Numerical solution of the time-dependent Navier-Stokes equation for variable density–variable viscosity

Owe Axelsson[*], Xin He[†] and Maya Neytcheva[†]

Abstract

We consider methods for numerical simulations of variable density incompressible fluids, modelled by the Navier-Stokes equations. Variable density problems arise, for instance, in interfaces between fluids of different densities in multiphase flows such as appear in porous media problems. It is shown that by solving the Navier-Stokes equation for the momentum variable instead of the velocity, the corresponding saddle point problem, which arises at each time step, becomes automatically regularized, enabling elimination of the pressure variable and leading to a, for the iterative solution, efficient preconditioning of the arising block matrix. We present also stability bounds and a second order operator splitting method. The theory is illustrated by numerical experiments. For reasons of comparison we also include test results for a method, based on coupling of the Navier-Stokes equations with a phase-field model.

Keywords: Navier-Stokes equations, variable density, variable viscosity, operator splitting methods, regularization, phase-field model, finite elements, iterative methods, preconditioning.

1 Introduction

Variable density problems arise in many complex fluid flow processes of current interest, and has been studied intensively via numerical simulations. As an example, which ranges over a large number of difficulties, we mention the general circulation models, dealing with the coupled water–atmosphere dynamics. In this model, both the density and the viscosity of the water and the air are varying. The water is volume-wise incompressible while the air is compressible and the processes are influenced by numerous other factors, such as temperature, salinity, moisture etc., leading to a system where high instabilities, such as turbulence, occur.

[*] Institute of Geonics AS CR, Ostrava, The Czech Republic and King Abdulaziz University, Jeddah, Saudi Arabia

[†] Department of Information Technology, Uppsala University, Sweden. Email: owe.axelsson, he.xin, maya.neytcheva@it.uu.se

Important variable density problems arise also in laminar flows and the work, presented in this article, deals primarily with the numerical solution of such models. Examples of the latter type are the variable density ground water flow phenomena, which have been intensively studied in the last decades. There, density-driven flow occur and the effect of variable density becomes significant, in particular when a fluid of high density overlays a fluid of lower density (c.f., e.g., [41]). Density-driven flows are of fundamental importance also when solving transport problems in porous media (see, e.g., [15]).

Another class of variable density problems arise in modelling of the interaction of several phases, for example droplet impact onto a solid or liquid surface, accurate tracking of interface surfaces between fluids of different density in multiphase flow problems, etc.

There exist two main classes of methods for solving multiphase problems - *sharp interface* and *diffuse interface* methods. As the names suggest, in the former case the interfaces are assumed to be of zero thickness and the aim is to resolve those exactly, taking into account the discontinuities of the problem coefficients across those interfaces. In the latter case it is assumed that the interfaces have some nonzero thickness and the problem parameters vary continuously between the different flow phases. The advantages and disadvantages of both approaches are discussed in a vast amount of related literature ([49]).

A major part of the numerical simulation of flow models is the ability to efficiently solve the nonlinear time-dependent variable density, variable-viscosity Navier-Stokes (NS) equations. NS is either combined with a mass balance equation for the density or with another equation, which describes the evolution of the interfaces between the different phases, such as the Cahn-Hilliard (CH) equations (cf., e.g., [47, 7]), where there is no need to solve a separate equation for the density, since a special *'concentration'* variable determines automatically the relative portion of the masses and an interface of chosen thickness. Due to its high complexity and nonlinearity, the corresponding coupled system of partial differential equations is usually solved using some operator splitting scheme. Therefore, a rigorous numerical study of the problem includes two important aspects - estimate of the discretization error, resulting from the splitting scheme and efficiency and robustness of the numerical solution techniques to solve the arising nonlinear and linear algebraic systems of discrete equations. We note that, in addition, when implementing those numerical solution methods, special attention has to be paid to the discretization meshes, which have to be fine enough to resolve the interfaces and follow their dynamics. In this work we deal mostly with NS, coupled to the mass balance equation for the density, representing the sharp interface models. However, we also include some numerical experiments within the diffuse interface approach, where NS is coupled with CH. Our main focus is on the numerical solution techniques, their robustness and computational efficiency.

In Section 2 we formulate the coupled Navier-Stokes-density equations to be solved. In order to prepare for the analysis of the numerical solution of this problem we present also some stability estimates. In Section 3 we present the operator splitting and linearization method used, and derive an error estimate for the splitting error, based on these approximations. In Section 4 we discuss the stability of the time-stepping method and a regularization of the problem to avoid locking and unphysical oscillations of the numerically computed pressure variable is presented. Then, in Section 5 we show that the regularization used leads to an efficient inner-outer iterative solution method of the arising block matrix system. Section 6 contains a presentation of a symmetric operator splitting method, by which errors due to solving N-S and the density equation separately leads to a higher order scheme in time than if we use a pure sequential operator splitting. This method needs less computational effort than the familiar methods by Marchuk [31] and Strang [42]. Section 7 contains numerical results. We conclude the paper with some remarks, such as the possibility to couple the equation with elasticity equations when solving porous media problems.

Remark 1.1 *In this work we do not consider the issue how to impose additional interior boundary conditions in the case of sharp interface methods. For details and recent achievements, we refer to [49].*

2 The Navier-Stokes equations with variable density

Let Ω be a smooth, bounded and connected domain in $\mathbb{R}^d$, $d = 2, 3$ and let $\mathbf{u}$, p, ρ, μ denote respectively the velocity vector, pressure, density and viscosity of a fluid in Ω.

2.1 Formulation of the coupled system

The variation of the variables $\mathbf{u}$ and ρ is described by the Navier-Stokes equations, augmented with an additional advection equation for the density. The fluid is assumed to be incompressible or, at least, nearly incompressible. As described, e.g, in [29], see also [18], the corresponding non-stationary incompressible equations take the form

$$\begin{aligned} \frac{\partial}{\partial t}(\rho\mathbf{u}) + \nabla\cdot(\rho\mathbf{u}\otimes\mathbf{u}) - \nabla\cdot(\mu D(\mathbf{u})) + \nabla p &= \rho\mathbf{f} && (1)\\ \nabla\cdot\mathbf{u} &= 0 && (2)\\ \frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) &= 0 && (3) \end{aligned}$$

in $\Omega \times (0, \infty)$. Here, $D(\mathbf{u}) = \frac{1}{2}(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)$ is the symmetrized deformation tensor, $\rho\mathbf{f}$ is a force function (per unit volume), where typically $\mathbf{f} = \mathbf{g}$, $\mathbf{g}$ being the gravity field. The incompressibility of the fluids is understood in the sense that density cannot be changed by changes in pressure (cf. e.g. [44]).

We clarify the term $\nabla\cdot(\rho\mathbf{u}\otimes\mathbf{u})$, which in some other works is denoted as $\nabla\cdot(\rho\mathbf{u}\,\mathbf{u})$. In 3D, in detailed form it reads as

$$\nabla\cdot(\rho\mathbf{u}\,\mathbf{u}) = \begin{bmatrix} \dfrac{\partial(\rho u^2)}{\partial x} + \dfrac{\partial(\rho uv)}{\partial y} + (\dfrac{\partial\rho uw)}{\partial z} \\ \dfrac{\partial(\rho uv)}{\partial x} + \dfrac{\partial(\rho v^2)}{\partial y} + (\dfrac{\partial\rho vw)}{\partial z} \\ \dfrac{\partial(\rho uw)}{\partial x} + \dfrac{\partial(\rho vw)}{\partial y} + (\dfrac{\partial\rho w^2)}{\partial z} \end{bmatrix},$$

where $\mathbf{u} = [u, v, w]$.

Using the incompressibility of the fluid, we can rewrite the above which appears as the term $\nabla\cdot(\rho\mathbf{u}\otimes\mathbf{u})$ in (1) as follows.
For the ith component it holds

$$\sum_j \partial_j(\rho u_i u_j) = \sum_j (u_j\partial_j(\rho u_i) + \rho u_i \partial_j u_j) = (\mathbf{u}\cdot\nabla)(\rho u_i) + \rho u_i \nabla\cdot\mathbf{u},$$

so

$$\nabla\cdot(\rho\mathbf{u}\otimes\mathbf{u}) = (\mathbf{u}\cdot\nabla)(\rho\mathbf{u}).$$

We assume that the viscosity depends on density as some Lipschitz-continuous, positive function $\mu(\rho)$ in the interval $[0,\infty)$. It follows then from (2) and (3) that a similar equation as for the density holds also for the viscosity. Namely, we have,

$$\frac{\partial\mu}{\partial t} + \mathbf{u}\cdot\nabla\mu = \frac{\partial\mu}{\partial\rho}\left(\frac{\partial\rho}{\partial t} + \mathbf{u}\cdot\nabla\rho\right) = \frac{\partial\mu}{\partial\rho}\left(\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u})\right) = 0. \tag{4}$$

Hence, we can add $\frac{\partial\mu}{\partial t} + \mathbf{u}\cdot\nabla\mu = 0$ or $\frac{\partial\mu}{\partial t} + \nabla\cdot(\mu\mathbf{u}) = 0$ to the above set of equations.

