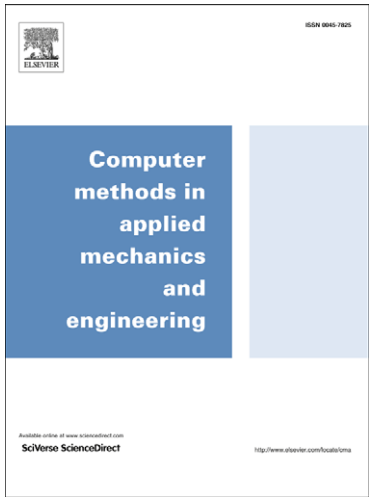

A finite element method for viscous membranes

Italo V. Tasso, Gustavo C. Buscaglia

PII: S0045-7825(12)00336-2
DOI: http://dx.doi.org/10.1016/j.cma.2012.10.021
Reference: CMA 9846

To appear in: *Comput. Methods Appl. Mech. Engrg.*

Received Date: 12 March 2012
Revised Date: 31 August 2012
Accepted Date: 30 October 2012

Please cite this article as: I.V. Tasso, G.C. Buscaglia, A finite element method for viscous membranes, *Comput. Methods Appl. Mech. Engrg.* (2012), doi: http://dx.doi.org/10.1016/j.cma.2012.10.021

A finite element method for viscous membranes

Italo V. Tasso[a,b], Gustavo C. Buscaglia[a,b,*]

[a] *Instituto de Ciências Matemáticas e Computação, Universidade de São Paulo, Av. do Trabalhador Sãocarlense 400, 13560-970 São Carlos, SP, Brazil*
[b] *Instituto Nacional de Ciência e Tecnologia em Medicina Assistida por Computação Científica, Brazil*

Abstract

The simulation of biological interfaces at the Living Cell scale relies on membrane models that are a combination of a finite–strain elastic part, typically modeling the contribution of a cytoskeleton, and a viscous part that models the contribution of the lipidic bilayer. The motion of these membranes is driven by a shape-dependent energy, modeled by means of the Canham–Helfrich formula or variants thereof. In this article we review the finite element formulation of elastic membranes, and then extend it so as to deal with the viscous behavior of lipidic bilayers. The resulting numerical method, which is easily implemented on codes developed for solid membranes, is assessed on the simulation of dynamical prolate–to–oblate transitions of simplified red blood cells under tweezing.

Keywords: Viscous membrane, lipidic bilayer, inextensibility, Canham-Helfrich energy, finite elements, Boussinesq-Scriven operator.

1. Introduction

Fluidic behavior is characterized by the impossibility of rest under shear. Both in Nature and in biomedical applications there exist highly–deformable membranes that exhibit fluidic behavior. The most important example is that of lipidic bilayers, which are a basic constituent of the living cell membrane. They consist of two molecular layers of amphiphilic phospholipids, each layer exposing the hydrophilic ends of their molecules to the adjacent water and thus also keeping the hydrophobic ends away from it. The molecules in lipidic membranes exhibit very high tangential mobility, with relatively low layer–to–layer transfer rate. Molecular simulations have greatly improved the understanding of these systems, in particular of their tangential behavior [43, 50, 16, 38]. However, for simulations at the scale of a whole Living Cell to be computable during biologically significant time lapses, continuum models are mandatory. The best

*Corresponding author.
Email addresses: italo@tasso.com.br (Italo V. Tasso), gustavo.buscaglia@icmc.usp.br (Gustavo C. Buscaglia)

candidate model corresponds to a two–dimensional fluid, flowing on a time–dependent, curved surface in three–dimensional space.

The actual rheological behavior of lipidic bilayers is predominantly viscous (i.e.; Newtonian) and area-preserving [47, 32, 34], with a surface viscosity μ of about $5 - 13 \times 10^{-9}$ Pa-s-m[55] that can take higher values, up to 2×10^{-6} Pa-s-m. Though some viscoelasticity may exist, recent rheometrical data suggest that it is not significant [21, 22].

In this work we propose a method for the finite element simulation of viscous membranes. It is strongly based on variational methods that are well established in the field of Solid Mechanics, obtaining the discrete equations by perturbations of the appropriate energy. The presentation begins with a brief review of the finite element treatment of *elastic* membranes, for which details can be found in [24] and biological applications in [41, 27, 52, 30, 28]. In a suitable limit, the elastic operator tends to the viscous operator, to which the zero–tangential–divergence (inextensibility) condition is added to arrive at a realistic approximation of the surface fluidic behavior. The inextensibility condition introduces a Lagrange multiplier field P which plays the role of a non–homogeneous surface tension. A stabilization term proportional to the surface Laplacian of P is added to allow for the use of the same interpolants for all fields. As driving force for the motion, the Canham-Helfrich model [11, 23] is added in order to study shape evolutions typical of biological membranes. Overall, the method seems to be the first to compute truly viscous and inextensible relaxation of membranes in general 3D geometries. This is illustrated by simulating the dynamical response of a (simplified) red blood cell under oscillatory tweezing.

Possible applications of the proposed method are numerous, such as dynamical studies of membrane adhesion [15], conformation [32], stomatocyte-discocyte-echinocyte [31] and other shape transformations [48, 39], among many others.

2. Elastic membranes

2.1. Membrane kinematics

The large–deformation kinematics of membranes is already well known but will be reviewed here as a basis for the extensions made later on. The interested reader is referred, for example, to the detailed articles by Holzapfel *et al* [24] and Bonet *et al* [6].

Consider an open set $\hat{T}$ in $\mathbb{R}^2$, which will typically be a master finite element. The material points of some part of the membrane are associated to $\hat{T}$, which acts as a material configuration. The position, at some instant t, of the material points of the membrane associated to $\hat{T}$ is described by some injective, continuously differentiable function

$$\boldsymbol{\varphi}^t : \hat{T} \ \rightarrow \ \mathbb{R}^3. \tag{1}$$

In the finite element implementation this function is defined elementwise as a linear combination of basis functions on $\hat{T}$, i.e.,

$$\boldsymbol{\varphi}^t(\boldsymbol{\xi}) = \sum_{m=1}^{M} \boldsymbol{X}_{(m)}(t)\ N_{(m)}(\boldsymbol{\xi}) \tag{2}$$

where $\boldsymbol{X}_{(m)}(t)$ is the position, at time t, of the m-th node of the element (we assume a Lagrangian finite element with M nodes for clarity). Notice that $\xi \in \mathbb{R}^2$.

The image of $\hat{T}$ by $\boldsymbol{\varphi}^t$ is denoted here by T^t. Because of the injectivity of $\boldsymbol{\varphi}^t$, it is possible to define the (also of class $\mathcal{C}^1$) inverse mapping

$$\boldsymbol{\psi}^t : T^t \ \rightarrow\ \hat{T}, \qquad \text{such that} \qquad \boldsymbol{\psi}^t(\boldsymbol{\varphi}^t(\boldsymbol{\xi})) = \boldsymbol{\xi} \quad \forall \boldsymbol{\xi} \in \hat{T} \tag{3}$$

Without loss of generality, it is assumed that the relaxed configuration of the membrane corresponds to $t = 0$. The deformation of the membrane is thus characterized by the mapping

$$\boldsymbol{\zeta}^t : T^0 \ \rightarrow\ T^t, \qquad \text{defined by} \qquad \boldsymbol{\zeta}^t(\boldsymbol{x}) := \boldsymbol{\varphi}^t(\boldsymbol{\psi}^0(\boldsymbol{x})) \tag{4}$$

To compute the deformation gradient some additional work is needed, since the gradient of $\boldsymbol{\varphi}^t$ is, with cartesian coordinates for T^t, a 3×2-matrix and thus not invertible (if $\hat{T}$ were an open set in $\mathbb{R}^3$, one would simply compute $\nabla\boldsymbol{\zeta}^t(\boldsymbol{x}) = \nabla\boldsymbol{\varphi}^t(\boldsymbol{\psi}^0(\boldsymbol{x}))\ \left[\nabla\boldsymbol{\varphi}^0(\boldsymbol{\psi}^0(\boldsymbol{x}))\right]^{-1}$).

The tangential deformation gradient is a rank–2 tensor from the tangent plane at $\boldsymbol{x} = \varphi^0(\boldsymbol{\xi})$ to the tangent plane at $\boldsymbol{y} = \zeta^t(\boldsymbol{x}) = \varphi^t(\boldsymbol{\xi})$. Take a cartesian basis $(\check{\boldsymbol{e}}^{(1)}, \check{\boldsymbol{e}}^{(2)})$ at $\hat{T}$, which is nothing but the canonical basis at the master element. Two linearly independent vectors $(\boldsymbol{G}^{(1)}, \boldsymbol{G}^{(2)})$ tangent to T^0 at $\boldsymbol{x}$ are defined as the infinitesimal images of the basis:

$$\boldsymbol{\varphi}^0(\boldsymbol{\xi} + \epsilon_1\,\check{\boldsymbol{e}}^{(1)} + \epsilon_2\,\check{\boldsymbol{e}}^{(2)}) = \boldsymbol{\varphi}^0(\boldsymbol{\xi}) + \epsilon_1\,\boldsymbol{G}^{(1)} + \epsilon_2\,\boldsymbol{G}^{(2)} + \mathcal{O}(\epsilon_1^2 + \epsilon_2^2) \tag{5}$$

and are calculated in cartesian components as

$$\left\{\boldsymbol{G}^{(i)}\right\}_j \ =\ \frac{\partial \varphi_j^0}{\partial \xi_i} \ =\ \sum_{m=1}^{M} X_{(m)j}(0)\ \frac{\partial N_{(m)}}{\partial \xi_i} \qquad i = 1, 2; \quad j = 1, 2, 3 \tag{6}$$

the normal $\check{\boldsymbol{N}}$ to T^0 at $\boldsymbol{x}$ is given by

$$\check{\boldsymbol{N}} = \frac{\boldsymbol{G}^{(1)} \times \boldsymbol{G}^{(2)}}{\|\boldsymbol{G}^{(1)} \times \boldsymbol{G}^{(2)}\|} \tag{7}$$

thus defining an orthonormal basis of $\mathbb{R}^3$ such that its two first vectors are tangent to T^0 (and thus a basis of the tangent plane) as

$$\boldsymbol{V}^{(1)} = \frac{\boldsymbol{G}^{(1)}}{\|\boldsymbol{G}^{(1)}\|}, \qquad \boldsymbol{V}^{(2)} = \check{\boldsymbol{N}} \times \boldsymbol{V}^{(1)}, \qquad \boldsymbol{V}^{(3)} = \check{\boldsymbol{N}} \tag{8}$$

Analogous vectors can be defined at $\boldsymbol{y}$ by replacing $\boldsymbol{\varphi}^0$ with $\boldsymbol{\varphi}^t$ in the previous definitions. They will be denoted by lowercase letters: $\boldsymbol{g}^{(1)}$, $\boldsymbol{g}^{(2)}$, $\check{\boldsymbol{n}}$, $\boldsymbol{v}^{(1)}$, $\boldsymbol{v}^{(2)}$ and $\boldsymbol{v}^{(3)}$.

