2002 Lipid membranes with an edge. *Phys. Rev. E* **66** (2), 021607.
- COTTRELL, J.A., HUGHES, T.J.R. & BAZILEVS, Y. 2009 *Isogeometric Analysis: Toward Integration of CAD and FEA*, 1st edn. John Wiley & Sons.
- CUVELIER, D., DERÉNYI, I., BASSEREAU, P. & NASSOY, P. 2005 Coalescence of membrane tethers: experiments, theory, and applications. *Biophys. J.* **88**, 2714–2726.
- DAI, J. & SHEETZ, M.P. 1995 Mechanical properties of neuronal growth cone membranes studied by tether formation with laser optical tweezers. *Biophys. J.* **68**, 988–996.
- DANISCH, S. & KRUMBIEGEL, J. 2021 Makie.jl: flexible high-performance data visualization for Julia. *J. Open Source Softw.* **6**, 3349.
- DATAR, A., BORN SCHLÖGL, T., BASSEREAU, P., PROST, J. & PULLARKAT, P.A. 2015 Dynamics of membrane tethers reveal novel aspects of cytoskeleton–membrane interactions in axons. *Biophys. J.* **108**, 489–497.
- DERÉNYI, I., JÜLICHER, F. & PROST, J. 2002 Formation and interaction of membrane tubes. *Phys. Rev. Lett.* **88**, 238101–238104.
- DHARMAVARAM, S. & HANNA, J.A. 2024 Shear stresses in fluid and solid membranes with bending elasticity. *arXiv preprint*, [arXiv:2410.13049](https://arxiv.org/abs/2410.13049).
- DOHRMANN, C.R. & BOCHEV, P.B. 2004 A stabilized finite element method for the Stokes problem based on polynomial pressure projections. *Int. J. Numer. Meth. Fluids* **46**, 183–201.
- DORTDIVANLIOGLU, B., KRISCHOK, A., BEIRÃO DA VEIGA, L. & LINDER, C. 2018 Mixed isogeometric analysis of strongly coupled diffusion in porous materials. *Int. J. Numer. Meth. Engng* **114**, 28–46.
- DU, Q., LIU, C. & WANG, X. 2004 A phase field approach in the numerical study of the elastic bending energy for vesicle membranes. *J. Comput. Phys.* **198**, 450–468.
- DUBEY, G.P. *et al.* 2016 Architecture and characteristics of bacterial nanotubes. *Dev. Cell.* **36**, 453–461.
- DZIUK, G. 2008 Computational parametric Willmore flow. *Numer. Math.* **111** (1), 55–80.
- ELLIOTT, C.M. & STINNER, B. 2010 Modeling and computation of two phase geometric biomembranes using surface finite elements. *J. Comput. Phys.* **229**, 6585–6612.
- EVANS, E.A. & SKALAK, R. 1980 *Mechanics and Thermodynamics of Biomembranes*. CRC Press.

- EVANS, E.A. 1974 Bending resistance and chemically induced moments in membrane bilayers. *Biophys. J.* **14**, 923–931.
- EVANS, E.A. & HOCHMUTH, R.M. 1978 Mechanochemical properties of membranes. In *Membrane Properties: Mechanical Aspects, Receptors, Energetics and Calcium-Dependence of Transport* (ed. F. Bronner & A. Kleinzeller), vol. 10, pp. 1–64. Academic Press.
- EVANS, E.A. & YEUNG, A. 1994 Hidden dynamics in rapid changes of bilayer shape. *Chem. Phys. Lipids* **73**, 39–56.
- FAIZI, H.A., GRANEK, R. & VLAHOVSKA, P.M. 2024 Curvature fluctuations of fluid vesicles reveal hydrodynamic dissipation within the bilayer. *Proc. Natl. Acad. Sci. USA* **121**, e2413557121.
- FENG, F. & KLUG, W.S. 2006 Finite element modeling of lipid bilayer membranes. *J. Comput. Phys.* **220**, 394–408.