The set of equations takes then the form

$$\frac{\partial}{\partial t}(\rho\mathbf{u}) + (\mathbf{u}\cdot\nabla)(\rho\mathbf{u}) - \nabla\cdot(\mu D(\mathbf{u})) + \nabla p = \rho\,\mathbf{f} \tag{5}$$

$$\frac{\partial}{\partial t}\begin{bmatrix}\rho\\ \mu\end{bmatrix} + \begin{bmatrix}\nabla\cdot(\rho\mathbf{u})\\ \nabla\cdot(\mu\mathbf{u})\end{bmatrix} = \begin{bmatrix}0\\ 0\end{bmatrix} \tag{6}$$

$$\nabla\cdot\mathbf{u} = 0 \tag{7}$$

Introducing the momentum variable $\mathbf{v} = \rho\mathbf{u}$ and using the relation

$$\nabla\cdot\mathbf{v} \equiv \nabla\cdot(\rho\mathbf{u}) = \rho\nabla\cdot\mathbf{u} + \mathbf{u}\cdot\nabla\rho = \mathbf{u}\cdot\nabla\rho,$$

equations (5)-(7) take the form

$$\frac{\partial}{\partial t}(\mathbf{v}) + (\mathbf{u}\cdot\nabla)(\mathbf{v}) - \nabla\cdot(\mu D(\mathbf{u})) + \nabla p = \rho\,\mathbf{f} \tag{8}$$

$$\frac{\partial}{\partial t}\begin{bmatrix}\rho\\ \mu\end{bmatrix} + \mathbf{u}\cdot\nabla\begin{bmatrix}\rho\\ \mu\end{bmatrix} = \begin{bmatrix}0\\ 0\end{bmatrix} \tag{9}$$

$$\nabla\cdot\mathbf{v} - \mathbf{u}\cdot\nabla\rho = 0 \tag{10}$$

where $\mathbf{u} = \frac{1}{\rho}\mathbf{v}$. The initial and boundary conditions for the system (8)-(10) are assumed to be

$$\begin{aligned}&\rho|_{t=0} = \rho_0, \quad (\rho\mathbf{u})|_{t=0} = \mathbf{v}_0, \quad \mu|_{t=0} = \mu_0 = \mu(\rho_0),\\ &\mathbf{u}|_\Gamma = \mathbf{b}, \qquad \rho|_\Gamma = a,\end{aligned}$$

where $\Gamma = \partial\Omega$ and $a > 0$.

Since the advection equation (6) has been rewritten in the form (9), i.e., as a first order hyperbolic equation, it follows that the boundary conditions for $\left[\rho,\ \mu\right]^T$ are given at a possible inflow boundary, $\Gamma_{in} = \{\mathbf{x} \in \Gamma, \mathbf{u}\cdot\mathbf{n} < 0\}$. Therefore, $\left[\rho,\ \mu\right]^T_{\Gamma_{in}} = \left[a,\ \mu(a)\right]^T$.

Note, that Γ_{in} can vary with time. However, in our test problems, we assume that Γ_{in} is fixed, possibly empty. As for Navier-Stokes (N-S) equations with constant density, there is no need to impose any initial or boundary conditions for the pressure variable. The pressure is uniquely defined only up to a constant term. To make it unique, one normally imposes the additional constraint $\int_\Omega p\, d\Omega = 0$.

We assume that the given data are such that $\rho(t)$ and $\mathbf{u}(\cdot, t)$ belong to some Sobolev space, which is smoother than $H^1(\Omega)$. In the sequel we assume that μ is a known function of ρ.

For existence, unicity of the solution and stability estimates, see [29], [18] and [30].

2.2 Stability properties of the underlying equations

With reference to the discussion on non-zero thickness of interfaces we assume that the density is described by a continuous function. We present some relations, which turn out to be useful for showing stability of the operator splitting scheme, to be analysed in the next section. Usually such relations are shown to hold for the numerical solution, however we present their analogues for the continuous solution. Similar estimates can be found in the related literature, see, e.g., [29], but up to the knowledge of the authors, they are not identical.

We assume here that the no penetration boundary condition $\mathbf{u}\cdot\mathbf{n}|_\Gamma = 0$ holds.

By the assumption of the boundary condition and incompressibility, it follows that

$$\int\limits_\Omega \rho\mathbf{u}\cdot\nabla\rho\, d\Omega = \int\limits_\Omega \frac{1}{2}\mathbf{u}\cdot\nabla(\rho^2)\, d\Omega = -\int\limits_\Omega \frac{1}{2}\nabla\cdot\mathbf{u}\,\rho^2\, d\Omega + \int\limits_{\partial\Omega} \frac{1}{2}\mathbf{u}\cdot\mathbf{n}\rho^2\, ds = 0.$$

Hence, a variational formulation of the mass balance equation leads to

$$\frac{1}{2}\frac{d}{d\,t}\left(\int\limits_\Omega \rho^2\, d\Omega\right) = \int\limits_\Omega \left[\rho\frac{\partial\rho}{\partial t} + \rho\mathbf{u}\cdot\nabla\rho\right] d\Omega = 0, \tag{11}$$

that is,

$$\|\rho(\cdot, t)\|_0 = \|\rho_0\|_0,\ t > 0, \tag{12}$$

where $\|\cdot\|_0$ denotes the $L^2(\Omega)$ norm. Hence, in this norm, density is constant in time.

Using a variational formulation of the momentum equation (6), we find

$$\int\limits_\Omega \left[\underbrace{\frac{\partial}{\partial t}(\mathbf{v})\cdot\mathbf{u} + (\mathbf{u}\cdot\nabla)(\mathbf{v})\cdot\mathbf{u}}_{(A)}\ \underbrace{-\nabla\cdot(\mu D(\mathbf{u}))\cdot\mathbf{u} + \nabla\, p\cdot\mathbf{u}}_{(B)}\right] d\Omega = \int\limits_\Omega \rho\,\mathbf{f}\cdot\mathbf{u}\, d\Omega.$$

Consider first the time derivative and the advection terms. Since $\mathbf{u} = (1/\rho)\mathbf{v}$, after multiplying by $\mathbf{u}$ and integrating over Ω, we obtain

$$
\begin{aligned}
(A): \quad & \int_\Omega \left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{v}\right)\frac{1}{\rho}\mathbf{v}\, d\Omega \\
= \; & \int_\Omega \left(\frac{1}{2\rho}\frac{\partial}{\partial t}(|\mathbf{v}|^2) + \frac{1}{2\rho}\mathbf{u}\cdot\nabla(|\mathbf{v}|^2)\right) d\Omega \\
= \; & \int_\Omega \left(\frac{1}{2\rho}\frac{\partial}{\partial t}(|\mathbf{v}|^2) - \frac{1}{2\rho}\nabla\cdot\mathbf{u}|\mathbf{v}|^2 - \mathbf{u}\cdot\nabla\left(\frac{1}{2\rho}\right)|\mathbf{v}|^2\right) d\Omega \\
= \; & \int_\Omega \left(\frac{1}{2\rho}\frac{\partial}{\partial t}(|\mathbf{v}|^2) + \frac{1}{2\rho^2}\mathbf{u}\cdot\nabla\rho|\mathbf{v}|^2\right) d\Omega,
\end{aligned}
$$

where we have used the divergence theorem, the incompressibility and the homogeneous boundary conditions for $\mathbf{u}$. Using the mass conservation equation $\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0$, we find

$$
\begin{aligned}
& \int_\Omega \left(\frac{1}{2\rho}\frac{\partial}{\partial t}(|\mathbf{v}|)^2 + \frac{1}{2\rho^2}\mathbf{u}\cdot\nabla\rho|\mathbf{v}|^2\right) d\Omega \\
= \; & \int_\Omega \frac{1}{2}\left[\frac{\partial}{\partial t}\left(\frac{1}{\rho}|\mathbf{v}|^2\right) - \frac{\partial}{\partial t}\left(\frac{1}{\rho}\right)|\mathbf{v}|^2 - \frac{1}{\rho^2}\frac{\partial\rho}{\partial t}|\mathbf{v}|^2\right] d\Omega \\
= \; & \int_\Omega \frac{1}{2}\frac{\partial}{\partial t}\left(\frac{1}{\rho}|\mathbf{v}|^2\right) d\Omega.
\end{aligned}
$$