The tangential deformation gradient is thus given by the infinitesimal deformation (by $\boldsymbol{\zeta}^t$) of the vectors $\boldsymbol{V}^{(1)}$ and $\boldsymbol{V}^{(2)}$, which is given by

$$\boldsymbol{\zeta}^t(\boldsymbol{x}+\epsilon_1\boldsymbol{V}^{(1)}+\epsilon_2\boldsymbol{V}^{(2)}) \;=\; \boldsymbol{\zeta}^t(\boldsymbol{x})+(\mathcal{F}_{11}\epsilon_1+\mathcal{F}_{12}\epsilon_2)\,\boldsymbol{v}^{(1)}+(\mathcal{F}_{21}\epsilon_1+\mathcal{F}_{22}\epsilon_2)\,\boldsymbol{v}^{(2)}+\mathcal{O}(\epsilon_1^2+\epsilon_2^2) \tag{9}$$

where the matrix $\underline{\underline{\mathcal{F}}}$ is given by

$$\mathcal{F}_{11} \;=\; \frac{\|\boldsymbol{g}^{(1)}\|}{\|\boldsymbol{G}^{(1)}\|} \tag{10}$$

$$\mathcal{F}_{12} \;=\; \frac{\boldsymbol{g}^{(2)}\cdot\boldsymbol{v}^{(1)}}{\boldsymbol{G}^{(2)}\cdot\boldsymbol{V}^{(2)}} \;-\; \frac{\|\boldsymbol{g}^{(1)}\|}{\|\boldsymbol{G}^{(1)}\|}\,\frac{\boldsymbol{G}^{(2)}\cdot\boldsymbol{V}^{(1)}}{\boldsymbol{G}^{(2)}\cdot\boldsymbol{V}^{(2)}} \tag{11}$$

$$\mathcal{F}_{21} \;=\; 0 \tag{12}$$

$$\mathcal{F}_{22} \;=\; \frac{\boldsymbol{g}^{(2)}\cdot\boldsymbol{v}^{(2)}}{\boldsymbol{G}^{(2)}\cdot\boldsymbol{V}^{(2)}} \tag{13}$$

Remark 2.1. *In fact, the left-hand side of (9) should read*

$$\boldsymbol{\zeta}^t\left(\boldsymbol{\Pi}_0(\boldsymbol{x}+\epsilon_1\boldsymbol{V}^{(1)}+\epsilon_2\boldsymbol{V}^{(2)})\right)$$

where $\boldsymbol{\Pi}_0:\mathbb{R}^3\to T^0$ is the closest-point projection onto T^0, because $\boldsymbol{x}+\epsilon_1\boldsymbol{V}^{(1)}+\epsilon_2\boldsymbol{V}^{(2)}$ does not belong to T^0 and thus $\boldsymbol{\zeta}^t$ is not defined at it.

Equation (9) defines the deformation–gradient tensor $\mathbb{F}^t$ which maps de tangent plane at $\boldsymbol{x}\in T^0$ onto the tangent plane at $\boldsymbol{y}\in T^t$ as the only linear operator satisfying, for all tangent vectors $\boldsymbol{t}$,

$$\boldsymbol{\zeta}^t\left(\boldsymbol{\Pi}_0(\boldsymbol{x}+\epsilon\,\boldsymbol{t})\right) \;=\; \boldsymbol{y} \;+\; \epsilon\,\mathbb{F}^t\,\boldsymbol{t} \;+\; \mathcal{O}(\epsilon^2) \tag{14}$$

To prove that (14) indeed holds, one starts from the identities (true by construction)

$$\boldsymbol{g}^{(i)} = \mathbb{F}^t\,\boldsymbol{G}^{(i)} \qquad\qquad i=1,2 \tag{15}$$

so that, since $\mathcal{F}_{11}$ is, from (9), equal to

$$\mathcal{F}_{11} = \boldsymbol{v}^{(1)}\cdot\mathbb{F}^t\,\boldsymbol{V}^{(1)}$$

it results that

$$\mathcal{F}_{11} = \boldsymbol{v}^{(1)}\cdot\mathbb{F}^t\,\boldsymbol{V}^{(1)} = \frac{\boldsymbol{g}^{(1)}}{\|\boldsymbol{g}^{(1)}\|}\cdot\mathbb{F}^t\left(\frac{\boldsymbol{G}^{(1)}}{\|\boldsymbol{G}^{(1)}\|}\right) = \frac{\boldsymbol{g}^{(1)}\cdot\mathbb{F}^t\,\boldsymbol{G}^{(1)}}{\|\boldsymbol{g}^{(1)}\|\,\|\boldsymbol{G}^{(1)}\|} = \frac{\|\boldsymbol{g}^{(1)}\|}{\|\boldsymbol{G}^{(1)}\|}$$

From the identity $\boldsymbol{G}^{(2)} = \boldsymbol{G}^{(2)}\cdot\boldsymbol{V}^{(1)}\;\boldsymbol{V}^{(1)} + \boldsymbol{G}^{(2)}\cdot\boldsymbol{V}^{(2)}\;\boldsymbol{V}^{(2)}$, applying $\mathbb{F}^t$ to both sides and rearranging, we obtain

$$\mathbb{F}^t\,\boldsymbol{V}^{(2)} = \frac{1}{\boldsymbol{G}^{(2)}\cdot\boldsymbol{V}^{(2)}}\left(\boldsymbol{g}^{(2)} \;-\; \frac{\boldsymbol{G}^{(2)}\cdot\boldsymbol{V}^{(1)}}{\|\boldsymbol{G}^{(1)}\|}\,\boldsymbol{g}^{(1)}\right)$$

and this yields

$$\mathcal{F}_{12} = \boldsymbol{v}^{(1)} \cdot \mathbb{F}^t \, \boldsymbol{V}^{(2)} = \frac{\boldsymbol{v}^{(1)} \cdot \boldsymbol{g}^{(2)}}{\boldsymbol{G}^{(2)} \cdot \boldsymbol{V}^{(2)}} - \frac{\boldsymbol{G}^{(2)} \cdot \boldsymbol{V}^{(1)}}{\boldsymbol{G}^{(2)} \cdot \boldsymbol{V}^{(2)}} \, \frac{\|\boldsymbol{g}^{(1)}\|}{\|\boldsymbol{G}^{(1)}\|}$$

The other two components of $\underline{\underline{\mathcal{F}}}$ are obtained similarly.

The right tangential Cauchy-Green tensor $\mathbb{C}^t = (\mathbb{F}^t)^T \mathbb{F}^t$, expressed in the basis $(\boldsymbol{V}^{(1)}, \boldsymbol{V}^{(2)})$, has components that are straightforward to calculate,

$$\underline{\underline{\mathcal{C}}}^t \;=\; \underline{\underline{\mathcal{F}}}^T \, \underline{\underline{\mathcal{F}}} \;=\; \begin{pmatrix} \mathcal{F}_{11}^2 & \mathcal{F}_{11}\mathcal{F}_{12} \\ \mathcal{F}_{11}\mathcal{F}_{12} & \mathcal{F}_{12}^2 + \mathcal{F}_{22}^2 \end{pmatrix} \tag{16}$$

and the Green–Saint Venant tensor $\mathbb{E}^t = (\mathbb{C}^t - \mathbb{I})/2$, in the same basis, is given by the matrix

$$\underline{\underline{\mathcal{E}}}^t = \frac{1}{2} \, \left(\underline{\underline{\mathcal{C}}}^t - \underline{\underline{\mathcal{I}}} \right) \tag{17}$$

with $\underline{\underline{\mathcal{I}}}$ the identity matrix. The energy density at $\boldsymbol{y} = \boldsymbol{\zeta}^t(\boldsymbol{x})$ of an isotropic elastic material is a function of the invariants of $\underline{\underline{\mathcal{C}}}^t$ (such as its eigenvalues, λ_1^2 and λ_2^2, always positive) or of $\underline{\underline{\mathcal{E}}}^t$ (with eigenvalues $\frac{1}{2}(\lambda_1^2 - 1)$ and $\frac{1}{2}(\lambda_2^2 - 1)$). If the deformation preserves area, then $det(\underline{\underline{\mathcal{C}}}^t) = \lambda_1^2 \, \lambda_2^2 = 1$. Also notice that the first invariant (the trace) is given by $tr(\underline{\underline{\mathcal{C}}}^t) = \lambda_1^2 + \lambda_2^2$.

2.2. Computing the elastic energy

From the previous section one can extract a systematic procedure to compute the elastic energy of an isotropic material. To do this with the minimal notation effort, let us first restrict the class of transformations to those depending linearly on a set of vector coefficients $\underline{\boldsymbol{X}}$, which for Lagrangian finite elements are in fact the positions of the nodes; i.e., which can be written as

$$\boldsymbol{\varphi}(\boldsymbol{\xi}) = \sum_{m=1}^{M} \boldsymbol{X}_{(m)} \, N_{(m)}(\boldsymbol{\xi}) \tag{18}$$

For Lagrangian finite elements, clearly, $\{N_{(m)}\}$ is the set of basis functions on the master (unit) element. In this way, since the elastic energy E_e depends on both the relaxed configuration $\underline{\boldsymbol{Y}}$ (that we adopted arbitrarily as corresponding to $t = 0$, i.e., $\underline{\boldsymbol{Y}} = \underline{\boldsymbol{X}}^0$) and the current configuration (time t), we write it as

$$E_e(\underline{\boldsymbol{Y}}, \underline{\boldsymbol{X}}^t) = \int_{\Gamma(t)} \rho \, e_e \; d\Gamma \tag{19}$$

where $\Gamma(t)$ is the surface occupied by the membrane at time t and ρ the surface mass density. For an isotropic material the dependence above reduces to a simple formula involving the eigenvalues of $\mathbb{C}^t$. For example, for a Mooney–Rivlin material,

$$e_e = c_1 \, (\lambda_1^2 + \lambda_2^2 - 2) \; + \; c_2 \, (\lambda_1^2 \lambda_2^2 - 1) \tag{20}$$

Given a relaxed configuration $\underline{\boldsymbol{Y}}$ and a current configuration $\underline{\boldsymbol{X}}^t$, we proceed to evaluate e_e at point $\boldsymbol{\xi}$ of the master element as follows:

Step 1: Compute the four vectors

$$\boldsymbol{G}^{(1)} = \sum_{m=1}^{M} \boldsymbol{Y}_{(m)} \frac{\partial N_{(m)}}{\partial \xi_1}(\boldsymbol{\xi}) \tag{21}$$

$$\boldsymbol{G}^{(2)} = \sum_{m=1}^{M} \boldsymbol{Y}_{(m)} \frac{\partial N_{(m)}}{\partial \xi_2}(\boldsymbol{\xi}) \tag{22}$$

$$\boldsymbol{g}^{(1)} = \sum_{m=1}^{M} \boldsymbol{X}^t_{(m)} \frac{\partial N_{(m)}}{\partial \xi_1}(\boldsymbol{\xi}) \tag{23}$$

$$\boldsymbol{g}^{(2)} = \sum_{m=1}^{M} \boldsymbol{X}^t_{(m)} \frac{\partial N_{(m)}}{\partial \xi_2}(\boldsymbol{\xi}) \tag{24}$$

and the normals

$$\check{\boldsymbol{N}} = \frac{\boldsymbol{G}^{(1)} \times \boldsymbol{G}^{(2)}}{\|\boldsymbol{G}^{(1)} \times \boldsymbol{G}^{(2)}\|} \; ; \qquad \check{\boldsymbol{n}} = \frac{\boldsymbol{g}^{(1)} \times \boldsymbol{g}^{(2)}}{\|\boldsymbol{g}^{(1)} \times \boldsymbol{g}^{(2)}\|} \tag{25}$$

Step 2: Compute the right Cauchy–Green tensor as the 2×2-matrix

$$\underline{\underline{\mathcal{C}}}^t = \underline{\underline{\mathcal{F}}}^T \underline{\underline{\mathcal{F}}} \tag{26}$$

with $\underline{\underline{\mathcal{F}}}$ given by (10)-(13).