- FONDA, P., AL-IZZI, S.C., GIOMI, L. & TURNER, M.S. 2020 Measuring Gaussian rigidity using curved substrates. *Phys. Rev. Lett.* **125**, 188002.
- FOURNIER, J.B. 1996 Nontopological saddle-splay and curvature instabilities from anisotropic membrane inclusions. *Phys. Rev. Lett.* **76**, 4436–4439.
- FYGENSON, D.K., MARKO, J.F. & LIBCHABER, A. 1997 Mechanics of microtubule-based membrane extension. *Phys. Rev. Lett.* **79**, 4497–4500.
- GOLDSTEIN, R.E., NELSON, P., POWERS, T. & SEIFERT, U. 1996 Front propagation in the pearlning instability of tubular vesicles. *J. Phys. II* **6**, 767–796.
- GOMIS PEREZ, C., DUDZINSKI, N.R., ROUCHES, M., LANDAJUELA, A., MACHTA, B., ZENISEK, D. & KARATEKIN, E. 2022 Rapid propagation of membrane tension at retinal bipolar neuron presynaptic terminals. *Science Adv.* **8** (1), eabl4411.
- GREEN, A.E. & NAGHDI, P.M. 1968 The Cosserat surface. In *Mechanics of Generalized Continua* (ed. E. Kröner), pp. 36–48. Springer.
- GROSS, B.J. & ATZBERGER, P.J. 2018 Hydrodynamic flows on curved surfaces: spectral numerical methods for radial manifold shapes. *J. Comput. Phys.* **371**, 663–689.
- GUVEN, J. 2004 Membrane geometry with auxiliary variables and quadratic constraints. *J. Phys. A: Math. General* **37**, L313–L319.
- HASSINGER, J.E., OSTER, G., DRUBIN, D.G. & RANGAMANI, P. 2017 Design principles for robust vesiculation in clathrin-mediated endocytosis. *Proc. Natl. Acad. Sci. USA* **114**, E1118–E1127.
- HELFICH, W. 1973 Elastic properties of lipid bilayers: theory and possible experiments. *Z. Naturforsch. C* **28**, 693–699.
- HIGGINS, M.K. & McMAHON, H.T. 2002 Snap-shots of clathrin-mediated endocytosis. *Trends Biochem. Sci.* **27**, 257–263.
- HU, D., ZHANG, P. & W., E. 2007 Continuum theory of a moving membrane. *Phys. Rev. E* **75**, 041605.
- IMACHI, H. *et al.* 2020 Isolation of an archaeon at the prokaryote–eukaryote interface. *Nature* **577**, 519–525.
- KOH, K.J., TOSHNIWAL, D. & CIRAK, F. 2022 An optimally convergent smooth blended B-spline construction for semi-structured quadrilateral and hexahedral meshes. *Comput. Meth. Appl. Mech. Engng* **399**, 115438.
- KOSTER, G., CACCIUTO, A., DERÉNYI, I., FRENKEL, D. & DOGTEROM, M. 2005 Force barriers for membrane tube formation. *Phys. Rev. Lett.* **94**, 068101.
- KOSTER, G., VANDUIJN, M., HOFS, B. & DOGTEROM, M. 2003 Membrane tube formation from giant vesicles by dynamic association of motor proteins. *Proc. Natl. Acad. Sci. USA* **100**, 15583–15588.
- LADYZHENSKAYA, O.A. 1969 The mathematical theory of viscous incompressible flow. In *Mathematics and Its Applications*. 2nd edn vol. 2. Gordon & Breach.
- LAUFFENBURGER, D.A. & HORWITZ, A.F. 1996 Cell migration: a physically integrated molecular process. *Cell* **84**, 359–369.
- LEDUC, C., CAMPÀS, O., ZELDOVICH, K.B., ROUX, A., JOLIMAITRE, P., BOUREL-BONNET, L., GOUD, B., JOANNY, J.-F., BASSEREAU, P. & PROST, J. 2004 Cooperative extraction of membrane nanotubes by molecular motors. *Proc. Natl. Acad. Sci. USA* **101**, 17096–17101.
