



Exact Newton method with third-order convergence to model the dynamics of bubbles in incompressible flow[☆]



Aymen Laadhari

Computer Vision Laboratory, Institut für Bildverarbeitung, Department of Information Technology and Electrical Engineering, Swiss Federal Institute of Technology Zürich (Eidgenössische Technische Hochschule Zürich), CH-8092 Zürich, Switzerland

ARTICLE INFO

Article history:

Received 3 November 2016

Received in revised form 16 January 2017

Accepted 16 January 2017

Available online 24 January 2017

Keywords:

Newton

Nonlinear problem

Third-order convergence

Navier–Stokes flow

Surface tension

Finite element method

ABSTRACT

In this letter, we present a computational framework based on the use of the Newton and level set methods and tailored for the modeling of bubbles with surface tension in a surrounding Newtonian fluid. We describe a fully implicit and monolithic finite element method that maintains stability for significantly larger time steps compared to the usual explicit method and features substantial computational savings. A suitable transformation avoids the introduction of an additional mixed variable in the variational problem. An exact tangent problem is derived and the nonlinear problem is solved by a quadratically convergent Newton method. In addition, we consider a generalization to the multidimensional case of the Kou's and McDougall's methods, resulting in a faster convergence. The method is benchmarked against known results with the aim of illustrating its accuracy and robustness.

© 2017 Elsevier Ltd. All rights reserved.

1. Introduction

The modeling of the dynamics of bubbles and multiphase flows with surface tension in a surrounding Newtonian flow has been the subject of several numerical investigations. The surface tension effect has a considerable importance in many real life phenomena and industrial applications, such as the bubbly wake of ships, air entrainment in oceans and gas–liquid flow in nuclear power plants. Several numerical methodologies have been developed using, for instance, the immersed boundary method [1], level set method [2], and Volume-of-Fluid method [3]. Most of the contributions have used fully explicit strategies and result in a severe stability constraint for the temporal resolution [4].

In this work a fully implicit and monolithic method based on the use of the Newton method is presented. Recently, a multitude of Newton variants have been derived, featuring faster convergence compared to the standard method without requiring higher order derivatives [5–7]. To our knowledge, such appealing

[☆] This work was supported by the Swiss Federal Institute of Technology Zürich and Swiss National Science Foundation grant #320030-149567.

E-mail address: laadhari@vision.ee.ethz.ch.

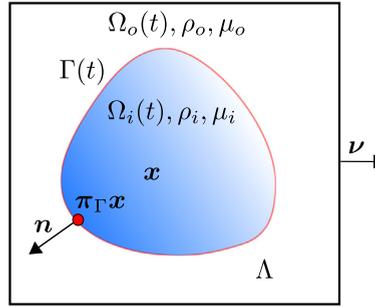


Fig. 1. A sketch for the interface and its surrounding domain.

methods have not been sufficiently investigated for such problems. We consider a generalization to the multidimensional case of the McDougall’s [7] and Kou’s [5] methods, and we show that the implicit solver allows substantial computational savings compared to the explicit approach. This framework is part of a larger, ongoing work to model red blood cells in microvasculature.

2. Mathematical formulation

Let $T > 0$ represent the period of the simulation. For any time $t \in (0, T)$, let $\Omega_i(t) \subset \mathbb{R}^d$ with $d = 2, 3$ represent the interior domain having a Lipschitz continuous boundary $\Gamma(t) = \partial\Omega_i(t)$. The interface $\Gamma(t)$ and its surrounding fluid $\Omega_o(t)$ are embedded in a larger computational domain Λ such that $\Gamma(t) \cap \partial\Lambda = \emptyset$, $\forall t \in (0, T)$, see Fig. 1. We denote by \mathbf{n} and $\boldsymbol{\nu}$ the outward unit normal vectors on $\Gamma(t)$ and $\partial\Lambda$, respectively. Let H be the total curvature on $\Gamma(t)$. Let the symbols \otimes and $:$ denote the tensorial product and the two times contracted product between tensors, respectively. The interface projector is expressed by $\boldsymbol{\pi}_\Gamma \equiv \mathbf{Id} - \mathbf{n} \otimes \mathbf{n}$, where \mathbf{Id} is the identity tensor. For any scalar field ψ and vector field \mathbf{v} , we introduce the surface gradient $\nabla_s \psi = \boldsymbol{\pi}_\Gamma \nabla \psi$ and the surface divergence $div_s \mathbf{v} = \text{tr}(\nabla_s \mathbf{v}) = \boldsymbol{\pi}_\Gamma : \nabla \mathbf{v}$.

From now, the explicit dependence of Ω_i , Ω_o and Γ from t will be understood.

2.1. Level set method

For any time $t \in (0, T)$, we follow implicitly the motion of Γ using the level set method. The interface is described as the iso-surface zero of a level set function φ in such a way that $\Gamma(t) = \left\{ (t, \mathbf{x}) \in (0, T) \times \Lambda : \varphi(t, \mathbf{x}) = 0 \right\}$. Let $\mathbf{u} \equiv \partial_t \mathbf{x}$ be the fluid velocity. The dynamics of Γ is described by a time-dependent partial differential equation (2.3). This problem is initialized with a signed distance function ϕ_0 to $\Gamma(0)$ and is equipped with a suitable boundary condition $\varphi = \varphi_b$ on the upstream boundary $\Sigma_- = \{ \mathbf{x} \in \partial\Lambda : \mathbf{u} \cdot \boldsymbol{\nu}(\mathbf{x}) < 0 \}$. All geometrical quantities are extended to Λ and are easily recovered implicitly in terms of φ , e.g. $\mathbf{n} = \frac{\nabla \varphi}{|\nabla \varphi|}$ and $H = div_s \mathbf{n}$. Let ε be a regularization parameter proportional to the mesh size, referred to as h . In practice, we usually set $\varepsilon = 2.5h$. The Dirac measure δ_Γ and the Heaviside function \mathcal{H} are regularized in a banded strip of width 2ε around Γ using the following expressions:

$$\delta_\varepsilon(\varphi) = \begin{cases} \frac{1}{2\varepsilon} \left(\varphi + \cos\left(\frac{\pi\varphi}{\varepsilon}\right) \right), & \text{when } |\varphi| \leq \varepsilon, \text{ and} \\ 0, & \text{otherwise} \end{cases}$$

$$\mathcal{H}_\varepsilon(\varphi) = \begin{cases} 0, & \text{when } \varphi < -\varepsilon \\ \frac{1}{2} \left(1 + \frac{\varphi}{\varepsilon} + \frac{1}{\pi} \sin\left(\frac{\pi\varphi}{\varepsilon}\right) \right), & \text{when } |\varphi| \leq \varepsilon, \\ 1, & \text{otherwise.} \end{cases}$$

For any function $\psi(\cdot)$ defined on Γ , let $\tilde{\psi}(\cdot)$ stand for an extension to the entire domain Λ in such a way that it is constant in the normal direction to the interface. Integrals over Γ are approximated by:

$$\int_{\Gamma} \psi(\mathbf{x}) \, ds = \int_{\Lambda} |\nabla\varphi| \delta_{\Gamma} \tilde{\psi}(\mathbf{x}) \, d\mathbf{x} \approx \int_{\Lambda} |\nabla\varphi| \delta_{\varepsilon}(\varphi) \tilde{\psi}(\mathbf{x}) \, d\mathbf{x}.$$

The transport of φ (2.3) degenerates the signed distance property and can deteriorate the computational accuracy on Γ . To reestablish the signed distance property, we solve a redistancing problem following the approach described in [8].

2.2. Statement of the nonlinear coupled problem

We consider the instationary Navier–Stokes equations and we assume piecewise constant density $\rho_{i/o}$ and viscosity $\mu_{i/o}$, see Fig. 1. In an Eulerian framework, a continuous description of the global density and viscosity is performed: $\rho_{\varepsilon}(\varphi) = \rho_i + (\rho_o - \rho_i)\mathcal{H}_{\varepsilon}(\varphi)$ and $\mu_{\varepsilon}(\varphi) = \mu_i + (\mu_o - \mu_i)\mathcal{H}_{\varepsilon}(\varphi)$. Let $\mathbf{D}(\mathbf{u}) = (\nabla\mathbf{u} + \nabla\mathbf{u}^T)/2$ and $\boldsymbol{\sigma}(\mathbf{u}, p, \varphi) = 2\mu(\varphi)\mathbf{D}(\mathbf{u}) - p\mathbf{Id}$ be the strain and fluid Cauchy stress tensors, respectively. Let $[\cdot]_{\pm}^{\pm}$ design the discontinuity across Γ and γ be the surface tension coefficient. The interface movement is dictated by the interplay between the hydrodynamic forces and the capillary force. The discontinuity of the normal stresses is calibrated by the interface force, whereas the interface and fluid move with the same velocity. The system is subject to an external body force \mathbf{g} .

We consider the non-dimensionalized problem and we introduce the density and viscosity ratios $\rho^* = \rho_i/\rho_o$ and $\mu^* = \mu_i/\mu_o$. The characteristic scales of length and time are D and $L/\sqrt{D|\mathbf{g}|}$, respectively, where D and L are the diameter of $\Gamma(0)$ and the size of Λ . The Reynolds and capillary numbers are defined by $Re = \rho_o U_g D/\mu_o$ and $Ca = \mu_o \sqrt{D|\mathbf{g}|}/\gamma$, respectively. From now, all quantities and domains are non-dimensionalized, while we use the same notations for ease of exposition. Equipped with suitable initial and boundary conditions, the coupled nonlinear problem describing the dynamics of the interface with surface tension reads:

(\mathcal{P}) find \mathbf{u} , p and φ such that

$$Re \rho_{\varepsilon}(\varphi) (\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u}) - \operatorname{div} (2\mu_{\varepsilon}(\varphi) \mathbf{D}(\mathbf{u})) + \nabla p = Re \rho_{\varepsilon}(\varphi) \mathbf{g} \quad \text{in } (0, T) \times \Lambda, \tag{2.1}$$

$$\operatorname{div} \mathbf{u} = 0 \quad \text{in } (0, T) \times \Lambda, \tag{2.2}$$

$$\partial_t \varphi + \mathbf{u} \cdot \nabla \varphi = 0 \quad \text{in } (0, T) \times \Lambda, \tag{2.3}$$

$$[\mathbf{u}]_{\pm}^{\pm} = \mathbf{0} \quad \text{and} \quad [\boldsymbol{\sigma} \mathbf{n}]_{\pm}^{\pm} = \frac{1}{Ca} H \mathbf{n} \quad \text{on } (0, T) \times \Gamma. \tag{2.4}$$