Step 3: Compute the eigenvalues of the right Cauchy–Green tensor and then the energy e_e using the constitutive law, for example (20). This gives the elastic energy density at point $\boldsymbol{\xi}$, which is then integrated over the element using a quadrature rule and then summed over elements to obtain E_e. Since the energy only involves first derivatives of the deformation, no contribution to the energy arises at the interelement boundaries, as long as one works with $\mathcal{C}^0$ finite elements.

2.3. Elastic membrane dynamics: Virtual work principle

Considering just the elasticity of the membrane and its inertia, the virtual work principle states that the virtual power of the internal forces equals the virtual power of the acceleration. Since the internal forces are elastic, their virtual power equals minus the derivative of the elastic energy with respect to the geometrical degrees of freedom $\underline{\boldsymbol{X}}$, that is

$$\lim_{\epsilon \to 0} \frac{E_e(\underline{\boldsymbol{X}}^0, \underline{\boldsymbol{X}}^t + \epsilon \underline{\boldsymbol{w}}) - E_e(\underline{\boldsymbol{X}}^0, \underline{\boldsymbol{X}}^t)}{\epsilon} + \int_{\Gamma(t)} \rho \boldsymbol{a} \cdot \boldsymbol{w} \, d\Gamma = 0 \tag{27}$$

where the configuration $\underline{\boldsymbol{X}^t + \epsilon \boldsymbol{w}}$ is defined by perturbing the position of the interface along the virtual velocity field $\boldsymbol{w}$ (in Lagrange elements this is typically done one node at a time) and $\boldsymbol{a}$ is the acceleration.

Of course the first term of (27) may be represented as a distribution of forces on the membrane, but this would require the introduction of more involved differential operators. We thus prefer, for clarity, to leave the computation of the elastic term as the limit of finite differences. Notice, however, that (27) is the classical momentum equation for the membrane, and that it can be replaced by any equivalent equivalent equation as obtained, for example, from a formulation based on forces or on tangential stress tensors.

We emphasize that (27) is a non–linear system of ordinary differential equations for the unknowns $\underline{\boldsymbol{X}}^t$, since the correspondence $\boldsymbol{\varphi}^t \leftrightarrow \underline{\boldsymbol{X}}^t$ is one–to–one and the nodal accelerations are given by $d^2\underline{\boldsymbol{X}}^t/dt^2$. To build the $3 \times M$ equations, the virtual velocity field $\boldsymbol{w}$ is, as usual, successively taken as

$$\boldsymbol{w} = \check{\boldsymbol{e}}_i \, N_{(m)}, \qquad i = 1, 2, 3; \qquad m = 1, \ldots, M \tag{28}$$

so that, remembering (2) and defining

$$\underline{\boldsymbol{D}}_{(i,m)} = \begin{cases} \check{\boldsymbol{e}}_i & \text{for node } m \\ 0 & \text{for nodes other than } m \end{cases} \tag{29}$$

one arrives at the system of equations (for all nodes m and coordinates i)

$$\lim_{\epsilon \to 0} \frac{E_e(\underline{\boldsymbol{X}}^0, \underline{\boldsymbol{X}}^t + \epsilon \, \underline{\boldsymbol{D}}_{(i,m)}) - E_e(\underline{\boldsymbol{X}}^0, \underline{\boldsymbol{X}}^t)}{\epsilon} + \sum_{\ell=1}^{M} \left[\int_{\Gamma(t)} \rho \, N_{(\ell)} \, N_{(m)} \, d\Gamma \right] \frac{d^2 X^t_{(\ell)i}}{dt^2} \; = \; 0 \tag{30}$$

where $X^t_{(\ell)i}$ is the i-th coordinate, at time t, of node ℓ, always considering Lagrangian elements. The bracketed term is the usual mass matrix on $\Gamma(t)$, which will be denoted by $\underline{\underline{A}}$, with components $A_{k\ell}$. In the above, the basis functions $N_{(m)}$ are as usual evaluated not at $\boldsymbol{x} \in \Gamma(t)$ but at $\boldsymbol{\psi}^t(\boldsymbol{x}) \in \hat{T}$ (the master element), as is common finite element practice. If the mass of the membrane is conserved ($\rho \, d\Gamma$ independent of t) the mass matrix does not depend on time.

2.4. Effect of the adjacent fluid

In general, to consider the full effect of the adjacent fluid one needs to solve the 3D Navier–Stokes equations coupled to (30), in which the forces exerted by the fluid need to be added. If fluid inertia is neglected, the boundary element formulation based on fundamental solutions of the Stokes equations has gained much popularity [41, 27, 52, 30]. Its advantage is that all computations are performed on the interface Γ, with no need of a volumetric mesh. We however prefer to leave this issue outside the scope of this article, which for viscous membranes can be physically justified if the size is smaller than 10 microns, as discussed later. We thus limit the discussion to the incorporation of gravitational energy and pressurization effects.

Let $\mathcal{V}(\underline{\boldsymbol{X}}^t)$ be the volume enclosed by the (closed) surface $\Gamma(t)$,

$$\mathcal{V}(\underline{\boldsymbol{X}}^t) = \int_{\Gamma(t)} \frac{1}{3} \boldsymbol{x} \cdot \check{\boldsymbol{n}} \, d\Gamma \tag{31}$$

and assume it contains a fluid of constant density ρ_ℓ and that gravity is also constant (equal to $-g\check{\boldsymbol{e}}_3$). Then the gravitational energy of the membrane together with that of the enclosed fluid is given by

$$E_{\text{grav}} = \int_{\Gamma(t)} \left(\rho\, x_3 \,+\, \rho_\ell \,\frac{x_3^2}{2}\, n_3 \right) g \;\, d\Gamma \tag{32}$$

Further, assume that the internal fluid is pressurized to some pressure $p(t)$, assumed known. Since the work of this pressure upon a virtual change in volume is $p(t)\,\Delta\mathcal{V}$, for the purpose of virtual work calculations there appears the term $p(t)\,\mathcal{V}(\underline{\boldsymbol{X}}^t)$.

Remark 2.2. *If the volume enclosed by the membrane is imposed, for example because the inner fluid is incompressible, then the internal pressure $p(t)$ becomes an unknown (Lagrange multiplier). Denoting by $\overline{\mathcal{V}}(t)$ the imposed value, the additional equation is*

$$\mathcal{V}(\underline{\boldsymbol{X}}^t) = \overline{\mathcal{V}}(t) \tag{33}$$

This can be incorporated into the energy as the term

$$p(t)\,[\mathcal{V}(\underline{\boldsymbol{X}}^t) - \overline{\mathcal{V}}(t)] \tag{34}$$

which does not change the dependence on $\underline{\boldsymbol{X}}^t$ but allows one to recover (33) by simply taking variations with respect to p.

2.5. Obstacles and contact

The presence of unilateral constraints such as solid obstacles can also be accounted for in the energy. In this work we adopt a simple penalty approach of the form

$$E_{\text{con}} = \int_{\Gamma(t)} \frac{\alpha_{\text{con}}}{2}\delta^2 \; d\Gamma \tag{35}$$

where δ is the *penetration depth* of the membrane into the solid obstacle.

The contact of the membrane with itself (self penetration) is dealt with in the same way. Further details of the contact implementation are left out for the sake of brevity, since in fact no contact takes place in the viscous membranes examples that are the focus of this work.

2.6. An algorithm for elastic membranes

Gathering the results of the previous sections it is possible to build a complete algorithm for the simulation of elastic membranes. For nodal positions $\underline{\boldsymbol{X}}$ of the membrane, we consider the energy

$$\mathcal{E} = E_e(\underline{\boldsymbol{Y}}, \underline{\boldsymbol{X}}) + E_{\text{grav}}(\underline{\boldsymbol{X}}) + E_{\text{con}}(\underline{\boldsymbol{X}}) + p\,[\mathcal{V}(\underline{\boldsymbol{X}}) - \overline{\mathcal{V}}] \tag{36}$$

and assume $\overline{\mathcal{V}}$ to be a given function of time, so that $p(t)$ is unknown (the case in which $p(t)$ is given is easier). The relaxed configuration $\underline{\boldsymbol{Y}}$ is given *and fixed*,

though not necessarily equal to the (given) initial configuration $\underline{\boldsymbol{X}}^0$. Also given is the initial nodal-velocity vector $\underline{\boldsymbol{U}}^0$.

Continuous-in-time discrete problem: For $t > 0$, find $(\underline{\boldsymbol{X}}^t, \underline{\boldsymbol{U}}^t, p(t))$ such that for all i and m,

$$\frac{d\underline{\boldsymbol{X}}^t}{dt} - \underline{\boldsymbol{U}}^t = 0 \tag{37}$$

$$\frac{\mathcal{E}(\underline{\boldsymbol{X}}^t + \epsilon \underline{\boldsymbol{D}}_{(i,m)}, p(t)) - \mathcal{E}(\underline{\boldsymbol{X}}^t, p(t))}{\epsilon} + \sum_{\ell=1}^{M} A_{m\ell} \frac{dU^t_{(\ell)i}}{dt} = 0 \tag{38}$$

$$\mathcal{V}(\underline{\boldsymbol{X}}^t) - \overline{\mathcal{V}} = 0 \tag{39}$$

where ϵ is an algorithmic parameter chosen so as to provide a good approximation to (30). Typically we choose $\epsilon = h/100$, with h the mesh size. The equations above constitutes a system of $6M + 1$ differential-algebraic equations that can be solved by standard methods. Notice that the only algebraic equation is (39).

2.7. Time discretization

For the purpose of illustrating the method, we implemented two numerical schemes for (37)-(39), an explicit fourth-order Runge-Kutta scheme and a fully-implicit backward Euler scheme.

Runge-Kutta scheme: Let $\mathbb{X} = (\underline{\boldsymbol{X}}, \underline{\boldsymbol{U}})$ be the array of nodal positions and velocities, and let $\mathbb{X}^n$ and p^n be the already calculated unknowns at time t_n. Then, for each tentative value p^* of the pressure at time t_{n+1}, and by performing one time step of the Runge-Kutta scheme for equations (37)-(38), one obtains a different $\mathbb{X}^{n+1}$. Formally, this can be written as

$$\mathbb{X}^{n+1}(p^*) = (\underline{\boldsymbol{X}}^{n+1}(p^*), \underline{\boldsymbol{U}}^{n+1}(p^*)) \tag{40}$$

The correct value of p^{n+1} is thus obtained by solving the volume-constraint equation

$$\mathcal{V}(\underline{\boldsymbol{X}}^{n+1}(p^{n+1})) = \overline{\mathcal{V}}(t_{n+1}) \tag{41}$$

A regula-falsi method is used to efficiently solve this scalar equation at each time step.