- LIN, C., KUMAR, D., RICHTER, C.M., WANG, S., SCHROEDER, C.M. & NARSIMHAN, V. 2021 Vesicle dynamics in large amplitude oscillatory extensional flow. *J. Fluid Mech.* **929**, A43.
- LYNESS, J.N. 1968 Differentiation formulas for analytic functions. *Maths Comput.* **22**, 352–362.
- LYNESS, J.N. & MOLER, C.B. 1967 Numerical differentiation of analytic functions. *SIAM J. Numer. Anal.* **4**, 202–210.
- MA, L. & KLUG, W.S. 2008 Viscous regularization and r-adaptive remeshing for finite element analysis of lipid membrane mechanics. *J. Comput. Phys.* **227**, 5816–5835.

- MAITRA, A., SRIVASTAVA, P., RAO, M. & RAMASWAMY, S. 2014 Activating membranes. *Phys. Rev. Lett.* **112**, 258101.
- MALKUS, D.S. & HUGHES, T.J.R. 1978 Mixed finite element methods – reduced and selective integration techniques: a unification of concepts. *Comput. Meth. Appl. Mech. Engng* **15**, 63–81.
- MCMAHON, H.T. & BOUCROT, E. 2011 Molecular mechanism and physiological functions of clathrin-mediated endocytosis. *Nat. Rev. Mol. Cell Biol.* **12**, 517–533.
- MERCKER, M., MARCINIAK-CZOCZRA, A., RICHTER, T. & HARTMANN, D. 2013 Modeling and computing of deformation dynamics of inhomogeneous biological surfaces. *SIAM J. Appl. Maths* **73**, 1768–1792.
- NAGHDI, P.M. 1973 The theory of shells and plates. In *Linear Theories of Elasticity and Thermoelasticity: Linear and Nonlinear Theories of Rods, Plates, and Shells* (ed. C. Truesdell), pp. 425–640. Springer.
- NARSIMHAN, V., SPANN, A.P. & SHAQFEH, E.S.G. 2015 Pearlring, wrinkling, and buckling of vesicles in elongational flows. *J. Fluid Mech.* **777**, 1–26.
- NICHOL, J.A. & HUTTER, O.F. 1996 Tensile strength and dilatational elasticity of giant sarcolemmal vesicles shed from rabbit muscle. *J. Physiol.* **493**, 187–198.
- OMAR, Y.A.D., SAHU, A., SAUER, R.A. & MANDADAPU, K.K. 2020 Non-axisymmetric shapes of biological membranes from locally induced curvature. *Biophys. J.* **119**, 1065–1077.
- OPPENHEIMER, N. & DIAMANT, H. 2010 Correlated dynamics of inclusions in a supported membrane. *Phys. Rev. E* **82**, 041912.
- OPPENHEIMER, N. & DIAMANT, H. 2011 In-plane dynamics of membranes with immobile inclusions. *Phys. Rev. Lett.* **107**, 258102.
- PAUL, K., ZIMMERMANN, C., MANDADAPU, K.K., HUGHES, T.J.R., LANDIS, C.M. & SAUER, R.A. 2020 An adaptive space-time phase field formulation for dynamic fracture of brittle shells based on LR NURBS. *Comput. Mech.* **65**, 1039–1062.
- PIEGL, L. & TILLER, W. 1997 *The NURBS Book*. 2nd edn. Springer-Verlag.
- POLLARD, J., AL-IZZI, S.C. & MORRIS, R.G. 2024 Gauge freedom and objective rates in the morphodynamics of fluid deformable surfaces: the Jaumann rate vs. the material derivative. *arXiv:2406.18014*
- POWERS, T.R., HUBER, G. & GOLDSTEIN, R.E. 2002 Fluid-membrane tethers: minimal surfaces and elastic boundary layers. *Phys. Rev. E* **65** (4), 041901.
- RAHIMI, M. & ARROYO, M. 2012 Shape dynamics, lipid hydrodynamics, and the complex viscoelasticity of bilayer membranes. *Phys. Rev. E* **86**, 11932–11946.