3. Finite element approximation and solution method

3.1. Semi-discretization in time

Let us divide $[0, T]$ into N subintervals $[t_n, t_{n+1}]$, $n = 0, \dots, N - 1$ of constant time steps Δt . For any $n > 0$, the unknowns \mathbf{u}_n , p_n and φ_n approximate \mathbf{u} , p and φ at time t_n , respectively. Similarly, $\boldsymbol{\pi}_{\Gamma}$ is approximated by $\boldsymbol{\pi}_{\Gamma, n} = \mathbf{Id} - \mathbf{n}_n \otimes \mathbf{n}_n$ at t_n . Let Σ_D be the set on which essential boundary conditions for \mathbf{u} are set. At the numerical level, the curvature usually represents a difficult term that should be accurately computed. To decrease the derivation order with respect to φ and avoid the introduction of a mixed variable, we use the following key transformation suitable for surface integrals:

$$\int_{\Gamma} H \mathbf{n} \cdot \mathbf{w} = \int_{\Gamma} \mathbf{w} \cdot \nabla_s \mathbf{1} + \int_{\Gamma} \operatorname{div}_s \mathbf{w} - \int_{\partial\Gamma=\emptyset} \boldsymbol{\nu}_{\partial\Gamma} \cdot \mathbf{w} = \int_{\Gamma} \boldsymbol{\pi}_{\Gamma} : \nabla \mathbf{w},$$

where $\nu_{\partial\Gamma}$ represents the co-normal vector that is normal to $\partial\Gamma$ and tangent to Γ , and \mathbf{w} represents a generic vector. We introduce the functional spaces:

$$\mathbb{V}(\mathbf{u}_b) = \left\{ \mathbf{v} \in (H^1(\Lambda))^d : \mathbf{v} = \mathbf{u}_b \text{ on } \Sigma_D \right\}, \mathbb{Q} = \left\{ q \in L^2(\Lambda) : \int_{\Lambda} q = 0 \right\} \text{ and}$$

$$\mathbb{X}(\varphi_b) = \left\{ \psi \in L^2(\Lambda) \cap W^{1,\infty}(\Lambda) : \psi = \varphi_b \text{ on } \Sigma_- \right\}.$$

For the time approximation, we use the backward differentiation scheme of second order for the time derivative terms. The scheme is bootstrapped by the initial conditions $\mathbf{u}_{-1} = \mathbf{u}_0 = \mathbf{u}(0)$ and $\varphi_{-1} = \varphi_0 = \phi_0(0)$, where \mathbf{u}_{-1} and φ_{-1} only represent suitable notations. Let $\chi_n \equiv (\mathbf{u}_n, p_n, \varphi_n)$ be the global vector of unknowns. After regularization and semi-discretization in time, the problem \mathcal{P} consists in finding $\chi_n \in \mathbb{V}(\mathbf{u}_b) \times \mathbb{Q} \times \mathbb{X}(\varphi_{n-1})$ such that

$$\begin{aligned} & Re \int_{\Lambda} \rho_{\varepsilon}(\varphi_n) \left(\frac{3\mathbf{u}_n - 4\mathbf{u}_{n-1} + \mathbf{u}_{n-2}}{2\Delta t} + \mathbf{u}_n \cdot \nabla \mathbf{u}_n \right) \cdot \mathbf{v} + \int_{\Lambda} 2\mu_{\varepsilon}(\varphi_n) \mathbf{D}(\mathbf{u}_n) : \mathbf{D}(\mathbf{v}) - \int_{\Lambda} p_n \operatorname{div} \mathbf{v} \\ & - \frac{1}{Ca} \int_{\Lambda} \delta_{\varepsilon}(\varphi_n) |\nabla \varphi_n| \boldsymbol{\pi}_{\Gamma,n} : \nabla \mathbf{v} - Re \int_{\Lambda} \rho_{\varepsilon}(\varphi_n) \mathbf{g} \cdot \mathbf{v} = 0, \quad \forall \mathbf{v} \in \mathbb{V}(0), \end{aligned} \tag{3.1}$$

$$\int_{\Lambda} q \operatorname{div} \mathbf{u}_n = 0, \quad \forall q \in \mathbb{Q}, \tag{3.2}$$

$$\int_{\Lambda} \frac{3\varphi_n - 4\varphi_{n-1} + \varphi_{n-2}}{2\Delta t} \psi + \int_{\Lambda} (\mathbf{u}_n \cdot \nabla \varphi_n) \psi = 0. \quad \forall \psi \in \mathbb{X}(0). \tag{3.3}$$

3.2. Exact tangent problem and Newton–Raphson method

Let $\mathcal{R}(\chi)$ be the global residual corresponding to (3.1)–(3.3) and $\langle \cdot, \cdot \rangle$ stand for the duality product. Let $D\mathcal{R}(\chi)[\delta\chi]$ denote the Gâteaux derivative of \mathcal{R} at χ along the direction $\delta\chi$. The standard Newton method reduces the nonlinear problem $\mathcal{R}(\chi) = \mathbf{0}$ into a sequence of linear sub-problems. Let $\delta\chi_n^k \equiv (\delta\mathbf{u}_n^k, \delta p_n^k, \delta\varphi_n^k)$ be the increment. We iteratively compute χ_n^k at t_n by a nonlinear Richardson method in such a way that, for any sub-iteration $k \geq 0$, the solution explicitly expresses as $\chi_n^{k+1} = \chi_n^k + \delta\chi_n^k$ with $D\mathcal{R}(\chi_n^k)[\delta\chi_n^k] = -\mathcal{R}(\chi_n^k)$. A second order extrapolation enables to assign the starting values, and the stopping criterion is based on the residual computation. The Newton tolerance is $\epsilon_{tot} = 10^{-8}$. In what follows, we drop the superscript n whenever it is clear from the context.