For the diffusion and pressure gradient terms, the divergence theorem shows that

$$
\begin{aligned}
(B): \quad & \int_\Omega (-\nabla\cdot(\mu D(\mathbf{u})) + \nabla p)\,\mathbf{u}\, d\Omega \\
= \; & \int_\Omega \mu|\nabla\mathbf{u}|^2 - \int_\Omega p\nabla\cdot\mathbf{u} + \int_{\partial\Omega} p\mathbf{u}\cdot\mathbf{n}\, d\Gamma = \int_\Omega \mu|\nabla\mathbf{u}|^2.
\end{aligned}
$$

Hence, it follows that

$$
\int_\Omega \left[\frac{\partial}{\partial t}\left(\frac{1}{\rho}|\mathbf{v}|^2\right) + 2\mu|\nabla\mathbf{u}|^2\right] d\Omega = 2\int_\Omega \rho\,\mathbf{f}\cdot\mathbf{u}\, d\Omega. \tag{13}
$$

Let

$$
\frac{\mu(\rho)}{\rho} \geq \alpha, \;\; \alpha > 0, \tag{14}
$$

and assume that $\int_\Omega |\nabla\mathbf{u}|^2\, d\Omega \geq \beta \int_\Omega |\mathbf{u}|^2\, d\Omega$ for some $\beta > 0$. Then, from (13) it follows that

$$
\int_\Omega \left[\frac{\partial}{\partial t}\left(\rho|\mathbf{u}|^2\right) + 2\alpha\beta\rho|\mathbf{u}|^2\right] d\Omega \leq 2\int_\Omega \sqrt{\rho}|\mathbf{u}|\sqrt{\rho}|\mathbf{f}|\, d\Omega \leq \alpha\beta\int_\Omega \rho|\mathbf{u}|^2\, d\Omega + \frac{1}{\alpha\beta}\int_\Omega \rho\,|\mathbf{f}|^2\, d\Omega,
$$

that is,

$$\frac{d}{dt}\left(\int\limits_\Omega \rho|\mathbf{u}|^2\,d\Omega\right)+\alpha\beta\int\limits_\Omega \rho|\mathbf{u}|^2\,d\Omega\le\frac{1}{\alpha\beta}\int\limits_\Omega \rho\,|\mathbf{f}|^2\,d\Omega \tag{15}$$

where we have used the Young's, i.e., the weighted Cauchy-Schwarz inequalities. By (12), it follows now

$$\begin{aligned}\frac{d}{dt}\left(\int\limits_\Omega \rho|\mathbf{u}|^2\,d\Omega\right)+\alpha\beta\int\limits_\Omega \rho|\mathbf{u}|^2\,d\Omega&\le\frac{1}{2(\alpha\beta)^2}\int\limits_\Omega \rho(\cdot,t)^2\,d\Omega+\frac{1}{2}\int\limits_\Omega |\mathbf{f}|^4\,d\Omega\\ &\le\frac{1}{2(\alpha\beta)^2}\|\rho_0(\cdot,t)\|^2+\frac{1}{2}\int\limits_\Omega |\mathbf{f}|^4\,d\Omega\equiv C,\end{aligned} \tag{16}$$

where C denotes a generic constant.

As is well known, the solution of the initial value problem $\mathbf{v}'=\sigma\mathbf{v}+\mathbf{f},\ \mathbf{v}(0)=\mathbf{v}_0$ has the form

$$\mathbf{v}(t)=e^{\sigma t}\mathbf{v}_0+\int\limits_0^t e^{\sigma(t-s)}\mathbf{f}\,ds$$

From that expression and estimate (16) we obtain

$$\begin{aligned}\|\rho(\cdot,t)^{1/2}\mathbf{u}(\cdot,t)\|_0^2 &\le e^{-\alpha\beta\,t}\|\rho_0^{1/2}|\mathbf{u}_0|\|_0^2+C\int\limits_0^t e^{-\alpha\beta(t-s)}ds\\ &\le e^{-\alpha\beta\,t}\|\rho_0^{1/2}|\mathbf{u}_0|\,\|_0^2+\tfrac{C}{\alpha\beta}(1-e^{-\alpha\beta t})\le e^{-\alpha\beta\,t}\|\rho_0^{1/2}|\mathbf{u}_0|\,\|_0^2+\tfrac{C}{\alpha\beta}.\end{aligned} \tag{17}$$

If $t=O(\tau)$, i.e., if we perform just one time step, the second term above is bounded by $C\tau=|O(\tau)|$.

If we do not use Young's inequality in (15), we obtain

$$\begin{aligned}\|\rho(\cdot,t)^{1/2}\mathbf{u}(\cdot,t)\|_0^2 &\le e^{-\alpha\beta\,t}\|\rho_0^{1/2}\mathbf{u}_0\|_0^2+\tfrac{1}{\alpha\beta}\int\limits_0^t e^{-\alpha\beta(t-s)}\left(\int\limits_\Omega\rho|\mathbf{f}|^2 d\Omega\right)ds\\ &\le e^{-\alpha\beta\,t}\|\rho_0^{1/2}\mathbf{u}_0\|_0^2+\tfrac{1}{(\alpha\beta)^2}\sup_{t>0}\left(\int\limits_\Omega\rho|\mathbf{f}|^2 d\Omega\right).\end{aligned} \tag{18}$$

Thus, estimates (12)–(18) show unconditional stability of the solution of the continuous problem. Furthermore, it is seen that, under the assumption (14), the influence of the initial condition decreases exponentially to zero with increasing time. These stability estimates are basic for the estimates in the following sections.

3 Time discretization, operator splitting scheme and linearization

The system of equations we consider in the sequel is the following

$$\frac{\partial \rho}{\partial t} + \mathbf{u}\cdot\nabla\rho = 0 \tag{19}$$

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{v} - \nabla\cdot(\mu D(\mathbf{u})) + \nabla p = \rho\,\mathbf{f} \tag{20}$$

$$\nabla\cdot\mathbf{v} - \mathbf{u}\cdot\nabla\rho = 0. \tag{21}$$

with $\mathbf{u} = \frac{1}{\rho}\mathbf{v}$ and appropriate boundary and initial conditions hold.

Equation (21) replaces the classical incompressibility constraint $\nabla \cdot \mathbf{u} = 0$. We refer to (21) as the divergence constraint for the momentum.

Equations (19)–(21) are to be solved on a sequence of time intervals $[t_k, t_k + \tau_k[$, $k = 0, 1, \cdots$, where $t_0 = 0$. The time steps τ_k may vary. The arising systems are nonlinear. Newton's method is applicable but less convenient to use for NS, in particular for convection-dominated (high Reynolds number) flows. Among the most prohibitive factors is the repeated recomputation of the nonlinear Jacobian. For practical reasons it is better to use some form of linearization through an operator splitting method. In principle, it is recommendable to use splittings that can be handled with off-the-shelf software.

The splitting methods used in this paper are motivated by two facts. First, since in general the initial pressure is not known, we must keep the coupled diffusion and divergence constraint in their coupled form intact, which also enables the computation of the pressure without use of (artificial) pressure boundary conditions. Therefore, instead we split off the advection part, which can be handled separately. Hence, the diffusion term with pressure and the divergence constraint (20)–(21), and the advection term (19) will be progressed separately at each time step.

Second, for reasons of stability and to avoid the use of very small time steps, we must use a stable implicit time integration method, preferably of second or higher order of accuracy.

Operator splitting schemes for (19)–(21) have been used since long (cf. [31, 42]). In the search of stable and computationally cheap solution methods it has been suggested to decouple all three equations, resulting in various projection-type methods, see e.g. [19].

In order to decouple the divergence-free constraint of the velocity from the momentum equation, in [14], see also [10], the diffusion-convection term is advanced at each time step without enforcing the incompressibility condition. The resulting, intermediate velocity field is then projected onto the space of discretely divergence-free vector fields. As pointed out in [37], this corresponds to non-physical boundary conditions for the pressure variable. In [37], it is indicated that the resulting pressure is still a reasonable approximation of the true pressure, at least in the interior of the domain Ω. However, as pointed out in [19], while the intermediate solution satisfies the given boundary conditions for the velocity, the projected (divergence-free) velocity does not. Therefore, one is left with a velocity approximation which is divergence-free but does not satisfy proper boundary conditions or an approximation, that satisfies the boundary conditions but is not divergence-free.