- RAHIMI, M., DESIMONE, A. & ARROYO, M. 2013 Curved fluid membranes behave laterally as effective viscoelastic media. *Soft Matt.* **9**, 11033–11045.
- RANGAMANI, P., AGRAWAL, A., MANDADAPU, K.K., OSTER, G. & STEIGMANN, D.J. 2012 Interaction between surface shape and intra-surface viscous flow on lipid membranes. *Biomech. Model. Mech.* **12**, 833–845.
- RANGAMANI, P., MANDADAPU, K.K. & OSTER, G. 2014 Protein-induced membrane curvature alters local membrane tension. *Biophys. J.* **107**, 751–762.
- RANGARAJAN, R. & GAO, H. 2015 A finite element method to compute three-dimensional equilibrium configurations of fluid membranes: optimal parameterization, variational formulation and applications. *J. Comp. Phys.* **297**, 266–294.
- RAUCHER, D. & SHEETZ, M.P. 1999 Characteristics of a membrane reservoir buffering membrane tension. *Biophys. J.* **77**, 1992–2002.
- REBOUCAS, R.B., FAIZI, H.A., MIKSIS, M.J. & VLAHOVSKA, P.M. 2024 Stationary shapes of axisymmetric vesicles beyond lowest-energy configurations. *Soft Matt.* **20**, 2258–2271.
- REUTHER, S., NITSCHKE, I. & VOIGT, A. 2020 A numerical approach for fluid deformable surfaces. *J. Fluid Mech.* **900**, R8.
- RODRIGUES, D.S., AUSAS, R.F., MUT, F. & BUSCAGLIA, G.C. 2015 A semi-implicit finite element method for viscous lipid membranes. *J. Comput. Phys.* **298**, 565–584.
- ROUX, A., CAPPELLO, G., CARTAUD, G., PROST, J., GOUD, B. & BASSEREAU, P. 2002 A minimal system allowing tubulation with molecular motors pulling on giant liposomes. *Proc. Natl. Acad. Sci. USA* **99**, 5394–5399.
- ROWER, D.A., PADIDAR, M. & ATZBERGER, P.J. 2022 Surface fluctuating hydrodynamics methods for the drift-diffusion dynamics of particles and microstructures within curved fluid interfaces. *J. Comput. Phys.* **455**, 110994.
- SABASS, B. & STONE, H.A. 2016 Role of the membrane for mechanosensing by tethered channels. *Phys. Rev. Lett.* **116**, 258101.
- SAHU, A. 2022 Irreversible thermodynamics and hydrodynamics of biological membranes, PhD thesis, University of California at Berkeley, USA.

- SAHU, A. 2024 MembraneAleFem.jl: Arbitrary Lagrangian–Eulerian lipid membrane finite element simulations in Julia. <https://github.com/sahu-lab/MembraneAleFem.jl>, source code archived at doi: [10.5281/zenodo.15611870](https://doi.org/10.5281/zenodo.15611870).
- SAHU, A., GLISMAN, A., TCHOUFAG, J. & MANDADAPU, K.K. 2020a Geometry and dynamics of lipid membranes: the Scrinien–Love number. *Phys. Rev. E* **101**, 052401.
- SAHU, A., OMAR, Y.A.D. & MANDADAPU, K.K. 2020b Arbitrary Lagrangian–Eulerian finite element formulation for curved and deforming surfaces I. General theory and application to fluid interfaces. *J. Comput. Phys.* **407**, 109253.
- SAHU, A., SAUER, R.A. & MANDADAPU, K.K. 2017 Irreversible thermodynamics of curved lipid membranes. *Phys. Rev. E* **96**, 042409.
- SAMANTA, R. & OPPENHEIMER, N. 2021 Vortex flows and streamline topology in curved biological membranes. *Phys. Fluids* **33**, 051906.
- SAUER, R.A. & DUONG, T.X. 2017 On the theoretical foundations of thin solid and liquid shells. *Maths Mech. Solids* **22**, 343–371.
- SAUER, R.A., DUONG, T.X., MANDADAPU, K.K. & STEIGMANN, D.J. 2017 A stabilized finite element formulation for liquid shells and its application to lipid bilayers. *J. Comput. Phys.* **330**, 436–466.