In addition, we consider a generalization to the multidimensional case of the McDougall’s method [7] and Kou’s method [5]. The McDougall’s method is a modification of the standard Newton method with a convergence of order $1 + \sqrt{2}$ at a cost of one residual evaluation and one Jacobian assembly and factorization per iteration. That results in an efficiency index (convergence order per function or derivative evaluation) of $(1 + \sqrt{2})^{1/2}$. This method with memory re-uses the same Jacobian matrix in the next step and requires the resolution of two linear systems per iteration. The Kou’s method requires two residual evaluations and one assembly and factorization of the Jacobian per iteration. The method is cubically convergent with an efficiency index of $\sqrt[3]{3}$.

| | | |
|---|---|--|
| Kou’s method [5]: $k \geq 0 : D\mathcal{R}(\chi^k) [\chi^{k+0.5} - \chi^k] = \mathcal{R}(\chi^k),$ $D\mathcal{R}(\chi^k) [\chi^{k+1} - \chi^{k+0.5}] = -\mathcal{R}(\chi^{k+0.5}).$ | } | McDougall’s method [7]: $k = 0 : \chi_{\star}^0 = \chi^0 \text{ and } D\mathcal{R}(\chi^0) [\chi^1 - \chi^0] = -\mathcal{R}(\chi^0)$ $k \geq 1 : D\mathcal{R} \left(\frac{\chi^{k-1} + \chi_{\star}^{k-1}}{2} \right) [\chi_{\star}^k - \chi^k] = -\mathcal{R}(\chi^k),$ $D\mathcal{R} \left(\frac{\chi^k + \chi_{\star}^k}{2} \right) [\chi_{\star}^{k+1} - \chi^k] = -\mathcal{R}(\chi^k).$ |
|---|---|--|

Thereafter, we proceed with the derivation of the exact tangent problem in the case of the standard Newton method. We first provide some useful directional derivatives in the direction of a level set increment $\delta\varphi$:

$$\begin{aligned}
D\mu_\varepsilon(\varphi)[\delta\varphi] &= (1 - \mu^*)\delta_\varepsilon(\varphi)\delta\varphi, & D\frac{1}{|\nabla\varphi|}[\delta\varphi] &= -\nabla\delta\varphi \cdot \frac{\nabla\varphi}{|\nabla\varphi|^3}, \\
D\mathbf{n}[\delta\varphi] &= \frac{\nabla\delta\varphi}{|\nabla\varphi|} - \frac{(\nabla\varphi \cdot \nabla\delta\varphi)\nabla\varphi}{|\nabla\varphi|^3} = \frac{\pi_\Gamma \nabla\delta\varphi}{|\nabla\varphi|} = \frac{\nabla_s\delta\varphi}{|\nabla\varphi|}, \\
D\pi_\Gamma[\delta\varphi] &= \frac{-1}{|\nabla\varphi|^2} \left(\nabla_s\delta\varphi \otimes \nabla\varphi + \nabla\varphi \otimes \nabla_s\delta\varphi \right) & \text{and} & \quad D \operatorname{div}_s \mathbf{v}[\delta\varphi] = D\pi_\Gamma[\delta\varphi] : \nabla \mathbf{v}.
\end{aligned}$$

To write the problem in a compact manner suitable for a straightforward finite element implementation, we introduce the weighted multi-linear forms defined for all $q \in L^2(\Lambda)$; $w \in L^\infty(\Lambda)$; $\varphi, \psi \in \mathbb{X}$; $\mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathbb{V}$ and $\boldsymbol{\tau} \in (L^\infty(\Lambda))^{d \times d}$:

$$\begin{aligned}
a(\mathbf{u}, \mathbf{v}; w) &= \int_\Lambda 2w \mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{v}); & b(\mathbf{u}, q; \boldsymbol{\tau}) &= - \int_\Lambda q \boldsymbol{\tau} : \nabla \mathbf{u}; \\
c(\mathbf{u}, \mathbf{v}; w, \mathbf{w}) &= \int_\Lambda w ((\mathbf{u} \cdot \nabla) \mathbf{w} + (\mathbf{w} \cdot \nabla) \mathbf{u}) \cdot \mathbf{v}; \\
e(\varphi, \psi) &= \int_\Lambda \varphi \psi; & h(\varphi, \mathbf{v}; w, \mathbf{w}) &= \int_\Lambda 2\varphi w \mathbf{D}(\mathbf{v}) : \mathbf{D}(\mathbf{w}); & k(\varphi, \mathbf{v}; \mathbf{w}, \boldsymbol{\tau}) &= \int_\Lambda \mathbf{w} \cdot \nabla \varphi (\boldsymbol{\tau} : \nabla \mathbf{v}); \\
i(\varphi, \psi; \mathbf{w}) &= \int_\Lambda \psi \mathbf{w} \cdot \nabla \varphi; \\
g(\varphi, \mathbf{v}; \mathbf{w}) &= \int_\Lambda \varphi \mathbf{v} \cdot \mathbf{w}; & m(\mathbf{u}, \mathbf{v}; w) &= \int_\Lambda w \mathbf{u} \cdot \mathbf{v}; \\
l(\varphi, \mathbf{v}; \mathbf{w}, \boldsymbol{\tau}) &= \int_\Lambda ((\boldsymbol{\tau} \cdot \nabla \varphi) \otimes \mathbf{w} + \mathbf{w} \otimes (\boldsymbol{\tau} \cdot \nabla \varphi)) : \nabla \mathbf{v}.
\end{aligned}$$