To ensure that the unconditional stability of the continuous solution holds also for the discrete solution, in [30] a method involving two projections per time step is proposed. We show that the use of such projections can be avoided if we use the equations for the momentum, $\mathbf{v} = \rho\mathbf{u}$. By the use of this variable instead of the velocity $\mathbf{u}$ one avoids also the problem of defining boundary conditions for $\mathbf{u}$ at a part of the boundary, where there is a vacuum, i.e., where $\rho = 0$. If the velocity is bounded, we can set $\mathbf{v} = \mathbf{0}$ there. Furthermore, since in general the velocity slows down when the flow enters a thicker medium, one can expect that $\mathbf{v}$ has a smoother behaviour, i.e. less strong variations, than $\mathbf{u}$ and can therefore be more accurately approximated numerically. If $\mathbf{f} = \mathbf{0}$, indeed, inequality (18) indicates that the L_2-norm of $\mathbf{u}(\cdot, t)$ slows down if $\rho(\cdot, t)$ increases. Note, that the artificial pressure boundary condition is avoided by solving a saddle point system, keeping equations (2)-(3), respectively (20)–(21) coupled.

The equations can be solved by use of an operator splitting method such that at each time interval we first compute the density, and then solve the momentum equation together with the divergence constraint. Furthermore, we linearize the equations using a "frozen coefficient" approach in a similar way as is done in the Oseen problem. In general, this method leads to a splitting error of first order. In Section 6 we present an alternative method which has a splitting error of second order.

3.1 Time discretization schemes

To introduce notation and formalize the proposed splitting scheme, we start by using the following semi-implicit time discretization scheme.

We find the approximate sequences $\{\rho^n, \mathbf{v}^n, \mathbf{u}^n, p^n\}_{n=0,1,\cdots,N}$ with initial conditions ($\rho^0 = \rho_0, \mathbf{v}^0 = \mathbf{v}_0, \mathbf{u}^0 = \mathbf{v}_0/\rho_0$) and for all time steps n from 0 to $N-1$.

Algorithm 1 (Backward Euler scheme)

A1-1: Compute ρ^{n+1} from

$$\frac{\rho^{n+1} - \rho^n}{\tau} + \mathbf{u}^n \cdot \nabla \rho^{n+1} = 0$$

A1-2: Compute $\mathbf{v}^{n+1}$ and p^{n+1} from

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\tau} + (\mathbf{u}^n \cdot \nabla)\mathbf{v}^{n+1} - \nabla \cdot (\mu^{n+1} D(\frac{\mathbf{v}^{n+1}}{\rho^{n+1}})) + \nabla p^{n+1} = \rho^{n+1}\, \mathbf{f}^{n+1}, \tag{22}$$

$$\nabla \cdot \mathbf{v}^{n+1} - \tau^2 \Delta p^{n+1} = \mathbf{u}^n \cdot \nabla \rho^{n+1}, \tag{23}$$

A1-3: Finally, obtain $\mathbf{u}^{n+1}$ as $\mathbf{u}^{n+1} = \mathbf{v}^{n+1}/\rho^{n+1}$.

In (22), we have additionally regularized the problem by adding the term $-\tau^2 \Delta p^{n+1}$ where $-\Delta$ is the negative Laplacian operator. More details on the regularization are given in Section 4.

The backward Euler scheme is only first order accurate. A straightforward way to obtain a algorithm of second-order of accuracy in time for the density and momentum equations, is to replace the first-order Euler backward time discretization, shown in Algorithm 1 with the three-level backward differentiation formula (BDF2).

The BDF2 scheme for constant time steps processes as following. First, one initializes $(\rho^0, \mathbf{v}^0, \mathbf{u}^0)$, and computes $(\rho^1, \mathbf{v}^1, \mathbf{u}^1, p^1)$ by using one step of the first-order Algorithm 1. Then for $n \geq 1$, proceed as follows.

Algorithm 2 (BDF2)

A2-1: Set the linearly extrapolated velocity at time level $n+1$ as

$$\mathbf{u}^\star = 2\mathbf{u}^n - \mathbf{u}^{n-1}.$$

A2-2: Compute ρ^{n+1} from

$$\frac{3\rho^{n+1} - 4\rho^n + \rho^{n-1}}{2\tau} + \mathbf{u}^\star \cdot \nabla \rho^{n+1} = 0 \tag{24}$$

A2-3: Compute $\mathbf{v}^{n+1}$ and p^{n+1} from

$$\begin{aligned} \frac{3\mathbf{v}^{n+1} - 4\mathbf{v}^n + \mathbf{v}^{n-1}}{2\tau} + (\mathbf{u}^\star \cdot \nabla)\mathbf{v}^{n+1} - \nabla \cdot (\mu^{n+1} D(\frac{\mathbf{v}^{n+1}}{\rho^{n+1}})) + \nabla p^{n+1} &= \rho^{n+1}\, \mathbf{f}^{n+1}, \\ \nabla \cdot \mathbf{v}^{n+1} - \tau^2 \Delta p^{n+1} &= \mathbf{u}^\star \cdot \nabla \rho^{n+1}, \end{aligned} \tag{25}$$

A2-4: Finally, recover the velocity $\mathbf{u}^{n+1}$ as $\mathbf{u}^{n+1} = \mathbf{v}^{n+1}/\rho^{n+1}$.

BDF2 is second order accurate in time for constant time steps is simple to implement. However, it is not fully stable in the sense of A- and B-stability for systems of ordinary differential equations (cf. e.g., [48]). Here A-stability refers to stability for linear systems,

$$\frac{d\xi(t)}{dt} + A\xi(t) = 0,\ t > 0,\ \xi(0) = \xi_0,$$

where A is a diagonalizable matrix with all eigenvalues in the positive real part of the complex plane. B-convergence refers to stability for, in general, nonlinear equations,

$$\frac{d\xi(t)}{dt} + F(t, \xi) = 0,\ t > 0,\ \xi(0) = \xi_0,$$

where F is monotone, that is $(F(t, \xi) - F(t, \eta), \xi - \eta) \geq 0$ for all $\{\xi, \eta\}$ in the domain of consideration. The numerical method is said to be stable if

$$\|\xi_n - \eta_n\| \leq \|\xi_{n-1} - \eta_{n-1}\|,\ n = 1, 2, \cdots$$

for all solutions $\{\xi_n, \eta_n\}$, corresponding to different initial values. The stability analysis for nonlinear problems is more complicated and one can not just rely on eigenvalues of the linearized (Jacobian) operator. It can be shown that methods, such as BDF2 or the traditional form of the trapezoidal method are not fully stable. This prevents the use of the methods for long time intervals.

In [2] it has been shown that the so-called '*one-leg*' (or '*one-sided*)' form of the θ-method for $\theta \leq 1/2 - |O(\tau]|$ is stable for monotone operators uniformly in time and, hence, is applicable for infinitely long time integration intervals. It has a second order of accuracy if $\theta = 1/2 - |O(\tau]|$, where τ is the time step. Because of its simplicity, when we do not need to integrate on very long time intervals, as well as for reasons of comparison with other related work, such as [17], the BDF2 method is also used in the numerical experiments.

We first briefly recall the one-leg θ-method (OLTM). The implicit one-leg form of the classical θ-method reads as follows:

$$\begin{aligned} \xi(t+\tau) - \xi(t) + \tau F(\bar{t}, \bar{\xi}) &= 0, \quad t = 0, \tau, 2\tau, \cdots \\ \xi(0) &= \xi_0, \end{aligned} \tag{26}$$

where τ is the time step and

$$\begin{aligned} &\bar{t} = \theta t + (1-\theta)(t+\tau) \\ &\bar{\xi} = \theta \xi(t) + (1-\theta)\xi(t+\tau), 0 \leq \theta \leq 1. \end{aligned}$$

The method can be written as an Euler backward (implicit) step, $t \to \bar{t}$, $\bar{t} = t + \theta\tau$,

$$\xi(\bar{t}) + \theta\tau F(\bar{t}\xi(\bar{t})) = \xi(t),$$

followed by an Euler forward (explicit) step, $\bar{t} \to t + \tau$

$$\xi(t+\tau) + (1-\theta)\tau F(\bar{t}, \xi(\bar{t})) = \mathbf{v}(\bar{t}).$$

These equations follow if we multiply (26) by θ and define $\mathbf{v}(\cdot)$ as a linear function in each interval $\bar{t} \to t + \tau$, so that $\overline{\mathbf{v}}(t) = \mathbf{v}(\bar{t})$. We refer to [2, 6] and [45] for more details on the one-leg form of the θ-scheme and its properties.

We present next an implementation of the one-leg form of the θ-scheme for the density and the momentum equation. Note that we need to compute $(\mathbf{u}(t_n + \tau/2), p(t_n + \tau/2))$, which is done by solving a Stokes problem of the form as in (27). To additionally simplify the computational procedure, we split the momentum $\mathbf{v}^{n+1}$ into two parts, i.e., $\mathbf{v}^{n+1} = (\mathbf{v}_1^{n+1} + \mathbf{v}_2^{n+1})/2$, where the component $\mathbf{v}_1^{n+1}$ recovers the convective character as in (29) and the other component $\mathbf{v}_2^{n+1}$ takes care of the diffusion property and the incompressible constrain as in (30).