- SCIAKY, N., PRESLEY, J., SMITH, C., ZAAL, K.J.M., COLE, N., MOREIRA, J.E., TERASAKI, M., SIGGIA, E. & LIPPINCOTT-SCHWARTZ, J. 1997 Golgi tubule traffic and the effects of brefeldin a visualized in living cells. *J. Cell Biol.* **139**, 1137–1155.
- SCOTT, Z.C., STEEN, S.B., HUBER, G., WESTRATE, L.M. & KOSLOVER, E.F. 2024 The endoplasmic reticulum as an active liquid network. *Proc. Natl. Acad. Sci. USA* **121**, e240975121.
- SCRIVEN, L.E. 1960 Dynamics of a fluid interface: equation of motion for Newtonian surface fluids. *Chem. Engng Sci.* **12**, 98–108.
- SEIFERT, U. 1997 Configurations of fluid membranes and vesicles. *Adv. Phys.* **46**, 13–137.
- SEIFERT, U., BERNDL, K. & LIPOWSKY, R. 1991 Shape transformations of vesicles: phase diagram for spontaneous-curvature and bilayer-coupling models. *Phys. Rev. A* **44**, 1182–1202.
- SEIFERT, U. & LANGER, S.A. 1993 Viscous modes of fluid bilayer membranes. *Europhys. Lett.* **23**, 71–76.
- SHI, Z., GRABER, Z.T., BAUMGART, T., STONE, H.A. & COHEN, A.E. 2018 Cell membranes resist flow. *Cell* **175**, 1769–1779.
- SHI, Z., INNES-GOLD, S. & COHEN, A.E. 2022 Membrane tension propagation couples axon growth and collateral branching. *Sci. Adv.* **8**, eab01297.
- SIGURDSSON, J.K. & ATZBERGER, P.J. 2016 Hydrodynamic coupling of particle inclusions embedded in curved lipid bilayer membranes. *Soft Matt.* **12**, 6685–6707.
- SOWINSKI, S. et al. 2008 Membrane nanotubes physically connect T cells over long distances presenting a novel route for HIV-1 transmission. *Nat. Cell Biol.* **10**, 211–219.
- STEIGMANN, D.J. 1998 On the relationship between the Cosserat and Kirchhoff–Love theories of elastic shells. *Maths Mech. Solids* **4**, 275–288.
- STEIGMANN, D.J. 1999 Fluid films with curvature elasticity. *Arch. Ration. Mech. Anal.* **150**, 127–152.
- STONE, H.A. 2010 Interfaces: in fluid mechanics and across disciplines. *J. Fluid Mech.* **645**, 1–25.
- STONE, H.A. & MASOUD, H. 2015 Mobility of membrane-trapped particles. *J. Fluid Mech.* **781**, 494–505.
- SUBRAMANIAM, A.B., GUIDOTTI, G., MANOHARAN, V.N. & STONE, H.A. 2013 Glycans pattern the phase behaviour of lipid membranes. *Nat. Mater.* **12**, 128–133.
- TANAKA, M., FUJIKAWA, M., BALZANI, D. & SCHRÖDER, J. 2014 Robust numerical calculation of tangent moduli at finite strains based on complex-step derivative approximation and its application to localization analysis. *Comput. Meth. Appl. Mech. Engng* **269**, 454–470.
- TERASAKI, M., CHEN, L.B. & FUJIWARA, K. 1986 Microtubules and the endoplasmic reticulum are highly interdependent structures. *J. Cell Biol.* **103**, 1557–1568.
- TERASAKI, M. et al. 2013 Stacked endoplasmic reticulum sheets are connected by helicoidal membrane motifs. *Cell* **154**, 285–296.
- TIMOSHENKO, S.P. 1921 On the correction for shear of the differential equation for transverse vibrations of prismatic bars. *Lond. Edinburgh Dublin Phil. Mag. J. Sci.* **41**, 744–746.