Backward-Euler scheme: In this case we simply discretize the time derivatives by backward differences transforming (37)-(39) into a nonlinear algebraic system. This system is then solved by Newton-Raphson iterations with a finite-difference evaluation of the Jacobian matrix.

2.8. A numerical example

Figure 1 shows a fluid-filled membrane under gravity interacting with several cylindrical obstacles. The initial radius is $R = 1$, the membrane mass is $M = 1$ and the density of the internal fluid is $\rho_f = 1$, under unit gravity $g = 1$. The material is Hookean with shear modulus $G = 1$ and bulk modulus $K = 2$. A small dissipative force, $\boldsymbol{f} = -10\,\boldsymbol{U}$, is added to dampen out standing oscillations. The time step is $\Delta t = 10^{-3}$

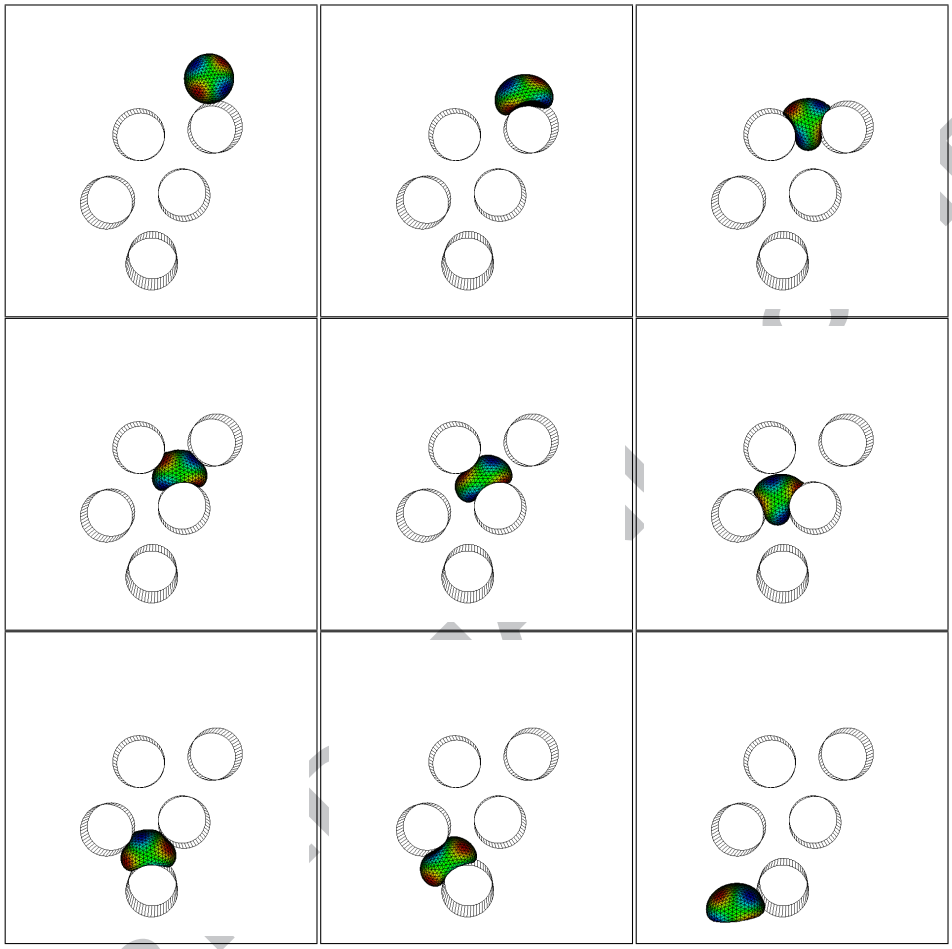

Figure 1: Elastic membrane under gravity interacting with solid obstacles.

3. Viscous membranes

3.1. Surface viscous behavior

A classical derivation of the viscous operator from conservation principles can be found in the pioneering article by Scriven [45]. A geometric form of the operator in the language of differential forms is derived in the article by Arroyo and DeSimone [1], with interesting two–dimensional examples of budding. The resulting equations turn out to be quite involved, with the additional complication of depending explicitly on the hard–to–discretize curvature tensor. As a consequence, no Continuum–Mechanics–based algorithm is available for the dynamic simulation of three–dimensional surface–viscous behavior. Most published methodologies focus on the obtention of the equilibrium state by gradient

flow[19, 33, 17, 4, 18, 7], disregarding the relaxation dynamics. Others compute the damping effect of the surrounding fluid but neglect the surface viscous effects[8, 44].

Surface viscous dissipation is nevertheless dominant for small enough membranes, more precisely for membranes smaller than the Saffman–Delbrück length[42]

$$\ell_{SD} = \frac{\mu}{\mu_b}$$

where μ is the membrane's surface viscosity and μ_b the viscosity of the bulk fluid. The dominance of surface dissipation over bulk-fluid dissipation for sizes smaller than ℓ_{SD} has been argued by Arroyo & DeSimone [1] and confirmed by Arroyo *et al* [2] by means of axisymmetric numerical simulations. Considering $\mu = 10^{-8}$ Pa-s-m [55] and $\mu_b = 10^{-3}$ Pa-s, one obtains $\ell_{SD} = 10^{-5}$ m, or 10 microns. Typical sizes of red blood cells, for example, are thus smaller than ℓ_{SD}, urging for surface viscous effects to be accounted for in simulations.

The approach we propose here is based on Maxwell's idea of viewing viscosity as "fugitive elasticity" [36, 35]. It is best explained considering an elastic spring for which the force depends on the current position x, on the elastic constant k and on the equilibrium position a as

$$F(k, a; x) = -k(x - a) \tag{42}$$

Defining the energy $E(k, a; x) = \frac{1}{2}k(x - a)^2$ it is readily seen that

$$F = -\lim_{\delta \to 0} \frac{E(k, a; x + \delta) - E(k, a; x)}{\delta} \tag{43}$$

so that the forces can be computed using (43) for some sufficiently small δ. Now assume that one wants to compute a *viscous* dashpot, i.e.; an element satisfying

$$F = -\mu\, v, \tag{44}$$

where $v = dx/dt$ is the velocity, using the code developped for the elastic spring above. Consider the elastic energy that results from taking the equilibrium position as $a = x - v\tau$, for some small time interval τ, and with elastic constant $k = \mu/\tau$. Substituting into (43) one obtains

$$F = -\lim_{\delta \to 0} \frac{E\left(\frac{\mu}{\tau}, x - v\tau; x + \delta\right) - E\left(\frac{\mu}{\tau}, x - v\tau; x\right)}{\delta} = -\mu\, v \tag{45}$$

which shows that the viscous behavior of the dashpot can be reproduced as the behavior of a *virtual spring* which has relaxed configuration $x - v\,\tau$ and elastic constant μ/τ, for τ sufficiently small. Notice that, in (45), the x appearing in the equilibrium position $a = x - v\tau$ is *not* perturbed when evaluating F. Applying this idea in an implicit dynamic algorithm, the relaxed configuration of the virtual spring at time t_{n+1} would be $x_{n+1} - v_{n+1}\,\tau$.

Remark 3.1. *A related procedure was used by Ma and Klug to regularize mesh movements [33]. They determined the tangential displacements by solving an elastic problem in which the relaxed configuration is continuously updated. Besides their goal being preserving mesh quality and not simulating surface viscosity, their algorithm differs from the one proposed here in that their relaxed configuration is simply the last computed state x_n.*

The previous idea, generalized to approximate the viscous operator on a membrane, is thus given by

$$\mathcal{A}(\boldsymbol{X}^t, \boldsymbol{U}^t; \boldsymbol{w}) \simeq \frac{E_e(\underline{\boldsymbol{X}^t - \tau \boldsymbol{U}^t}, \underline{\boldsymbol{X}^t + \epsilon\, \boldsymbol{w}}) - E_e(\underline{\boldsymbol{X}^t - \tau \boldsymbol{U}^t}, \underline{\boldsymbol{X}^t})}{\epsilon} \tag{46}$$

where the elastic energy must correspond to a material having shear modulus $G = \mu/\tau$. Notice that when embedding the operator above into a time–stepping scheme it is possible to take $\tau = \Delta t$ and thus $\boldsymbol{X}^t - \tau \boldsymbol{U}^t \simeq \boldsymbol{X}^{t-\Delta t}$, which amounts to taking the previously computed configuration as reference configuration. We however leave τ as a free parameter to be tuned later on. This also emphasizes the velocity dependence of the viscous term, which would be hidden into $\boldsymbol{X}^{t-\Delta t}$ otherwise.

3.2. The inextensibility condition

It is well–established that lipidic bilayers tend to be *inextensible*, in the sense that the area of every part of the surface is preserved under deformation. In most simulations up to now this condition has only been imposed *globally* [19, 33, 17, 7], requiring that the total area is preserved.

Here we enforce (weakly) the inextensibility condition throughout the surface. This has also been performed recently by Veerapaneni *et al* [53] in a spectral formulation and by Boedec *et al* [5] in an unstructured boundary element formulation. These authors, however, did not consider surface viscosity effects.

A suitable weak form of the inextensibility condition is

$$\int_{\Gamma(t)} Q\,(d\Gamma - d\Gamma_{\boldsymbol{Y}}) = 0 \qquad\qquad \forall\, Q \,\in\, L^2(\Gamma(t)) \tag{47}$$

where $d\Gamma_{\boldsymbol{Y}}$ is the area differential in the reference configuration $\underline{\boldsymbol{Y}}$. This constraint defines a Lagrange–multiplier field P^t on $\Gamma(t)$, which plays the role of an unknown surface tension and whose virtual work must be accounted for in the momentum equation. Once again, this can be incorporated into the global energy from which the forces are computed by perturbation. The term to be added is linear in P^t, reads

$$\mathcal{B}(\underline{\boldsymbol{Y}}, \underline{\boldsymbol{X}}^t; P^t) = -\int_{\Gamma(t)} P^t\,(d\Gamma - d\Gamma_{\boldsymbol{Y}}) \tag{48}$$

and simply consists of the difference between the integrals P^t over the current and reference configurations (keeping values at the nodes fixed).

3.3. Stabilization of the inextensibility constraint

Equation (47) is equivalent to

$$\int_{\Gamma(t)} Q \, \mathrm{div}_\Gamma \boldsymbol{u} \; d\Gamma = 0 \tag{49}$$

where div_Γ is the *surface divergence* operator and $\boldsymbol{u}$ the Eulerian velocity field. In simple words, inextensibility is nothing but two–dimensional incompressibility of the membrane material considered as a surface fluid. It is thus not surprising that, when discretizing (47) with finite elements of the same polynomial degree for $\boldsymbol{X}$, $\boldsymbol{U}$ and P, violent oscillations appear in P indicative of spurious modes. To stabilize the formulation, we add a surface diffusion term for P which turns (47) into

$$\int_{\Gamma(t)} Q \, (d\Gamma - d\Gamma_{\boldsymbol{Y}}) + \int_{\Gamma(t)} \zeta \, \nabla_\Gamma P \cdot \nabla_\Gamma Q \; d\Gamma = 0 \tag{50}$$

where ζ is a mesh–dependent parameter taken as

$$\zeta = c_\zeta \frac{h^2}{\mu} \tag{51}$$

from previous experience with finite element methods for Navier-Stokes equations[25, 12, 10, 13]. The algorithmic constant c_ζ is chosen as 1, which works well in all tested cases. The *surface gradient* of a scalar field Q, denoted by $\nabla_\Gamma Q$, is the 3D vector

$$\nabla_\Gamma Q = \mathbb{P} \, \nabla \tilde{Q} \tag{52}$$

where $\mathbb{P} = \mathbb{I} - \check{\boldsymbol{n}} \otimes \check{\boldsymbol{n}}$ is the tensor that projects onto the tangent plane and $\tilde{Q}$ is *any* extension of Q to a 3D neighborhood of Γ. The surface gradient of a vector field is defined by applying the same formula componentwise. More details about these surface differential operators can be found in [46, 51, 40, 7, 9]. They are vectors and tensors given in the global Cartesian basis of $\mathbb{R}^3$ and are thus easy to compute and manipulate.