We choose $\theta = 1/2$ to guarantee second order accuracy in time.

Algorithm 3 (OLTM)

A3-1: Compute $\mathbf{v}(t_n+\tau/2), \mathbf{u}(t_n+\tau/2), p(t_n+\tau/2)$ by solving

$$\begin{aligned}\frac{\mathbf{v}^{n+\frac{1}{2}}-\mathbf{v}^n}{\tau/2} - \nabla\cdot(\mu^n D(\frac{\mathbf{v}^{n+\frac{1}{2}}}{\rho^n})) + \nabla p^{n+\frac{1}{2}} &= \rho^n\,\mathbf{f}^{n+\frac{1}{2}} - (\mathbf{u}^n\cdot\nabla)\mathbf{v}^n,\\ \nabla\cdot\mathbf{v}^{n+\frac{1}{2}} - \tau^2\Delta p^{n+\frac{1}{2}} &= \mathbf{u}^n\cdot\nabla\rho^n,\end{aligned} \tag{27}$$

with $\mathbf{u}^{n+\frac{1}{2}} = \mathbf{v}^{n+\frac{1}{2}}/\rho^n$.

A3-2: Compute ρ^{n+1} by solving

$$\frac{\rho^{n+1}-\rho^n}{\tau} + \mathbf{u}^{n+\frac{1}{2}}\cdot\nabla\frac{\rho^{n+1}+\rho^n}{2} = 0. \tag{28}$$

A3-3: Define $\rho^{n+\frac{1}{2}} = (\rho^{n+1}+\rho^n)/2$ and $\mu^{n+\frac{1}{2}} = (\mu^{n+1}+\mu^n)/2$, and compute $(\mathbf{v}_1^{n+1}, \mathbf{v}_2^{n+1}, \tilde{p}^{n+1})$ by solving

$$\frac{\mathbf{v}_1^{n+1}-\mathbf{v}^n}{\tau} + \mathbf{u}^{n+\frac{1}{2}}\cdot\nabla\frac{\mathbf{v}_1^{n+1}+\mathbf{v}^n}{2} = \rho^{n+\frac{1}{2}}\,\mathbf{f}^{n+\frac{1}{2}} + \nabla\cdot(\mu^{n+\frac{1}{2}}D(\mathbf{u}^{n+\frac{1}{2}})) - \nabla p^{n+\frac{1}{2}}, \tag{29}$$

$$\begin{aligned}\frac{\mathbf{v}_2^{n+1}-\mathbf{v}^n}{\tau} - \nabla\cdot(\mu^{n+\frac{1}{2}}D(\frac{\mathbf{v}_2^{n+1}}{\rho^{n+1}} + \mathbf{u}^n)/2) + \nabla\tilde{p}^{n+1} &= \rho^{n+\frac{1}{2}}\,\mathbf{f}^{n+\frac{1}{2}} - (\mathbf{u}^{n+\frac{1}{2}}\cdot\nabla)\frac{\mathbf{v}_1^{n+1}+\mathbf{v}^n}{2},\\ \nabla\cdot\mathbf{v}_2^{n+1} - \tau^2\Delta\tilde{p}^{n+1} &= 2\mathbf{u}^{n+\frac{1}{2}}\cdot\nabla\rho^{n+1} - \nabla\cdot\mathbf{v}_1^{n+1}.\end{aligned} \tag{30}$$

A3-4: Finally, we compute $(\mathbf{v}^{n+1}, \mathbf{u}^{n+1}, p^{n+1})$ as

$$\mathbf{v}^{n+1} = \frac{\mathbf{v}_1^{n+1}+\mathbf{v}_2^{n+1}}{2},\ \ \mathbf{u}^{n+1} = \frac{\mathbf{v}^{n+1}}{\rho^{n+1}},\ \ p^{n+1} = \frac{\tilde{p}^{n+1}+p^{n+\frac{1}{2}}}{2}.$$

The form of the constraint in (30) is motivated by the relation

$$\nabla\cdot\mathbf{v}^{n+1} = \nabla\cdot\frac{\mathbf{v}_1^{n+1}+\mathbf{v}_2^{n+1}}{2} = \nabla\cdot(\rho^{n+1}\mathbf{u}^{n+1}) = \mathbf{u}^{n+1}\cdot\nabla\rho^{n+1} + \rho^{n+1}\nabla\cdot\mathbf{u}^{n+1}.$$

Using the assumption $\nabla\cdot\mathbf{u}^{n+1} = 0$ one obtains

$$\nabla\cdot\mathbf{v}_2^{n+1} = 2\mathbf{u}^{n+1}\cdot\nabla\rho^{n+1} - \nabla\cdot\mathbf{v}_1^{n+1}.$$

Since $\mathbf{u}^{n+1}$ is not known, we use $\mathbf{u}^{n+\frac{1}{2}}$ to replace it.

Table 1 summarizes the computational complexity of Algorithm 2 and Algorithm 3 at each time level. As is well known, to precondition and solve the Stokes problem is much easier than to precondition and solve the Oseen problem, especially for small values of the viscosity. Thus, the efficient preconditioned iterative solutions of the Stokes equations will pay off the heavier assembling and computing work in OLTM, which, we believe, can still remain an attractive method to use.

Table 1: Comparison of the computational complexity of BDF2 and the one-leg θ-scheme

BDF2	One-leg θ-scheme
Solve the hyperbolic equations as in (24) Solve the Oseen type problem as in (25)	Solve the hyperbolic equations twice as in (28),(29) Solve the Stokes type problem twice as in (27),(30)
Reassemble matrices corresponding to $\mathbf{u}^\star\cdot\nabla$, $\nabla\cdot(\mu^{n+1}D)$	Reassemble matrices corresponding to $\mathbf{u}^n\cdot\nabla, \mathbf{u}^{n+\frac{1}{2}}\cdot\nabla$, $\nabla\cdot(\mu^n D), \nabla\cdot(\mu^{n+\frac{1}{2}}D)$
	Recompute more right hand vectors

3.2 Operator splitting and related splitting error

As already stated, equations (19)–(21) will be solved by use of an operator splitting, where at each time interval we first compute the density (19) and then solve the momentum equation (20) together with the divergence constraint (21). Furthermore, we linearize the equations using a ”frozen coefficient” approach in a similar way as is done in the Oseen problem.

To this end, consider a time interval $(t_0, t_0 + \tau)$. Denote the already computed values at t_0 by $\mathbf{u}_0$, μ_0 and ρ_0 and let $\widetilde{\mathbf{u}}$, $\widetilde{\mathbf{v}}$, $\widetilde{p}$, $\widetilde{\mu}$ and $\widetilde{\rho}$ be the correspondingly computed values after linearization and operator splitting. Then, for $t \in (t_0, t_0 + \tau)$ we solve

$$\begin{aligned} \frac{\partial\widetilde{\rho}}{\partial t} + \nabla\cdot(\mathbf{u}_0\widetilde{\rho}) &= 0 && (31)\\ \frac{\partial\widetilde{\mathbf{v}}}{\partial t} + \mathbf{u}_0\cdot\nabla\widetilde{\mathbf{v}} - \nabla\cdot(\mu_0 D(\widetilde{\mathbf{u}})) + \nabla\widetilde{p} &= \widetilde{\rho}\,\mathbf{f} && (32)\\ \nabla\cdot\widetilde{\mathbf{v}} - \mathbf{u}_0\cdot\nabla\widetilde{\rho} &= 0 && (33) \end{aligned}$$

Here we solve first (31) and then (32)–(33). Further, $\widetilde{\mathbf{u}} = \widetilde{\rho}^{-1}\widetilde{\mathbf{v}}$.

The relation $\widetilde{\mu} = \mu(\widetilde{\rho})$ is assumed to be known. The above relations do not provide a divergence-free velocity field. For this purpose we use the mass conservation equation for the density a second time to solve

$$\frac{\partial\widetilde{\widetilde{\rho}}}{\partial t} + \nabla\cdot(\widetilde{\mathbf{u}}\widetilde{\widetilde{\rho}})0.$$

Assuming that we use quadratic basis functions for $\mathbf{v}$, we can form a linear combination of the quadratic polynomial interpolants of the functions $\frac{1}{\rho_0}\widetilde{\mathbf{v}}$, $\frac{1}{\widetilde{\rho}}\widetilde{\mathbf{v}}$ and $\frac{1}{\widetilde{\widetilde{\rho}}}\widetilde{\mathbf{v}}$, to make in each element $\widetilde{\widetilde{\mathbf{u}}} = (\alpha\frac{1}{\rho_0} + \beta\frac{1}{\widetilde{\rho}} + \gamma\frac{1}{\widetilde{\widetilde{\rho}}})\widetilde{\mathbf{v}}$ satisfy $\nabla\cdot\widetilde{\widetilde{\mathbf{u}}} = 0$ This leads to three equations to solve for the coefficients α, β and γ.