The exact tangent system associated to $\mathcal{P}(t_n)$ (3.1)–(3.3) reads: given $\boldsymbol{\chi}^k$, find $\delta\boldsymbol{\chi}^k \in \mathbb{V}(\mathbf{u}_b) \times \mathbb{Q} \times \mathbb{X}(\varphi_{n-1})$ such that

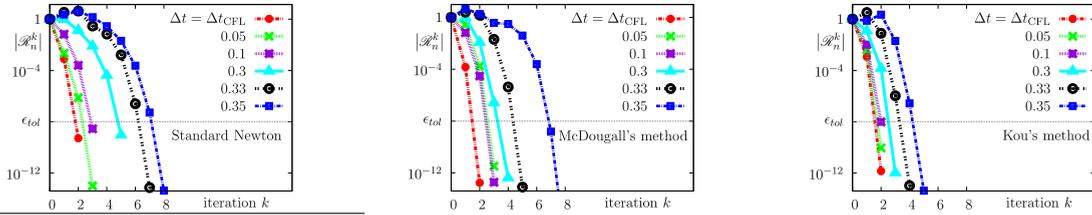
$$\begin{aligned}
& \frac{3Re}{2\Delta t} m(\delta\mathbf{u}^k, \mathbf{v}; \rho_\varepsilon(\varphi^k)) + Re c(\delta\mathbf{u}^k, \mathbf{v}; \rho_\varepsilon(\varphi^k), \mathbf{u}^k) + a(\delta\mathbf{u}^k, \mathbf{v}; \mu_\varepsilon(\varphi^k)) + b(\mathbf{v}, \delta p^k; \mathbf{Id}) \\
& + (1 - \mu^*) h(\delta\varphi^k, \mathbf{v}; \delta_\varepsilon(\varphi^k), \mathbf{u}^k) \\
& + Re(1 - \rho^*) g(\delta\varphi^k, \mathbf{v}; \delta_\varepsilon(\varphi^k) \left(\frac{3\mathbf{u}^k - 4\mathbf{u}_{n-1} + \mathbf{u}_{n-2}}{2\Delta t} + \mathbf{u}^k \cdot \nabla \mathbf{u}^k - \mathbf{g} \right)) \\
& + \frac{1}{Ca} l(\delta\varphi^k, \mathbf{v}; \delta_\varepsilon(\varphi^k) \frac{\nabla\varphi^k}{|\nabla\varphi^k|}, \boldsymbol{\pi}_\Gamma^k) \\
& - \frac{1}{Ca} k(\delta\varphi^k, \mathbf{v}; \delta_\varepsilon(\varphi^k) \frac{\nabla\varphi^k}{|\nabla\varphi^k|}, \boldsymbol{\pi}_\Gamma^k) + \frac{1}{Ca} b(\mathbf{v}, \delta\varphi^k; \delta'_\varepsilon(\varphi^k) |\nabla\varphi^k| \boldsymbol{\pi}_\Gamma^k) = - \langle \mathcal{R}_\boldsymbol{\chi}(\boldsymbol{\chi}^k), \mathbf{v} \rangle_{\mathbb{V}(\mathbf{0})', \mathbb{V}(\mathbf{0})}, \\
b(\delta\mathbf{u}^k, q; \mathbf{Id}) &= - \langle \mathcal{R}_p(\mathbf{u}^k), q \rangle_{\mathbb{Q}', \mathbb{Q}}, \\
\frac{3}{2\Delta t} e(\delta\varphi^k, \psi) + i(\delta\varphi^k, \psi; \mathbf{u}^k) + g(\psi, \delta\mathbf{u}^k; \nabla\varphi^k) &= - \langle \mathcal{R}_\varphi(\varphi^k, \mathbf{u}^k), \psi \rangle_{\mathbb{X}(\mathbf{0})', \mathbb{X}(\mathbf{0})},
\end{aligned}$$

for all $\mathbf{v} \in \mathbb{V}(\mathbf{0})$, $q \in \mathbb{Q}$ and $\psi \in \mathbb{X}(\mathbf{0})$, where the corresponding residuals are expressed by:

$$\begin{aligned}
\langle \mathcal{R}_\boldsymbol{\chi}(\boldsymbol{\chi}^k), \mathbf{v} \rangle_{\mathbb{V}(\mathbf{0})', \mathbb{V}(\mathbf{0})} &= \frac{Re}{2\Delta t} m(3\mathbf{u}^k - 4\mathbf{u}_{n-1} + \mathbf{u}_{n-2}, \mathbf{v}; \rho_\varepsilon(\varphi^k)) + \frac{Re}{2} c(\mathbf{u}^k, \mathbf{v}; \rho_\varepsilon(\varphi^k), \mathbf{u}^k) \\
& + \frac{1}{Ca} b(\mathbf{v}, \delta_\varepsilon(\varphi^k); |\nabla\varphi^k| \boldsymbol{\pi}_\Gamma^k) + a(\mathbf{u}^k, \mathbf{v}; \mu_\varepsilon(\varphi^k)) + b(\mathbf{v}, p^k; \mathbf{Id}) - Re m(\mathbf{g}, \mathbf{v}; \rho_\varepsilon(\varphi^k)), \\
\langle \mathcal{R}_p(\mathbf{u}^k), q \rangle_{\mathbb{Q}', \mathbb{Q}} &= b(\mathbf{u}^k, q; \mathbf{Id}) \quad \text{and} \\
\langle \mathcal{R}_\varphi(\varphi^k, \mathbf{u}^k), \psi \rangle_{\mathbb{X}(\mathbf{0})', \mathbb{X}(\mathbf{0})} &= \frac{1}{2\Delta t} e(3\varphi^k - 4\varphi_{n-1} + \varphi_{n-2}, \psi) + i(\varphi^k, \psi; \mathbf{u}^k).
\end{aligned}$$