3.4. Bending energy of viscous membranes

It is quite intuitive that inertial and gravitational forces do not play a significant role in the biological interfaces modeled as viscous membranes. This leaves the model presented up to now with no driving force to generate motion, making any initial configuration a trivial solution.

Let us thus add a bending component to the previous ingredients, so that the membrane will be driven towards minimizers of the bending energy. The most accepted model is the Canham–Helfrich energy[11, 23], with variants introduced by Seifert and coworkers [47]. The minima of this energy reproduce with astonishing realism the shapes encountered in living cells and lipidic vesicles [32, 48, 31, 54, 47].

In its simplest form the Canham-Helfrich energy is given by

$$E_{CH} = \frac{C_{CH}}{2} \int_{\Gamma(t)} H^2 \; d\Gamma \tag{53}$$

with H the mean curvature of the surface and C_{CH} a dimensional constant.

It is not obvious how to discretize (53), since a convergent approximation for H in $L^2(\Gamma(t))$ must be devised. We refer to the literature[3, 19, 33, 17, 7] for some options. In this work, since the emphasis is on the surface viscous and incompressibility operators, we adopted the simple discretization of H proposed by Meyer et al [37], leaving further improvements on this topic for later work.

3.5. An algorithm for viscous membranes

Summing up the several contributions defined in the previous sections and neglecting inertial and gravitational terms, the algorithm for viscous membranes reads:

For $\underline{\boldsymbol{X}}^n$ given, compute $\underline{\boldsymbol{X}}^{n+1}$, $\underline{\boldsymbol{U}}^{n+1}$, p^{n+1} and P^{n+1} satisfying, for all i, m and Q:

$$\frac{\underline{\boldsymbol{X}}^{n+1} - \underline{\boldsymbol{X}}^n}{\Delta t} - \underline{\boldsymbol{U}}^{n+1} = 0 \quad (54)$$

$$\frac{\mathcal{E}(\underline{\boldsymbol{Y}}, \underline{\boldsymbol{X}}^{n+1} + \epsilon \underline{\boldsymbol{D}}_{(i,m)}, p^{n+1}, P^{n+1}) - \mathcal{E}(\underline{\boldsymbol{Y}}, \underline{\boldsymbol{X}}^{n+1}, p^{n+1}, P^{n+1})}{\epsilon} = 0 \quad (55)$$

$$\mathcal{V}(\underline{\boldsymbol{X}}^{n+1}) - \overline{\mathcal{V}}(t_{n+1}) = 0 \quad (56)$$

$$\mathcal{B}(\underline{\boldsymbol{Y}}, \underline{\boldsymbol{X}}^{n+1}; Q) + \int_{\Gamma^{n+1}} \zeta \, \nabla_\Gamma P^{n+1} \cdot \nabla_\Gamma Q \, d\Gamma = 0 \quad (57)$$

where $\underline{\boldsymbol{Y}} = \underline{\boldsymbol{X}}^{n+1} - \tau \underline{\boldsymbol{U}}^{n+1}$ and

$$\mathcal{E}(\underline{\boldsymbol{Y}}, \underline{\boldsymbol{X}}, p, P) = E_e(\underline{\boldsymbol{Y}}, \underline{\boldsymbol{X}}) + E_{CH}(\underline{\boldsymbol{X}}) - p[\mathcal{V}(\underline{\boldsymbol{X}}) - \overline{\mathcal{V}}(t)] + \mathcal{B}(\underline{\boldsymbol{Y}}, \underline{\boldsymbol{X}}, P) \quad (58)$$

This variational formulation is discretized with $\mathcal{P}_1$ conforming finite elements for the fields $\boldsymbol{X}$, $\boldsymbol{U}$ and P. Only initial conditions for $\boldsymbol{X}$ are needed, since no time derivatives of $\boldsymbol{U}$, P or p appear. The numerical parameter ϵ is chosen as $0.01\,h$, while τ is chosen such that $\tau \|\boldsymbol{U}\|_{\max} < 0.01\,h$, where the maximum is taken ovel all nodes of the mesh. The one–sided finite difference in (55) is in practice replaced by a fourth–order centered finite difference, yielding a more reliable scheme. The resulting non–linear system is solved all at once by means of a Newton–Raphson algorithm with a finite–difference approximation of the Jacobian matrix.

4. Numerical examples

4.1. Cylinder under traction

Consider a circular cylinder of radius R and length L, and thus area $A = 2\,\pi\,R\,L$, subject to a longitudinal total traction force F uniformly distributed over its boundaries, as sketched in Fig. 2. The bending energy is taken as zero to test just the viscous operator. This problem has as exact solution a uniform extensional flow, in which the axial stresses equal $F/(2\pi\,R)$ and the

circumferential stresses equal $R\,\Delta p$, with Δp the pressure difference across the membrane. The exact solution for P is a uniform field of value

$$P = -\frac{1}{2}\left(\frac{F}{2\pi\,R} + R\,\Delta p\right)$$

and the exact extension rate is

$$\frac{dL}{dt} = \frac{L}{4\mu}\left(\frac{F}{2\pi\,R} - R\,\Delta p\right)$$

Taking $\Delta p = 0$, $F = 8$, $R(0) = L(0) = \mu = 1$ one obtains

$$L(t) = \frac{1}{1-\frac{t}{\pi}}, \qquad R(t) = 1 - \frac{t}{\pi}, \qquad P(t) = -\frac{2}{\pi - t}$$

In Fig. 3 we perform a mesh convergence study for this problem. We use structured meshes of 16×8, 32×16 and 64×32 subdivisions (each quadrilateral then divided into two triangles), with the nodal positions perturbed so that the edges are not aligned with the longitudinal or circumferential directions (otherwise superconvergence is achieved). A small membrane mass (0.05) is added, since otherwise the problem is ill-posed (the zero-dissipation-modes of the Stokes operator *on a cylinder* are infinitely many, and not just the rigid-body modes). The simulations are carried out with $\Delta t = 10^{-2}$.

Good agreement with the exact solution is observed both for $L(t)$ and for $\overline{P}(t)$ (the average of P over $\Gamma(t)$). Numerical fluctuations are observed in P_h, as depicted in Fig. 3 (bottom). Their amplitude seems to converge to zero with order $\mathcal{O}(h)$.

Convergence as $\Delta t \to 0$ is also observed. An upper bound for Δt arises from the lack of convergence of the Newton iterations. For $\Delta t = 10^{-1}$, for example, the algorithm ceases to converge at $t \simeq 2$.

4.2. Vesicle tweezing

Optical tweezing is an experimental method by which a laser beam can exert a force and move a bead attached to a membrane [49]. It is one of the emerging technique for the study of cell mechanobiology [26]. Continuum Mechanics simulations of red blood cells under tweezing have been reported by Dao *et al* [14] using an elastic membrane model in commercial software ABAQUS. A similar model was adopted for the immersed boundary simulations of Le *et al* [28]. The Canham–Helfrich bending energy was considered by Lee and coworkers [29] in their study of DOPC giant unilamellar vesicles. More complex, multiscale models have been recently advocated by Peng *et al* [39] for accurate simulation of red blood cells.

In this section we report a simulation of a viscous membrane in an optical tweezing configuration, with a time–dependent, sinusoidal applied force. We study a vesicle of reduced volume $v = 0.807$, for which the stationary shape

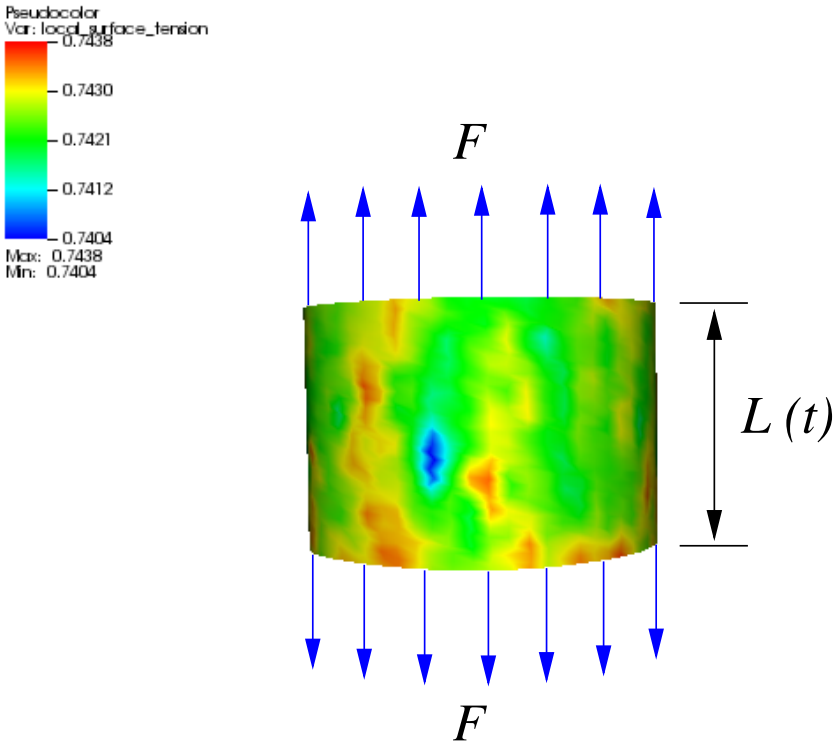

Figure 2: Viscous cylinder under traction. Sketch of the geometry and contour levels of P_h at $t = 0.44$.

of minimal energy is a prolate shape [47] with a normalized Canham–Helfrich energy of

$$E_{\text{prolate}} \;=\; \frac{1}{16\pi}\int_\Gamma H^2 \, d\Gamma \;\simeq\; 1.37$$