To derive an error estimates of the operator splitting and linearization errors, we assume that the error in the velocity, $\mathbf{u}-\widetilde{\mathbf{u}}$ is divergence-free. For simplicity of notation the approximations are also denoted by $\widetilde{\rho}, \widetilde{\mathbf{u}}, \widetilde{\mathbf{v}}$.

We subtract (31) from (19), (32) from (20) and (33) from (21). Since for any vectors $\mathbf{a}, \mathbf{b}, \widetilde{\mathbf{a}}, \widetilde{\mathbf{b}}$ it holds

$$\mathbf{a}\cdot\mathbf{b}-\widetilde{\mathbf{a}}\cdot\widetilde{\mathbf{b}}=(\mathbf{a}-\widetilde{\mathbf{a}})\cdot\mathbf{b}+\widetilde{\mathbf{a}}\cdot(\mathbf{b}-\widetilde{\mathbf{b}})=(\mathbf{a}-\widetilde{\mathbf{a}})\cdot\mathbf{b}+\mathbf{a}\cdot(\mathbf{b}-\widetilde{\mathbf{b}})-(\mathbf{a}-\widetilde{\mathbf{a}})\cdot(\mathbf{b}-\widetilde{\mathbf{b}}),$$

we obtain

$$\frac{\partial}{\partial t}(\rho-\widetilde{\rho})-\mathbf{u}\cdot\nabla(\rho-\widetilde{\rho})=(\mathbf{u}-\mathbf{u}_0)\cdot\nabla\rho-(\mathbf{u}-\mathbf{u}_0)\cdot\nabla(\rho-\widetilde{\rho})\equiv\Psi_1 \tag{34}$$

$$\begin{aligned}\frac{\partial}{\partial t}(\mathbf{v}-\widetilde{\mathbf{v}}) \;&+\; \mathbf{u}\cdot\nabla(\mathbf{v}-\widetilde{\mathbf{v}})-\nabla\cdot(\mu D(\mathbf{u}-\widetilde{\mathbf{u}}))+\nabla(p-\widetilde{p})\\ &=\; -(\mathbf{u}-\mathbf{u}_0)\cdot\nabla\mathbf{v}+(\mathbf{u}-\mathbf{u}_0)\cdot\nabla(\mathbf{v}-\widetilde{\mathbf{v}})+\nabla\cdot((\mu-\mu_0)D(\mathbf{u}))\\ &\quad\; -\nabla\cdot((\mu-\mu_0)D(\mathbf{u}-\widetilde{\mathbf{u}}))+(\rho-\rho_0)\mathbf{f}\equiv\Psi_2\end{aligned} \tag{35}$$

$$\nabla\cdot(\mathbf{v}-\widetilde{\mathbf{v}})-\mathbf{u}\cdot\nabla(\rho-\widetilde{\rho})=(\mathbf{u}-\mathbf{u}_0)\cdot\nabla\rho-(\mathbf{u}-\mathbf{u}_0)\cdot\nabla(\rho-\widetilde{\rho})\equiv\Psi_1 \tag{36}$$

Thus, it holds that $\frac{\partial}{\partial t}(\rho-\widetilde{\rho})-\nabla\cdot(\mathbf{v}-\widetilde{\mathbf{v}})=0$.

Here we have collected all error terms from the previous time step and secind order error terms on the right hand side.

In order to derive error bounds for $\mathbf{v}-\widetilde{\mathbf{v}}$ and $\rho-\widetilde{\rho}$, we take then a variational formulation in (35) multiplying with functions $\mathbf{v}-\widetilde{\mathbf{v}}$, $p-\widetilde{p}$ and $\rho-\widetilde{\rho}$. In addition, we rewrite $\mathbf{u}-\widetilde{\mathbf{u}}$ in the form

$$\mathbf{u}-\widetilde{\mathbf{u}}=\frac{1}{\rho}\mathbf{v}-\frac{1}{\widetilde{\rho}}\widetilde{\mathbf{v}}=\frac{1}{\rho}(\mathbf{v}-\widetilde{\mathbf{v}})-\Phi$$

where

$$\Phi=\left(\frac{1}{\widetilde{\rho}}-\frac{1}{\rho}\right)\mathbf{v}+\left(\frac{1}{\rho}-\frac{1}{\widetilde{\rho}}\right)(\mathbf{v}-\widetilde{\mathbf{v}}).$$

The terms included in Φ will then go to the right hand side, resulting from (35). As in the derivation of (13), we obtain

$$\begin{aligned}&\int_\Omega\left[\frac{\partial}{\partial t}\left(\frac{1}{\rho}|\mathbf{v}-\widetilde{\mathbf{v}}|^2\right)+2\mu|\nabla(\mathbf{u}-\widetilde{\mathbf{u}}|^2\right]d\Omega\\ \le\;&\int_\Omega\frac{1}{\rho}\,\Psi\cdot(\mathbf{u}-\widetilde{\mathbf{u}})\,d\Omega+\int_\Omega\left[\frac{\partial}{\partial t}(\mathbf{v}-\widetilde{\mathbf{v}})+\mathbf{u}\cdot\nabla(\mathbf{v}-\widetilde{\mathbf{v}})\right]\Phi\,d\Omega.\end{aligned}$$

As in the derivation of (18), after integration from t_0 to $t_0+\tau$, the latter shows that

$$\left\|\frac{1}{\sqrt{\rho(\cdot,t)}}|\mathbf{v}-\widetilde{\mathbf{v}}|(\cdot,t)^2\right\|+\alpha\beta\left\|\sqrt{\rho(\cdot,t)}|\mathbf{v}-\widetilde{\mathbf{v}}(\cdot,t)|\right\|^2\le\tau\,\mathrm{rhs},\ t_0<t<t_0+\tau,. \tag{37}$$

A similar bound can be shown for $\|(\rho - \widetilde{\rho})(\cdot, t)\|$. Here we have used Young's inequality and moved terms involving $\mathbf{v} - \widetilde{\mathbf{v}}$ and $\mathbf{u} - \widetilde{\mathbf{u}}$ to the left hand side. The factor τ comes from the integration from t_0 to $t \leq t_0 + \tau$.

The rhs involves terms including factors such as $\|(\mathbf{u} - \mathbf{u}_0)(\cdot, t)\|$ and $\|(\rho - \rho_0)(\cdot, t)\|$, $t_0 < t < t_0 + \tau$. We assume that enough regularity of the solutions $\mathbf{v}$, $\mathbf{u}$ and ρ hold so that these terms are $O(\tau^{\xi})$, $0 < \xi \leq 1$.

Let

$$e_n = \left\| \frac{1}{\sqrt{\rho(\cdot, t_n)}} |(\mathbf{v} - \widetilde{\mathbf{v}})(\cdot, t_n)| \right\| + \|(\rho - \widetilde{\rho})(\cdot, t_n)\| .$$

Then, using a number of Young's inequalities to bound the remaining part of rhs, it follows from (37) that the following bound holds,

$$e_{n+1} \leq e^{-\alpha\beta\tau} e_n + \tau^{1+\xi} C, n = 0, 1, \cdots$$

where C is a constant, depending on the regularity of the solution and $\mathbf{f}$.

By recursion, this shows that

$$e_{n+1} \leq e^{-\alpha\beta\tau} e_0 + C\tau^{1+\xi} \Sigma_{k=0}^{n} e^{-\alpha\beta k\tau} \leq e^{-\alpha\beta\tau} e_0 + C\tau^{1+\xi}/(1 - e^{-\alpha\beta\tau}) \leq e^{-\alpha\beta\tau} e_0 + \frac{C}{\alpha\beta} \tau^{\xi}$$

where $T = n\tau$ (a corresponding bound can be proved even if the time steps vary).

Since by assumption $e_0 = O(\tau^{1+\xi})$, we have then

$$e_n \leq \frac{C}{\alpha\beta} \tau^{\xi}.$$

Hence, if the regularity of the solution permits local errors of second order, i.e. $\xi = 1$, then the splitting and linearization error is of first order, $O(\tau)$.

A similar bound for pressure p can be derived. The above bounds are derived for the continuous equations and show that we have full control of the error due to the operator splitting and the linearization during integration over all time intervals.