Table 1

Convergence curves of the residuals and the corresponding orders of convergence, obtained with $h = 1/30$ and $\Delta t_{\text{CFL}} \approx 0.011503407$.



| k | $\Delta t = \Delta t_{\text{CFL}}$ | 0.05 | 0.1 | 0.3 | 0.33 | 0.35 |
|-----|------------------------------------|-------|-------|--------|--------|-------|
| 2 | 1.997 | 1.306 | 2.058 | 41.280 | 0.187 | 0.670 |
| 3 | | 1.980 | 2.045 | 1.440 | 11.673 | 1.852 |
| 4 | | | | 1.853 | 0.534 | 1.532 |
| 5 | | | | 2.049 | 2.586 | 1.634 |
| 6 | | | | | 2.354 | 1.700 |
| 7 | | | | | 1.712 | 1.877 |
| 8 | Standard Newton | | | | | |
| | | | | | | 1.668 |

| k | $\Delta t = \Delta t_{\text{CFL}}$ | 0.05 | 0.1 | 0.3 | 0.33 | 0.35 |
|-----|------------------------------------|-------|-------|-------|-------|--------|
| 2 | 2.334 | 5.784 | 2.846 | 2.419 | 0.585 | 0.588 |
| 3 | | | 2.414 | 2.450 | 1.247 | 7.868 |
| 4 | | | | | 1.979 | 0.123 |
| 5 | | | | | 2.098 | 10.514 |
| 6 | | | | | | 2.462 |
| 7 | McDougall's method | | | | | |
| | | | | | | 2.361 |

| k | $\Delta t = \Delta t_{\text{CFL}}$ | 0.05 | 0.1 | 0.3 | 0.33 | 0.35 |
|-----|------------------------------------|-------|-------|-------|-------|-------|
| 2 | 3.042 | 3.206 | 3.059 | 3.184 | 3.100 | 3.984 |
| 3 | | | | 2.788 | 3.001 | 4.327 |
| 4 | | | | | 1.436 | 2.374 |
| 5 | Kou's method | | | | | |
| | | | | | | 1.397 |

Table 2

(Left) Computation time CPU on one processor for the implicit methods for several values of Δt . (middle) Comparison of the total simulation time with respect to the fully explicit method. (Right) Maximum time step size against the time step limit from the stability criterion.

| Δt | CPU[s]: | Newton | McDougall's | Kou's |
|------------|------------|--------|-------------|--------|
| 0.1 | | 28.43 | 29.05 | 36.51 |
| 0.33 | | 83.89 | 79.72 | 76.08 |
| 0.36 | | 153.03 | 136.62 | 121.66 |
| | $h = 1/50$ | | | |

| h | CPU[s]: | explicit | Kou's |
|-------|---------|-----------|-----------|
| 1/20 | | 186.90 | 169.44 |
| 1/40 | | 363.51 | 304.35 |
| 1/80 | | 5'429.13 | 3'966.21 |
| 1/100 | | 14'015.72 | 11'186.39 |

| $1/h$ | Δt_{CFL} (Stability condition) | Δt_{max} (Kou's) |
|-------|---|---------------------------------|
| 20 | 2.11×10^{-2} | 0.540 |
| 40 | 7.74×10^{-3} | 0.375 |
| 80 | 2.64×10^{-3} | 0.310 |
| 160 | 9.34×10^{-4} | 0.205 |

4. Sample numerical result

This framework has been implemented using the Rheolef environment for scientific computing [9]. Parallelism relies on MPI, while MUMPS is used for the factorization and as direct solver on distributed-memory architectures.

To validate our computational framework, we consider the rising bubble benchmark [2]. Consider a circular bubble having a radius $r_0 = 0.25$ and centered on $(0.5, 0.5)$ in a computational domain $[0, 1] \times [0, 2]$. The physical parameters are $\rho_i = 100$, $\rho_o = 1000$, $\mu_i = 1$, $\mu_o = 10$, $\gamma = 24.5$ and $|g| = 0.98$. The no-slip wall condition $\mathbf{u}_b = \mathbf{0}$ is prescribed on the horizontal boundaries, whereas the free slip condition $\mathbf{u} \cdot \boldsymbol{\nu} = 0$ and $\boldsymbol{\pi}_{\partial A} \mathbf{D}(\mathbf{u}) \cdot \boldsymbol{\nu} = \mathbf{0}$ is used on the vertical boundaries.

We first provide further insights into the performances of the fully implicit methods. For an explicit method, the capillary force is considered as a source term in (3.1), while the fluid and level set problems are solved in a segregated manner. This method is conditionally stable and a stability constraint imposes restrictions on the temporal resolution $\Delta t < \Delta t_{\text{CFL}} = \sqrt{\frac{\rho_i + \rho_o}{4\pi\gamma}} h^{3/2}$ [4]. The convergence orders, $\frac{\ln(|\mathcal{R}(\mathbf{x}_n^k)|/|\mathcal{R}(\mathbf{x}_n^{k-1})|)}{\ln(|\mathcal{R}(\mathbf{x}_n^{k-1})|/|\mathcal{R}(\mathbf{x}_n^{k-2})|)}$ with $k \geq 2$, and the residual curves are reported in Table 1 for several values of Δt . We clearly see the quadratic convergence of the standard Newton method, the improvement by the McDougall's method and the cubic convergence of the Kou's method.