There exist however several other local minima, in particular there is an oblate shape with slightly higher energy

$$E_{\text{oblate}} \;\simeq\; 1.44$$

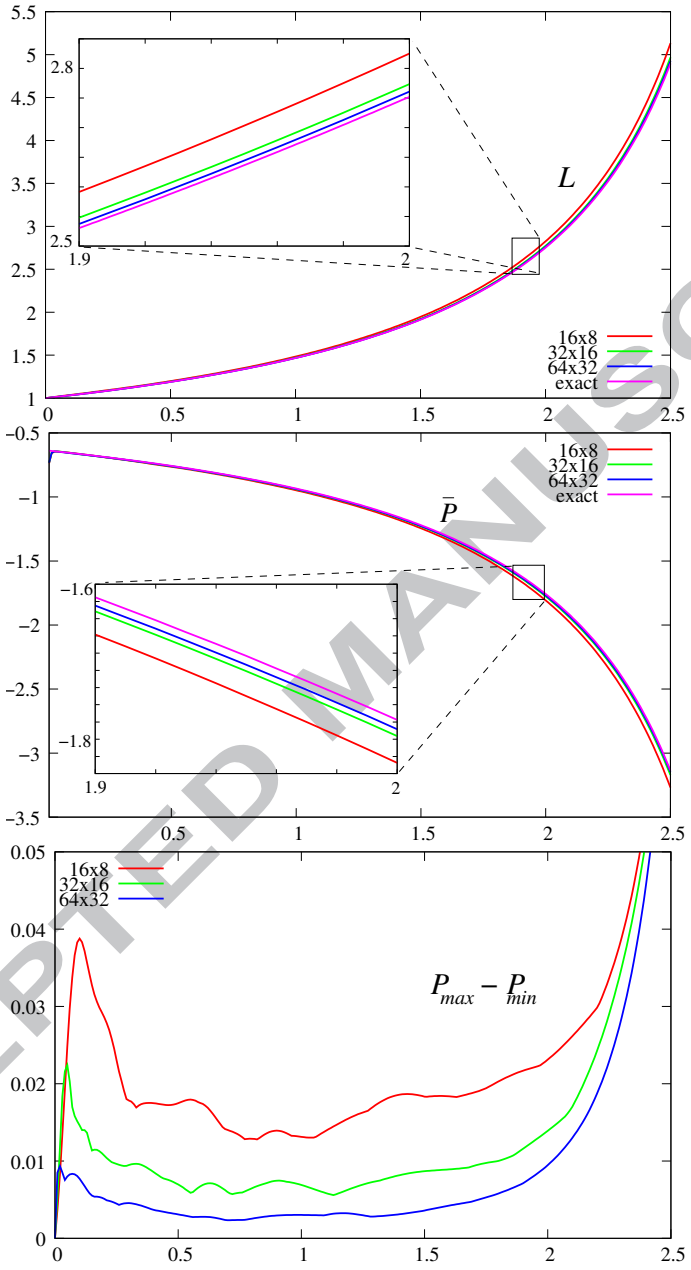

Figure 3: Viscous cylinder under traction. Mesh convergence study. Top: $L(t)$. Middle: $P_h(t)$ averaged over the membrane. Bottom: Difference between maximum and minimum values of P_h over the membrane.

Our purpose is to assess the performance of the proposed algorithm in simulating transitions between these two shapes as induced by a tweezing force of the form $F = A \sin((2\,\pi\,t/T)$. This force is applied on two opposite sides of the vesicles (uniformly distributed over a small set of elements), such that $F > 0$ is defined as traction and $F < 0$ as compression.

In Figs. 4 and 5 the normalized energy E is plotted as a function of time, together with the shapes corresponding to some instants. The vesicle starts from an arbitrary configuration, which in fact corresponds to a prolate shape oblique to the force direction. It is stretched sideways up to $t \simeq 0.2\,T$, when a transition occurs towards a prolate shape aligned with the force. This shape then relaxes towards equilibrium at $t = 0.5\,T$, when the applied force returns to zero. The value of E decreases to a value quite close to E_{prolate}, the difference being attributed to discretization errors. This prolate shape is then compressed by the force, and it consequently deforms increasing its energy until at $t \simeq 0.7T$ a second transition takes place that takes the vesicle to an oblate shape. This oblate shape persists after the compression force goes to zero at $t = T$, at which time $E \simeq 1.416$, in good agreement with E_{oblate}.

After $t = T$ the force pushes the concave sides of the disk outwards. This configuration renders the shape soon unstable and a transition back to the aligned prolate shape is observed at $t \simeq 1.16\,T$. This shape again relaxes to equilibrium at $t = 1.5\,T$ and then undergoes axial compression up to $t \simeq 1.7$, when it transitions again to oblate. The transitions at $t \simeq 0.7\,T$ and $t \simeq 1.7\,T$ are quite alike. In particular, they consist of first a relatively slow evolution towards a square–tile–like shape, followed by a sudden transition to the metastable, circular disk–like shape (metastable oblate).

Looking at the plot of $E(t)$, one observes the time scale of the applied force itself (of order T) and the time scales of oblate–to–prolate and prolate–to–oblate transitions, which is much smaller. An estimate of these time scales can be obtained from Fig. 6, in which $E(t)$ and the internal pressure $p(t)$ are plotted from $t = 0.7$ to $t = 0.8$. By visual inspection one may conclude that the time yielded by the simulation for the transition is, roughly, $\tau \sim 10^{-2}T = 80$. This is consistent with dimensional analysis. In fact, from the surface viscosity $\mu = 1$ [force-time/length], and the adopted $C_{CH} = 10^{-3}$ [energy], and assuming from the graphs that a typical length scale for the transitions is $L \simeq 0.3$ (the radius of the equilibrium prolate shape), the dimensional combination that renders a characteristic time is

$$\frac{\mu}{L^2\,C_{CH}} \simeq 100$$

not far from the predicted τ.

Further studies on the dynamical aspects of these shape transitions are the subject of ongoing work. The purpose here is just to illustrate the applicability of the proposed algorithm to extract qualitative and quantitative information of vesicle mechanics.

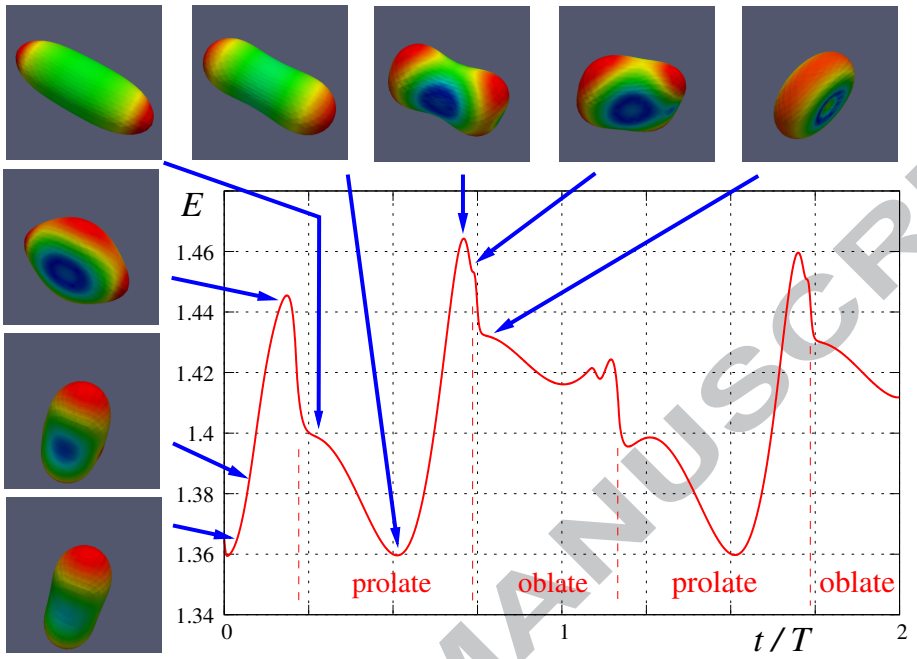

Figure 4: Vesicle tweezing with a force $F = A\sin(2\pi t/T)$ (positive/negative on traction/compression, respectively). Normalized energy ($\frac{1}{16\pi}\int_\Gamma H^2 d\Gamma$) as a function of t/T. Several instantaneous shapes show the transitions from oblate shape (metastable) to prolate shape (stable) and viceversa. The colors correspond to the local mean curvature (blue: minimum, red: maximum).

4.3. Surface viscous flow versus L^2-gradient flow

Several recent studies of biological membranes modeled by means of the Canham-Helfrich energy (or variants thereof) have focused on gradient flows evolving towards equilibrium [19, 33, 17, 7]. More specifically, this amounts to replacing the viscous operator $\mathcal{A}$ defined in (46) by an L^2-projection $\mathcal{P}$, i.e.;

$$\mathcal{P}(\boldsymbol{U}^t;\boldsymbol{w}) = \gamma \int_{\Gamma(t)} \boldsymbol{U}^t \cdot \boldsymbol{w}\; d\Gamma \tag{59}$$

Notice that this is equivalent to replacing the surface-viscous forces by a drag force of the form $-\gamma \boldsymbol{U}^t$, with γ the drag coefficient.

For comparison purposes, gradient flows were implemented in the same code with which the previous simulations were performed. Together with the drag forces (59), the volume and inextensibility constraints are active in much the same way as before. Further, cases in which the inextensibility constraint was only enforced globally (preservation of the total area alone) were considered. This is easily done by selecting for P^{n+1} the one-dimensional space of constant functions over Γ^{n+1}, instead of the $\mathcal{P}_1$ finite element space.

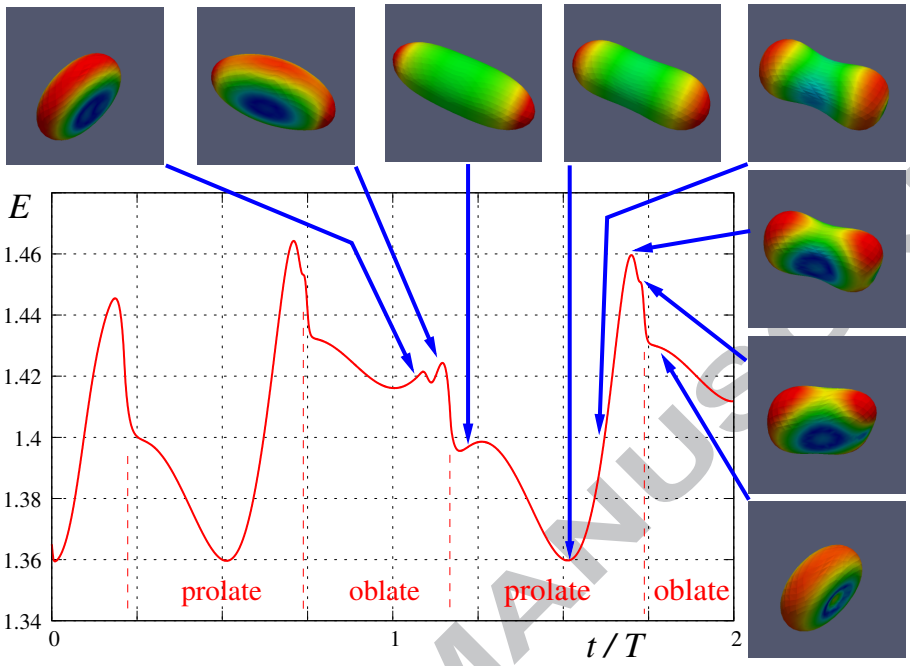

Figure 5: Continued.