4 Regularization/stabilization issues

It remains now to solve the linearized problem (31)-(33).

These equations shall be solved by use of a stable time-stepping method - BDF2 or OLTM.

The handling of the nonlinearity of the problem will be controlled by choosing sufficiently small time steps. For the space discretization at each time step ($t \to t + \tau$) we use a finite element mesh and proper basis functions. At each time step we replace then the initial values of $\mathbf{u}$ and μ in (31), (32), and ρ in (19) with the computed values at time t of the previous time step ($t - \tau \to t$). This must be done at each time interval $(t, t + \tau)$. We note that the approximation we have used corresponds to an operator splitting method.

Using equation (31), we can compute the density after we have computed the momentum, respectively, the velocity. This holds also for the viscosity. Therefore it remains to solve the linearized N-S equation (equations (32)) for the momentum and the pressure with the divergence constraint (33).

In general, the solution of the discrete problem (31)-(33) may require several different stabilization terms.

4.1 Stabilization of the momentum equation

For problems with dominant convection (high Reynolds numbers) we need to stabilize the convective term in (32). Which technique to use and what are the consequences of the stabilization is among the research topics, which have been dealt for years and is still not fully resolved.

In general, the flow is advection-dominated, that is, $\mathbf{u}/\mu$ takes large values. If the elementwise Peclet number, $Pe = h\,\mathbf{u}_h/\mu$, is larger than one, then spurious node-to-node oscillations occur around steep gradients. Various techniques to avoid this can be used. The simplest is to use backward first order approximation of the advection term, which however degrades the accuracy of the discrete solution. The streamline upwind Petrov-Galerkin method (see, e.g., [22]) improves on this but that entails complicated codes to account for moving gradients in time and the necessity to dynamically refine and de-refine the mesh to keep the number of degrees of freedom as few as possible. If one uses a finite element mesh defined by the streamlines, one can use local Green's functions (see [5, 6]) to damp out the spurious oscillations.

Choosing the mesh size sufficiently small to satisfy the condition regarding the local Peclet number can lead to very small elements and large linear systems to be solved. In practice it suffices to use locally refined meshes where there are steep gradients. In this paper we do not discuss the above mentioned methods any further but assume that the mesh is locally sufficiently fine to avoid the appearance of such oscillations.

Somewhat underestimated are instabilities due to poor mass conservation in equation (32). For example, it is known that Taylor-Hood elements, which we use in the numerical experiments, are globally but not locally mass conservative, cf. e.g., [28, 1]. It is shown in [28] that in certain cases this may have dramatic consequences. Taylor-Hood elements are usually stabilized by adding to so-called *grad-div* term, the continuous form of which is $\gamma\nabla(\nabla\cdot)$, γ being some stabilization constant ([34]). It is argued, cf. e.g. [28], that for Taylor-Hood element discretizations, for high Reynolds numbers, due to loss of divergence-free velocity, numerical errors are amplified by a factor $1/\mu$. This stabilization is often neglected in practice since the resulting linear systems become more difficult to solve (see also [25]).

4.2 Stabilization of the density equation

Equation (31) also needs stabilization. With no claims to be exhaustive on this topic, we mention the technique, used in [17] and analysed in [20], referred to as the *entropy* *viscosity method*, which has been shown to be among the most efficient techniques, known today. Stabilization turns out to be less crucial for problems, where ρ is smooth enough. Therefore in the numerical experiments, performed here, (31) is solved with no additional stabilization.

4.3 Stabilizing the saddle point problem

Stabilization is required to avoid instabilities if the discrete LBB condition does not hold, which results in pressure oscillations.

Problem (32)–(33) to be discretized is in the form of an Oseen problem, namely, linearized Navier-Stokes problem with variable viscosity. For error estimates of space discretized problems of this type we refer, for instance, to [3, 4].

To avoid locking in the pressure variable we use an inf-sup, i.e., Ladyzhenskaya-Babuška-Brezzi (LBB) stable finite element pair of discrete subspaces V_h and P_h and basis functions for $\mathbf{v}$ and p. The LBB stability means that

$$\inf_{p_h \in P_h} \sup_{\mathbf{v}_h \in V_h} \frac{(p_h \nabla \cdot \mathbf{v}_h)_\Omega}{\|p_h\|_0 \, \|\mathbf{v}_h\|_{H^1(\Omega)}} \geq \gamma > 0$$

for some γ, that does not depend on the space discretization parameter h.

As it turns out, see e.g. [35], also [4], in some cases this does not fully avoid unphysical oscillations. For this, but also for more general reasons, as we shall see, we additionally regularize the problem as follows. We add namely the term $(1-2\nu)p$ or, better, $(1-2\nu)(-\Delta p)$, to $\nabla \cdot (\rho^{-1}\mathbf{v}) = 0$, forming the perturbed equation

$$\nabla \cdot (\mathbf{v}) + (1-2\nu)p = \mathbf{u} \cdot \rho \tag{38}$$

or

$$\nabla \cdot (\mathbf{v}) + (1-2\nu)(-\Delta p) = \mathbf{u} \cdot \rho \tag{39}$$

Here $-\Delta$ is the negative Laplacian operator and ρ $\mathbf{u}_0$ are already computed.

We remark that a somewhat similar perturbation has been used in [43], see also [14]. In [43], the divergence constraint $\nabla \cdot \mathbf{u} = 0$ is perturbed to

$$\varepsilon \frac{\partial p_\varepsilon}{\partial t} + \nabla \cdot \mathbf{u}_\varepsilon = 0$$

where $\varepsilon > 0$ is a small perturbation parameter. After time discretization this corresponds to the equation

$$\nabla \cdot \mathbf{u}_\varepsilon + \varepsilon/\tau \, p(t+\tau) = \varepsilon/\tau \, p(t).$$

One can choose $\varepsilon = O(\tau^2)$. In this setting, however, the initial pressure at $t = 0$ is needed but is in general not available. Furthermore, since we use a perturbation based on the momentum equation, as we shall see below, it does actually not lead to a perturbation of the solution, but is just a regularization of the expression for the divergence of the momentum variable.

With these choices of stabilization terms we do not have to impose the earlier mentioned condition $\int_\Omega p\, d\Omega = 0$ but instead we can fix the pressure in some boundary point to make it positive in Ω. Actually, this is not needed when we know the null vector corresponding to p so the pressure component, corresponding to that vector can be filtered out afterwards to obtain a unique pressure solution. Since it is the gradient of the pressure that enters the Navier-Stokes equation, there is however no problem having chosen a pressure scale, leading to negative values at some points.

Further in (38), (39), $0 < \nu < 1/2$ and ν can be seen as the so-called Poisson ratio used in elasticity problems, which characterizes the compressibility of the medium. The (volume-wise) incompressibility is described by $\nu = 1/2$ and near incompressibility – by values of ν closer to 1/2. In order to make the perturbation small, we let ν be close to 1/2 and take $\nu = 1/2 - |O(\tau)|$ so that $1 - 2\nu = |O(\tau)|$. This choice can also be motivated by equation (32) itself, $\nabla\cdot\mathbf{v} - \nabla\rho\cdot\mathbf{u} = 0$. If we consider $\mathbf{u}$ as an increase (or decrease) of the velocity, then we expect that $-\nabla\rho\cdot\mathbf{u} \geq 0$, since the velocity normally increases or decreases with decreasing, respectively increasing density of the fluid. Therefore, $1 - 2\nu$ should be taken positive, as we have also taken it.

Note that $\frac{\partial\mathbf{v}}{\partial t}$ and ∇p have the same unit, namely, mass per unit volume times acceleration. Therefore, as we have already chosen it, $1 - 2\nu$ has the dimension of the time unit (s). Hence, since $\Delta = \nabla\cdot\nabla$, the choice (39) is the formally correct one as far as units are concerned.

It is possible to let $\nu = \nu_\ell$ be variable so that

$$\int\limits_{\Omega_{e_\ell}} (1-2\nu)p^2\, d\Omega = \left| \int\limits_{\Omega_{e_\ell}} p\nabla\rho\cdot\mathbf{u}\, d\Omega \right|,$$

respectively

$$\int\limits_{\Omega_{e_\ell}} (1-2\nu)|\nabla p|^2\, d\Omega = \left| \int\limits_{\Omega_{e_\ell}} p\nabla\rho\cdot\mathbf{u}\, d\Omega \right|$$

holds within each element e_ℓ. However, since the velocity and pressure change with time, to avoid recalculation of ν we may prefer to use fixed values of ν or ν_ℓ, computed, say, at the initial time step.

This relation can also be used to choose a proper value of p at some boundary point to make p positive on Ω. As we have already remarked, however, it is not necessary to have positive computed values of p.