We consider a mesh with 13'194 elements and we report in Table 2 the computing times for serial simulations on an Intel® Core™ i7-4790 (3.6 GHz) processor. The McDougall's method is usually slightly better than the standard method, while the Kou's method performs better for large Δt . For large Δt , the convergence requires more iterations since the starting values are not close enough to the expected solutions. The extra cost of the additional residual evaluation in the Kou's method is offset by the cubic convergence. In what follows, the Kou's method is our preferred method. We prefer this method to the cubically convergent method described in [6] since the residual evaluation is three times cheaper than the Jacobian assembly

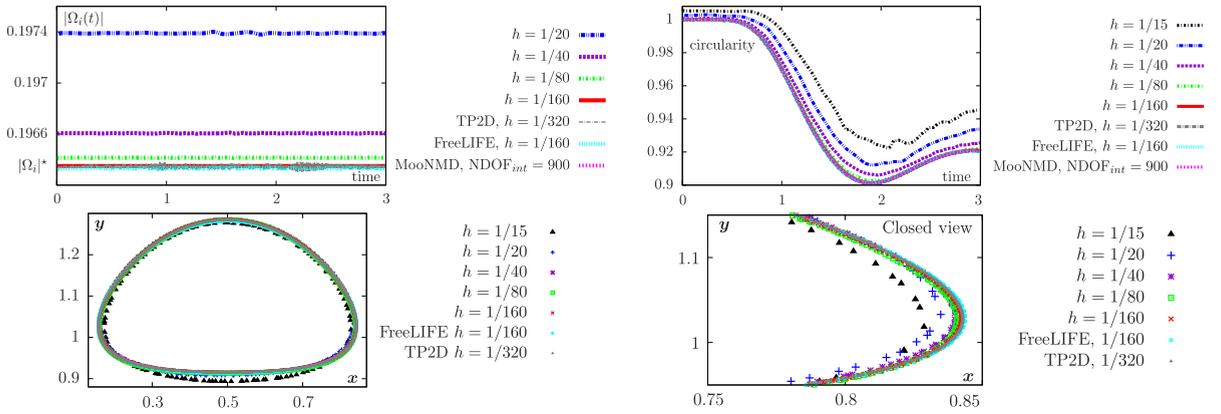


Fig. 2. Time evolution of the area and ϕ for different spatial resolutions and comparison of the final shapes with the reference solutions [2].

and factorization. For a mesh having 135'110 elements, the CPU times of the residual evaluation, Jacobian assembly and Jacobian factorization are 31.61 s, 42.59 s and 52.11 s, respectively.

We now perform a comparison with the fully explicit method. Results in Table 2(middle) depict significant computational savings when using the implicit method. To investigate the stabilizing capabilities of this method, we compare in Table 2(right) the maximum time step size Δt_{\max} to Δt_{CFL} allowed by the stability condition. Results reveal that the implicit method has a significant stabilizing effect and allows to use larger time steps, up to 200 times Δt_{CFL} .

Finally, we proceed with quantitative comparisons with the benchmark results obtained by six different codes in [2]. Let $|\Omega_i|$ and $|\Gamma|$ be the area of the interior domain Ω_i and the perimeter of the interface Γ , respectively. Let us introduce the circularity $\phi(t) = 2\sqrt{\pi|\Omega_i|/|\Gamma|}$, minimum circularity ϕ_{\min} and incidence time $t|_{\phi=\phi_{\min}}$, center of mass Y_c , rise velocity V_c , maximal velocity $V_{c,\max}$ and incidence time $t|_{V_c=V_{c,\max}}$. Let $|\Omega_i|^* = \pi/16$ represent the exact area of the interior domain. Comparisons of the final shapes at $t = 3$ and detailed view in the zone of maximal discrepancy reveal quite good congruence for finer meshes, see Fig. 2. In addition, results show good mass preservation and good agreement of the time evolution of ϕ with the reference solutions. In Table 3, quantitative comparisons with the computational results in [2,3,10,11] show overall very consistent results.

Let NTS be the number of time steps. The spatial accuracy of the numerical approximation of a quantity ζ_t is measured by computing normalized errors on successively refined meshes against a reference solution ζ_t^r (with $h = 1/160$). We compute:

$$|e|_1 = \frac{\sum_{t=1}^{\text{NTS}} |\zeta_t^r - \zeta_t|}{\sum_{t=1}^{\text{NTS}} |\zeta_t^r|}, \quad |e|_2 = \left(\frac{\sum_{t=1}^{\text{NTS}} |\zeta_t^r - \zeta_t|^2}{\sum_{t=1}^{\text{NTS}} |\zeta_t^r|^2} \right)^{1/2}, \quad |e|_\infty = \frac{\max_t |\zeta_t^r - \zeta_t|}{\max_t |\zeta_t^r|}$$

and the convergence rate

$$\text{ROC} = \frac{\ln(|e^{l-1}|/|e^l|)}{\ln(h^{l-1}/h^l)},$$

where l is the mesh refinement level. Results from Table 3 suggest that the bubble's area $|\Omega_i|$ has a convergence order of 2.5 in all norms, while the rise velocity V_c approaches a convergence order of 2 in all norms.