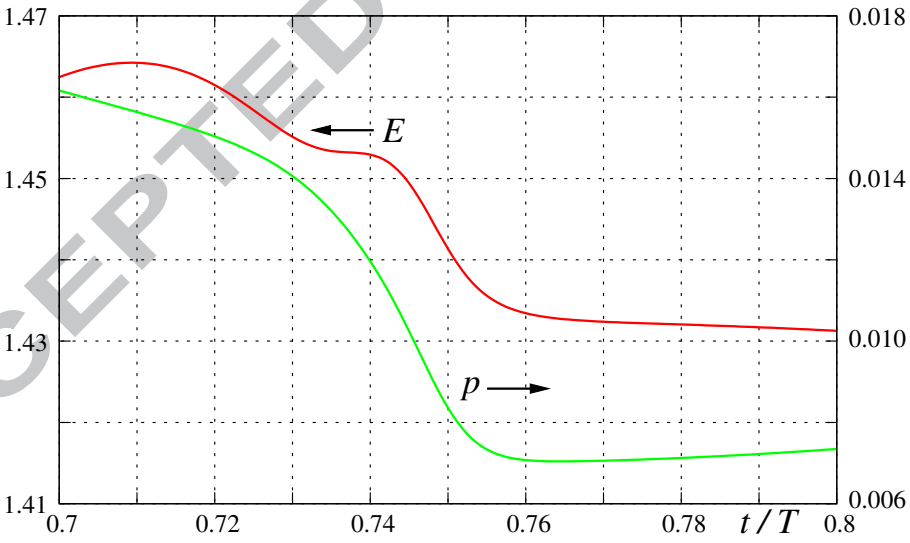

Figure 6: Energy E and internal pressure p plotted between $t = 0.7T$ and $t = 0.8T$.

Simulations were run for several values of μ and γ starting from the deformed shape corresponding to $t/T = 0.74$ in the tweezing simulation (Figs. 4-6) and without any external forces (pure relaxation). Curve 1 in Figs. 7 and 8 correspond to a gradient flow, with $\gamma = 0.1$, $\mu = 0$ and global inextensibility constraint (P^{n+1} uniform over Γ^{n+1}). Curve 5 in those figures, on the other hand, corresponds to viscous relaxation, with $\gamma = 0$, $\mu = 1$ and local inextensibility (P^{n+1} in $\mathcal{P}_1$). The evolution of these two cases is completely different. The main reason is that the mesh deteriorates very rapidly in gradient flow, the maximum angle in the mesh rises steeply and the simulation explodes. The surface viscous operator, on the other hand, is much nicer to the mesh and allows for the shape to evolve towards equilibrium with angles smaller than 145 degrees. This has already been exploited by Ma & Klug [33], who used a viscous regularization of the mesh in gradient flow computations. It is clear that the urgency for remeshing is much alleviated in surface viscous flow, though for large shape deformations remeshing is eventually needed.

Curves 2, 3 and 4 in Figs. 7 and 8 correspond to intermediate cases between gradient flow and inextensible viscous flow. Curve 2 corresponds to gradient flow with local inextensibility, which somewhat improves the mesh quality but still leads to collapse. Curves 3 and 4 corresponds to locally inextensible gradient flow with some added surface viscosity ($\mu = 0.01$ and $\mu = 0.1$, respectively). The surface viscosity penalizes tangential shear and thus leads to better-behaved meshes.

The shapes corresponding to Curves 4 and 5 are shown in Fig. 9 for several intermediate times. The left of each figure corresponds to the viscous case (Curve 5) and the right to the gradient case with added viscosity 0.1 (Curve 4). Both evolutions first approach the (metastable) oblate shape and end up in the (stable) prolate shape, though not following the exact same path. Viscous flow lingers for more time close to the oblate shape before transitioning (more abruptly) to the prolate shape. Though gradient and viscous flows will in general tend to the same equilibrium, the physically realistic viscous simulation is essential to evaluate any biological process associated to the *relaxation path* of a lipidic membrane.

5. Concluding remarks

Nature, at the length scale of the Living Cell, is a complex combination of some fundamental structures or building blocks. One such fundamental structure is a bidimensional arrangement of lipidic molecules that constitutes an important part of the cell's membrane and of several intracellular agents: Lipidic bilayers. In Mechanical terms, this bidimensional material admits a remarkably simple mathematical model: That of a bidimensional Newtonian fluid of surface viscosity μ, with a shape–dependent energy density proportional to the square of the local mean curvature. The need for accurate simulation procedures for this material is increased by the existing possibility of creating and manipulating lipidic bilayers for biochemical applications.

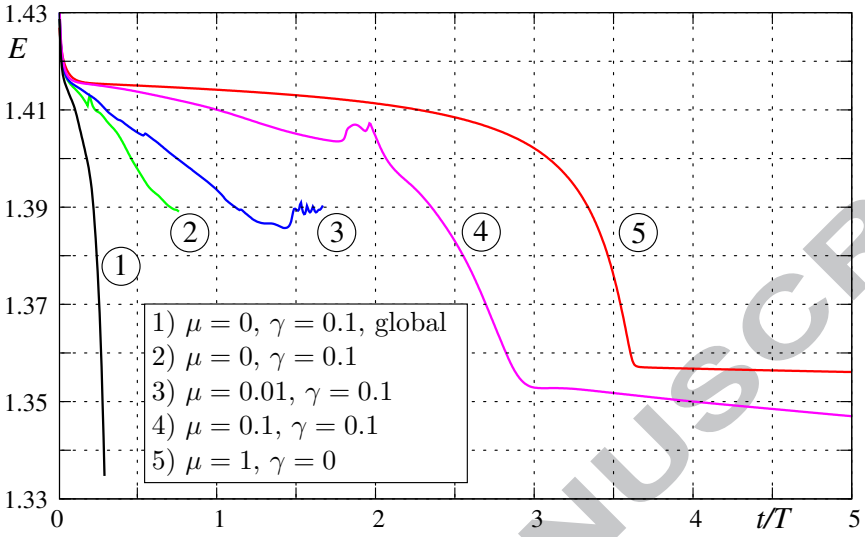

Figure 7: Normalized bending energy vs. time for relaxation with initial shape corresponding to $t/T = 0.74$ in Figures 4-6. Relaxation is simulated for different values of μ and γ. The first curve corresponds to imposing the inextensibility constraint only globally (preservation of total area).

Two–dimensional viscous behavior over a curved, evolving surface is practically inexistent in the macroscopic world. The simulation method for viscous membranes proposed in this article is built starting from well–established techniques used for macroscopic elastic membranes, from which viscous behavior can be obtained as elastic response with respect to an evanescent, continually updated reference configuration. The elastic energy "stored" with respect to this evanescent configuration is "forgotten" at each time step, which leads to an effective loss of energy or *dissipation*. This is an effective way of simulating a two-dimensional viscous *fluid* with well-established codes built to simulate two-dimensional elastic *solids*. The proposed method is also advantageous for modeling membranes that have both an elastic component *and* a viscous component.

Another, more CFD-like, methodology that is currently under evaluation is to directly discretize (in both time and space) the tangential viscous operator. It is given by [20]

$$\mathcal{A}(\boldsymbol{X}^t, \boldsymbol{U}^t; \boldsymbol{w}) \;=\; \int_{\Gamma(t)} 2\,\mu\, D_\Gamma \boldsymbol{U}^t : D_\Gamma \boldsymbol{w}\; d\Gamma \tag{60}$$

where, for any field $\boldsymbol{w}$ defined on $\Gamma(t)$,

$$D_\Gamma \boldsymbol{w} = \frac{1}{2}\left(\nabla_\Gamma \boldsymbol{w} + \nabla_\Gamma \boldsymbol{w}^T\right) \tag{61}$$

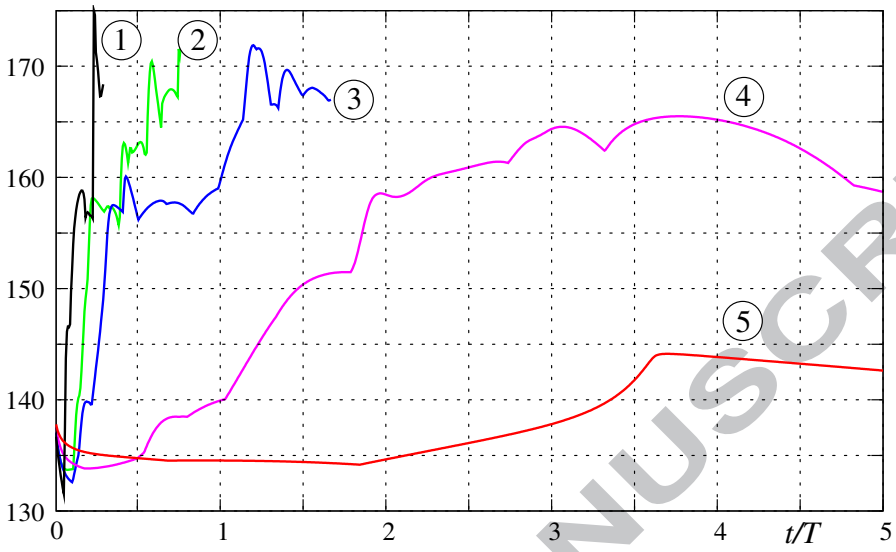

Figure 8: Maximum angle in the triangulation vs. time for relaxation with initial shape corresponding to $t/T = 0.74$ in Figures 4-6. Relaxation is simulated for different values of μ and γ, as explained in Fig. 7.

If inertia is neglected, the exact formulation can be seen as the autonomous evolution equation

$$\frac{d\boldsymbol{X}}{dt} = \boldsymbol{\mathcal{U}}(\boldsymbol{X}) \tag{62}$$

where the operator $\boldsymbol{\mathcal{U}}$ is the solution of the tangential Stokes problem (or Boussinesq-Stokes-Scriven problem) for the given shape $\boldsymbol{X}$ (eventually constrained to preserve internal volume) with the Canham-Helfrich forces driving the motion. These forces, which act along the normal to the surface, are given by

$$\boldsymbol{F} = C_{CH}\left(\nabla_\Gamma^2 H + \frac{1}{2}H^3 - 2\,K\,H\right)\tilde{\boldsymbol{n}} \tag{63}$$

(with K the Gaussian curvature) and thus only depend on the configuration $\boldsymbol{X}$ of Γ. Effective time-integration algorithms can then be applied to (62). Care must be taken, however, in that $\boldsymbol{\mathcal{U}}$ is defined up to a rigid-body motion in general. A word of caution: In two dimensions (meaning that Γ is a *curve*) the operator $\boldsymbol{\mathcal{U}}$ is ill defined, and for the problem to be well posed some inertia must be retained, or the viscosity of the internal fluid accounted for.

The version of the surface viscous operator given by (60) is advantageous for Eulerian methods such as volume-of-fluid or level set formulations. In these formulations, however, the implementation of the inextensibility condition is far from being obvious.