- TIMOSHENKO, S.P. 1922 X. On the transverse vibrations of bars of uniform cross-section. *Lond. Edinburgh Dublin Phil. Mag. J. Sci.* **43**, 125–131.
- TORRES-SÁNCHEZ, A., MILLÁN, D. & ARROYO, M. 2019 Modelling fluid deformable surfaces with an emphasis on biological interfaces. *J. Fluid Mech.* **872**, 218–271.
- TOSHNIWAL, D., SPELEERS, H. & HUGHES, T.J.R. 2017 Smooth cubic spline spaces on unstructured quadrilateral meshes with particular emphasis on extraordinary points: geometric design and isogeometric analysis considerations. *Comput. Meth. Appl. Mech. Engng* **327**, 411–458.

- TRAN, P.D., BLANPIED, T.A. & ATZBERGER, P.J. 2022 Protein drift-diffusion dynamics and phase separation in curved cell membranes and dendritic spines: hybrid discrete-continuum methods. *Phys. Rev. E* **106**, 044402.
- VENKATESH, A., BHARGAVA, A. & NARSIMHAN, V. 2025 Stability of cylindrical, multicomponent vesicles. *J. Fluid Mech.* **1003**, A18.
- VENKATESH, A. & NARSIMHAN, V. 2024 Shape dynamics of nearly spherical, multicomponent vesicles under shear flow, [arXiv:2409.15102](https://arxiv.org/abs/2409.15102)
- VLAHOVSKA, P.M. 2016 Dynamics of membrane-bound particles: capsules and vesicles. In *Fluid–Structure Interactions in Low-Reynolds-Number Flows* (eds. C. DUPRAT & H.A. STONE), pp. 313–346. The Royal Society of Chemistry.
- VLAHOVSKA, P.M. & GRACIA, R.S. 2007 Dynamics of a viscous vesicle in linear flows. *Phys. Rev. E* **75**, 016313.
- WALANI, N., TORRES, J. & AGRAWAL, A. 2015 Endocytic proteins drive vesicle growth via instability in high membrane tension environment. *Proc. Natl. Acad. Sci. USA* **112**, E1423–E1432.
- WAXMAN, A.M. 1984 Dynamics of a couple-stress fluid membrane. *Stud. Appl. Maths* **70**, 63–86.
- WEI, X., ZHANG, Y.J., TOSHNIWAL, D., SPELEERS, H., LI, X., MANNI, C., EVANS, J.A. & HUGHES, T.J.R. 2018 Blended B-spline construction on unstructured quadrilateral and hexahedral meshes with optimal convergence rates in isogeometric analysis. *Comput. Meth. Appl. Mech. Engng* **341**, 609–639.
- YU, Q. & KOŠMRLJ, A. 2023 Pattern formation of phase-separated lipid domains in bilayer membranes. [arXiv:2309.05160](https://arxiv.org/abs/2309.05160)
- ZHONG-CAN, O.-Y. & HELFRICH, W. 1989 Bending energy of vesicle membranes: general expressions for the first, second, and third variation of the shape energy and applications to spheres and cylinders. *Phys. Rev. A* **39**, 5280–5288.
- ZHU, C., LEE, C.T. & RANGAMANI, P. 2022 Mem3DG: modeling membrane mechanochemical dynamics in 3D using discrete differential geometry. *Biophys. Rep.* **2**, 100062.
- ZIENKIEWICZ, O.C. & CODINA, R. 1995 A general algorithm for compressible and incompressible flow – characteristic-based scheme. *Intl J. Numer. Meth. Fluids* **20**, 869–885.
- ZIENKIEWICZ, O. & TAYLOR, R.L. 2014 The finite element method for solid and structural mechanics, 7th edn. Butterworth-Heinemann.
- ZIENKIEWICZ, O., TAYLOR, R.L. & ZHU, J.Z. 2013 The finite element method: its basis and fundamentals, 7th edn. Butterworth-Heinemann.
- ZIENKIEWICZ, O.C. & NAKAZAWA, S. 1984 On variational formulation and its modifications for numerical solution. *Comput. Struct.* **19**, 303–313.