Table 3

Comparisons with available published results and convergence history with respect to the spatial resolution for some benchmark quantities.

| $1/h$ | 20 | 40 | 80 | 160 | Benchmark [2] | Results [3, 10, 11] | $1/h$ | $ e _1$ | ROC | $ e _2$ | ROC | $ e _\infty$ | ROC | |
|-------------------------|--------|--------|--------|---------------|---------------------|---------------------|------------|---------|---------|---------|---------|--------------|---------|-------|
| ϕ_{\min} | 0.9121 | 0.9060 | 0.9026 | 0.9014 | 0.9012 ± 0.0001 | 0.8974 ± 0.0098 | Ω_i | 20 | 5.87E-3 | | 5.87E-3 | 5.89E-3 | | |
| $t _{\phi=\phi_{\min}}$ | 1.9286 | 1.9714 | 1.8928 | 1.8697 | 1.8895 ± 0.0145 | 1.9180 ± 0.0445 | | 40 | 1.21E-3 | 2.279 | 1.21E-3 | 2.279 | 1.24E-3 | 2.252 |
| $V_{c,\max}$ | 0.2404 | 0.2417 | 0.2418 | 0.2418 | 0.2419 ± 0.0002 | 0.2405 ± 0.0058 | | 80 | 2.37E-4 | 2.352 | 2.37E-4 | 2.352 | 2.42E-4 | 2.353 |
| $t _{V_c=V_{c,\max}}$ | 1.0000 | 0.9571 | 0.9357 | 0.9252 | 0.9263 ± 0.0050 | 0.9333 ± 0.0183 | V_c | 20 | 2.08E-2 | | 2.37E-2 | 4.10E-2 | | |
| $Y_c(t=3)$ | 1.0768 | 1.0798 | 1.0799 | 1.0799 | 1.0808 ± 0.0009 | 1.0730 ± 0.0085 | | 40 | 4.80E-3 | 2.118 | 6.00E-3 | 1.983 | 1.16E-2 | 1.822 |
| | | | | | | | | 80 | 1.37E-3 | 1.812 | 1.71E-3 | 1.813 | 3.38E-3 | 1.776 |

5. Conclusion

In this letter a fully implicit and monolithic method to model the dynamics of bubbles has been developed. We have derived an exact tangent problem without introducing additional mixed variables and have investigated numerically the performances of two Newton variants with faster convergence orders. We have shown that the method has great practical utility compared to the explicit method commonly used. It features an affordable computational burden and maintains stability for significantly larger time steps. This is part of an ongoing work to model red blood cells [1,8,12,13]. We foresee the coupling with the bilayer bending force and the cytoskeleton elasticity model.

References

- [1] M.-C. Lai, Y. Seol, A short note on Navier-Stokes flows with an incompressible interface and its approximations, *Appl. Math. Lett.* 65 (2017) 1–6.
- [2] S. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, L. Tobiska, Quantitative benchmark computations of two-dimensional bubble dynamics, *Internat. J. Numer. Methods Fluids* (ISSN: 1097-0363) 60 (11) (2009) 1259–1288.
- [3] J. Klostermann, K. Schaake, R. Schwarze, Numerical simulation of a single rising bubble by VOF with surface compression, *Internat. J. Numer. Methods Fluids* 71 (8) (2013) 960–982.
- [4] J.U Brackbill, D.B Kothe, C. Zemach, A continuum method for modeling surface tension, *J. Comput. Phys.* 100 (2) (1992) 335–354.
- [5] J. Kou, Y. Li, X. Wang, A modification of Newton method with third-order convergence, *Appl. Math. Comput.* 181 (2) (2006) 1106–1111.
- [6] J. Kou, Y. Li, X. Wang, Third-order modification of Newton’s method, *J. Comput. Appl. Math.* 205 (1) (2007) 1–5.
- [7] T.J. McDougall, S.J. Wotherspoon, A simple modification of Newton’s method to achieve convergence of order $1 + \sqrt{2}$, *Appl. Math. Lett.* 29 (2014) 20–25.
- [8] A. Laadhari, P. Saramito, C. Misbah, An adaptive finite element method for the modeling of the equilibrium of red blood cells, *Internat. J. Numer. Methods Fluids* (ISSN: 1097-0363) 80 (7) (2016) 397–428.
- [9] Efficient C++ finite element computing with Rheolef, 2016, www.ljk.imag.fr/membres/Pierre.Saramito/rheolef (Accessed 1 November 2016).
- [10] L. Štrubelj, I. Tiselj, B. Mavko, Simulations of free surface flows with implementation of surface tension and interface sharpening in the two-fluid model, *Int. J. Heat Fluid Flow* 30 (4) (2009) 741–750.
- [11] V. Doyeux, Y. Guyot, V. Chabannes, C. Prud’homme, M. Ismail, Simulation of two-fluid flows using a finite element/level set method. Application to bubbles and vesicle dynamics, *J. Comput. Appl. Math.* 246 (2013) 251–259.
- [12] E.M. Kolahdouz, D. Salac, A numerical model for the trans-membrane voltage of vesicles, *Appl. Math. Lett.* 39 (2015) 7–12.
- [13] A. Laadhari, P. Saramito, C. Misbah, G. Székely, Computing the dynamics of biomembranes and capillary interfaces by combining the level set and Newton methods, *J. Comput. Phys.*, submitted for publication.