Future work involves the incorporation of a remeshing algorithm, since oth-

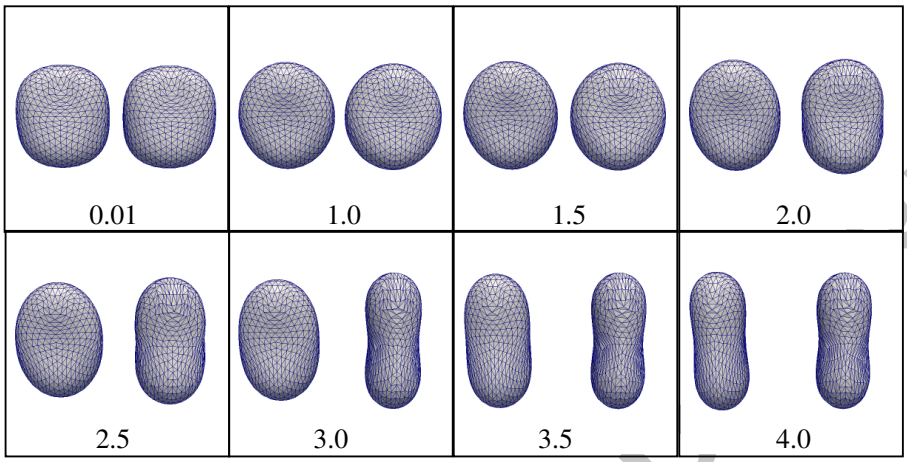

Figure 9: Shapes and meshes corresponding to Curves 4 and 5 of Figs. 7 and 8. The left side of each frame corresponds to viscous flow (Curve 5) and the right to gradient flow with added viscosity 0.1 (Curve 4). The time (in units of T) is given in each frame.

erwise phenomena involving very large deformations (e.g.; tethering) are untractable.

Acknowledgements

This work was partially supported by the Brazilian agencies CNPq, CAPES and FAPESP. The support of these agencies is gratefully acknowledged. The authors thank Q. Du, P. Morin, S. Pauletti and M. Arroyo for useful discussions.

References

[1] M. Arroyo and A. DeSimone. Relaxation dynamics of fluid membranes. *Phys. Rev. E*, 79:031915 (17 pages), 2009.

[2] M. Arroyo, A. DeSimone, and L. Heltai. The role of membrane viscosity in the dynamics of fluid membranes. Technical report, arXiv:1007.4934v1, 2010.

[3] E. Bänsch, P. Morin, and R. Nochetto. A finite element method for surface diffusion: the parametric case. *J. Comput. Phys.*, 203:321–343, 2005.

[4] J. Barrett, H. Garcke, and R. Nürnberg. On the parametric finite element approximation of evolving hypersurfaces in $\mathbb{R}^3$. *J. Comp. Phys.*, 227:4281–4307, 2008.

[5] G. Boedec, M. Leonetti, and M. Jaeger. 3D vesicle dynamics simulations with a linearly triangulated surface. *J. Comp. Phys.*, 230:1020–1034, 2011.

[6] J. Bonet, R. Wood, J. Mahaney, and P. Heywood. Finite element analysis of air supported membrane structures. *Comput. Methods Appl. Mech. Engrg.*, 190:579–595, 2000.

[7] A. Bonito, R. Nochetto, and S. Pauletti. Parametric FEM for geometric biomembranes. *J. Comp. Phys.*, 229:3171–3188, 2010.

[8] A. Bonito, R. Nochetto, and S. Pauletti. Dynamics of biomembranes: Effect of the bulk fluid. *Math. Model. Nat. Phenom.*, 6:25–43, 2011.

[9] G. Buscaglia and R. Ausas. Variational formulations for surface tension, capillarity and wetting. *Computer Meth. Appl. Mech. Engng.*, 200:3011–3025, 2011.

[10] G. Buscaglia, F. Basombrío, and R. Codina. Fourier analysis of an equal-order incompressible flow solver stabilized by pressure-gradient projection. *Int. J. Numer. Meth. in Fluids*, 34:65–92, 2000.

[11] P. Canham. The minimum energy of bending as a possible explanation of the biconcave shape of the human red blood cell. *J. Theoretical Biology*, 26:61–81, 1970.

[12] R. Codina. A stabilized finite element method for generalized stationary incompressible flows. *Comput. Methods Appl. Mech. Engrg.*, 190:2681–2706, 2001.

[13] R. Codina, J. Blasco, G. Buscaglia, and A. Huerta. Implementation of a stabilized finite element formulation for the incompressible Navier-Stokes equations based on a pressure gradient projection. *Int. J. Numer. Meth. in Fluids*, 37:410–444, 2001.

[14] M. Dao, C. Lim, and S. Suresh. Mechanics of the human red blood cell deformed by optical tweezers. *J. Mech. Phys. Solids*, 51:2259–2280, 2003.

[15] S. Das and Q. Du. Adhesion of vesicles to curved substrates. *Phys. Rev. E*, 77:011907, 2008.

[16] W. den Otter and S. Shkulipa. Intermonolayer friction and surface shear viscosity of lipid bilayer membranes. *Biophysical J.*, 93:423–433, 2007.

[17] G. Dziuk. Computational parametric Willmore flow. *Numer. Math.*, 111:55–80, 2008.

[18] C. Elliott and B. Stinner. Modeling and computation of two phase geometric biomembranes using surface finite elements. *J. Comp. Phys.*, 229:6585–6612, 2010.

[19] F. Feng and W. Klug. Finite element modeling of lipid bilayer membranes. *J. Comp. Phys.*, 220:394–408, 2006.

[20] S. Gross and A. Reusken. *Numerical Methods for Two-phase Incompressible Flows*. Number 40 in Series in Computational Mathematics. Springer, 2011.

[21] C. Harland, M. Bradley, and R. Parthasarathy. Phospholipid bilayers are viscoelastic. *Proc. Natl. Acad. Sci. USA*, 107:19146–19150, 2010.

[22] C. Harland, M. Bradley, and R. Parthasarathy. Retraction. *Proc. Natl. Acad. Sci. USA*, www.pnas.org/cgi/doi/10.1073/pnas.1111381108, 2011.

[23] W. Helfrich. Elastic properties of lipid bilayers – theory and possible experiments. *Zeitschrift für Naturforschung C*, 28:693, 1973.

[24] G. Holzapfel, R. Eberlein, P. Wriggers, and H. Weizsäcker. Large strain analysis of soft biological membranes: Formulation and finite element analysis. *Comput. Methods Appl. Mech. Engrg*, 132:45–61, 1996.

[25] T.J.R. Hughes, L. Franca, and M. Balestra. A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška–Brezzi condition: A stable Petrov-Galerkin formulation of the Stokes problem accommodating equal–order interpolations. *Comput. Methods Appl. Mech. Engrg.*, 59:85–99, 1986.

[26] D.H. Kim, P.K. Wong, J. Park, A. Levchenko, and Y. Sun. Microengineered platforms for cell mechanobiology. *Annual Review of Biomedical Engineering*, 11:203–233, 2009.

[27] E. Lac and D. Barthès-Biesel. Deformation of a capsule in simple shear flow: Effect of membrane prestress. *Physics of Fluids*, 17:072105, 2005.

[28] D. Le, J. White, J. Peraire, K. Lim, and B. Khoo. An implicit Immersed Boundary method for three-dimensional membrane-fluid flow interactions. *J. Comp. Phys.*, 229:8427–8445, 2009.

[29] H. Lee, E. Peterson, R. Phillips, W. Klug, and P. Wiggins. Membrane shape as a reporter for applied forces. *Proc. Nat. Acad. Sci.*, 105:19253–19257, 2008.

[30] X. Li and K. Sarkar. Front tracking simulation of deformation and buckling instability of a liquid capsule enclosed by an elastic membrane. *J. Comp. Phys.*, 227:4998–5018, 2008.

[31] G. Lim, M. Wortis, and R. Mukhopadhyay. Stomatocyte–discocyte–echinocyte sequence of the human red blood cell: Evidence for the bilayer–couple hypothesis from membrane mechanics. *Proc. Nat. Acad. Sci.*, 99:16766–16769, 2002.

[32] R. Lipowsky. The conformation of membranes. *Nature*, 349:475–481, 1991.

[33] L. Ma and W. Klug. Viscous regularization and r-adaptive remeshing for finite element analysis of lipid membrane mechanics. *J. Comp. Phys.*, 227:5816–5835, 2008.

[34] M.A. Mader, V. Vitkova, M. Abkarian, A. Viallat, and T. Podgorski. Dynamics of viscous vesicles in shear flow. *The European Physical Journal E: Soft Matter and Biological Physics*, 19:389–397, 2006.

[35] B. Massey and J. Ward-Smith. *Mechanics of fluids*. Stanley Thornes Ltd., 1998.

[36] A. Maude. Action and viscous flow. *British J. Applied Physics*, 8:343–344, 1957.

[37] M. Meyer, M. Desbrun, P. Schröder, and A. Barr. Discrete differential–geometry operators for triangulated 2–manifolds. *Visualization and Mathematics*, 3:34–57, 2002.

[38] H. Noguchi. Membrane simulation models from nanometer to micrometer scale. *J. Phys. Soc. Japan*, 78:041007, 2009.

[39] Z. Peng, R. Asaro, and Q. Zhu. Multiscale simulation of erythrocyte membranes. *Phys. Rev. E*, 81(3):031904, Mar 2010.

[40] C. Pozrikidis, editor. *Modeling and Simulation of Capsules and Biological Cells*. Mathematical Biology and Medicine Series. Chapman & Hall/CRC, 2003.

[41] S. Ramanujan and C. Pozrikidis. Deformation of liquid capsules enclosed by elastic membranes in simple shear flow: large deformations and the effect of fluid viscosities. *Journal of Fluid Mechanics*, 361:117–143, 1998.

[42] P. Saffman and M. Delbrück. Brownian motion in biological membranes. *Proc. Nat. Acad. Sci. USA*, 72:3111–3113, 1975.

[43] L. Saiz, S. Bandyopadhyay, and M. Klein. Towards an understanding of complex biological membranes from atomistic Molecular Dynamics simulations. *Bioscience Reports*, 22:151–173, 2002.

[44] D. Salac and M. Miksis. A level set projection model of lipid vesicles in general flows. *J. Comp. Phys.*, in press, 2011.

[45] L. Scriven. Dynamics of a fluid interface. *Chem. Eng. Sci.*, 12:98–108, 1960.

[46] T. Secomb and R. Skalak. Surface flow of viscoelastic membranes in viscous fluids. *Quart. J. Mech. Appl. Math.*, 35:233–248, 1982.

[47] U. Seifert. Configurations of fluid membranes and vesicles. *Advances in Physics*, 46:13–137, 1997.

[48] U. Seifert, K. Berndl, and R. Lipowsky. Shape transformations of vesicles: Phase diagram for spontaneous- curvature and bilayer-coupling models. *Phys. Rev. A*, 44(2):1182–1202, Jul 1991.

[49] M. Sheetz, editor. *Laser Tweezers in Cell Biology.* Academic Press, 1998.

[50] S. Shkulipa, W. den Otter, and W. Briels. Surface viscosity, diffusion and intermonolayer friction: Simulating sheared amphiphilic bilayers. *Biophysical J.*, 89:823–829, 2005.

[51] J. Sokolowski and J. Zolésio. *Introduction to shape optimization.* Springer–Verlag, 1992.

[52] Y. Sui, XB Chen, YT Chew, P. Roy, and HT Low. Numerical simulation of capsule deformation in simple shear flow. *Computers and Fluids*, 2009.

[53] S. Veerapaneni, A. Rahimian, G. Biros, and D. Zorin. A fast algorithm for simulating vesicle flows in three dimensions. *J. Comp. Phys.*, 230:5610–5634, 2011.

[54] X. Wang and Q. Du. Modelling and simulations of multi–component lipid membranes and open membranes via diffuse interface approaches. *Mathematical Biology*, 56:347–371, 2008.

[55] R. Waugh. Surface viscosity measurements from large bilayer vesicle tether formation. *Biophys. J.*, 38:29–37, 1982.