5 Block-matrix structure and preconditioning

The saddle point problem to be solved at each time step is

$$\begin{aligned}\frac{\partial \mathbf{v}}{\partial t} + \mathbf{u}_0\cdot\nabla\mathbf{v} - \nabla\cdot(\mu_0 D(\mathbf{u})) + \nabla p &= \rho\,\mathbf{f}\\ \nabla\cdot\mathbf{v} + (1-2\nu)(-\Delta p) &= \mathbf{u}_0\cdot\nabla\rho\end{aligned}$$

or with $-\Delta p$ replaced by p in the second equation. Here $\mathbf{u} = \frac{1}{\rho_0}\mathbf{v}$, where ρ_0 is constant in each element. (For more clear notations, in this section we drop the notation $\tilde{}$.)

After time discretization (for simplicity we present here only the Euler backward method) we must solve

$$\begin{aligned}\mathbf{v}(t+\tau) + \tau\mathbf{u}_0\cdot\nabla\mathbf{v}(t+\tau) - \qquad&\\ \tau\nabla\cdot(\mu_0 D(\tfrac{1}{\rho(t+\tau)}\mathbf{v}(t+\tau)) + \tau\nabla p(t+\tau) &= \mathbf{v}(t) + \tau\rho(t+\tau)\,\mathbf{f}(t+\tau)\\ \nabla\cdot\mathbf{v}(t+\tau) + (1-2\nu)(-\Delta p(t+\tau)) &= \mathbf{u}_0\cdot\nabla\rho(t+\tau)\end{aligned} \tag{40}$$

Clearly, the time step τ can vary between time steps but for notational simplicity we use a fixed notation.

After discretizing in space using some proper finite element pair, such as the Taylor-Hood elements, we obtain a block matrix structure of the algebraic system of the following form

$$\mathcal{A}\begin{bmatrix}\mathbf{v}_h(t+\tau)\\ p_h(t+\tau)\end{bmatrix} = \text{rhs}, \quad \text{where} \quad \mathcal{A} = \begin{bmatrix}A & B^T\\ -B & (1-2\nu)C\end{bmatrix}. \tag{41}$$

Here, $\mathbf{v}_h$, p_h denote the corresponding finite element approximations. For simplicity, we assume also that ν is constant. The matrix block A has the form $A = M + \tau E$, where M is the velocity mass matrix and E comes from the discrete diffusion and convection terms. The block B arises from the divergence operator and C is the discrete Laplacian operator or the pressure mass matrix, depending on the choice (39) or (38), respectively.

As is well known, for inf-sup stable finite element pairs the matrix B has full rank implying a unique solution of the system (32).

Due to its large size, systems with $\mathcal{A}$ are solved by a suitable preconditioned iterative method, in this case a generalized conjugate gradient method, such as GMRES ([36]) or GCG ([3]). Among the best known preconditioners for saddle point matrices are those, based on some approximate block factorization of $\mathcal{A}$.

For the sake of readability, we briefly recall the preconditioners, based on the exact block-factorization of a matrix in a two-by-two block form. Consider a general nonsingular saddle point matrix

$$\mathcal{A} = \begin{bmatrix}A & B^T\\ B & -C\end{bmatrix}$$

where A, C are nonsingular and B is full rank. (In our particular case C is symmetric positive semi-definite but when constructing the preconditioner we can make it nonsingular by adding a properly chosen diagonal matrix.) The matrix $\mathcal{A}$ possesses the following exact matrix factorizations:

$$\mathcal{A}=\begin{bmatrix}A & 0\\ B & -S_A\end{bmatrix}\begin{bmatrix}I_1 & A^{-1}B^T\\ 0 & I_2\end{bmatrix}\quad\text{and}\quad\mathcal{A}=\begin{bmatrix}S_C & B^T\\ 0 & -C\end{bmatrix}\begin{bmatrix}I_1 & 0\\ C^{-1}B & I_2\end{bmatrix}$$

where $S_A=C+BA^{-1}B^T$ and $S_C=A+B^TC^{-1}B$ are the corresponding Schur complement matrices, and I_1, I_2 are identity matrices of proper order. For saddle point problems, to precondition $\mathcal{A}$ it suffices to use an approximation of the block lower-triangular factor, $\begin{bmatrix}A & 0\\ B & -S_A\end{bmatrix}$ or of the block upper-triangular factor $\begin{bmatrix}S_C & B^T\\ 0 & -C\end{bmatrix}$ (cf. e.g. [8]). The quality of the preconditioner depends on how accurate we solve systems with A, respectively, C and how well we approximate S_A and S_C.

As is well-known, the most difficult problem is to find good quality approximations of the Schur complement matrices. Further, systems with the matrix A can be somewhat costly to solve due to the term E, since it can be strongly nonsymmetric.

In the choice of the two possible block factorizations of $\mathcal{A}$, we could choose therefore the one, that regularizes the pivot block, that is

$$\mathcal{A}=\begin{bmatrix}A+(1-2\nu)^{-1}B^TC^{-1}B & B^T\\ 0 & (1-2\nu)C\end{bmatrix}\begin{bmatrix}I_1 & 0\\ -\frac{1}{1-2\nu}C^{-1}B & I_2\end{bmatrix}, \tag{42}$$

where I_1 and I_2 are identity matrices of proper order. Note, that this is in fact an exact factorization. In addition to two solves with the matrix C, an application of this factorization requires a solution with the matrix $\widetilde{A}=A+(1-2\nu)^{-1}B^TC^{-1}B$. Since the target problems can have very large scale, all systems are to be solved by inner iterations to some relative stopping criterion. Because of the latter, it might be advisable to once repeat the outer process in a defect-correction manner.

As there are several possible efficient methods to precondition C, the iterative solutions of the arising systems with the matrix C do not cause any problem. Further, if we use rough stopping criteria for the inner iterations or delete the right block matrix factor in (42), it is advisable to use a flexible outer iteration method, such as GCG with a variable preconditioner (c.f. [9] and [46]) or a flexible variant of GMRES (c.f. [38]).

In earlier papers we have shown that the Augmented Lagrangian techniques are applicable for stationary NS problems with constant and variable viscosity, see [24, 25].

In this particular setting, however, due to the fact that the problem is time-dependent, the pivot block A includes a mass matrix, which in general improves the conditioning of the block. Furthermore, we use a stabilized form of the saddle point matrix. This enables us to approximate S_A and to use the classical lower-triangular preconditioner

$$\mathcal{P}=\begin{bmatrix}A & 0\\ B & -\widetilde{S}\end{bmatrix}$$

where we directly approximate $\widetilde{S}^{-1}$ as

$$\widetilde{S}^{-1}=-\frac{1}{\tau}L_p^{-1}-\mu\widetilde{M}_p^{-1},$$

where $\widetilde{M_p}$ is the diagonal part of the mass matrix for the pressure unknowns. It is trivial to solve linear systems with the diagonal matrix $\widetilde{M_p}$, and for the linear systems with L_p we use an off-the-shelf aggregation-based multigrid solver. The above approximation of S_A^{-1} is analyzed in [32] and [45], also see [11] and the related references therein.

6 A second order operator splitting method

When $\mathbf{v}_h(t+\tau)$ and $p_h(t+\tau)$ have been computed, we can compute $\rho_h(t+\tau)$ and $\mu_h(t+\tau)$, and continue with the next time step.

This corresponds to a sequential operator splitting method, see, for example [19] for discussions on operator splitting methods. The global error, due to this operator splitting method is $O(\tau)$. A symmetric operator splitting method was proposed in [42] and [31], which leads to second order ($O(\tau^2)$) global errors.

Here we present an alternative method. We consider then a nonlinear system consisting of two coupled equations.

$$\begin{aligned} \frac{d\mathbf{u}^{(1)}(t)}{dt} &= \mathcal{A}^{(1)}(t, \mathbf{u}^{(1)}(t), \mathbf{u}^{(2)}(t)), \\ \frac{d\mathbf{u}^{(2)}(t)}{dt} &= \mathcal{A}^{(2)}(t, \mathbf{u}^{(1)}(t), \mathbf{u}^{(2)}(t)), \end{aligned} \tag{43}$$

$t > 0$, where $\mathbf{u}^{(1)}(0)$, $\mathbf{u}^{(2)}(0)$ are given and the operator has been split into two terms,

$$\mathcal{A}(t, \mathbf{u}) = \mathcal{A}^{(1)}(t, \mathbf{u}) + \mathcal{A}^{(2)}(t, \mathbf{u}).$$

In our application, $\mathcal{A}^{(2)}$ corresponds to the Navier-Stokes equation for the unknowns $\mathbf{v}$ and p and $\mathcal{A}^{(1)}$ to the evolution equations for the density ρ. Each equation in (43) is assumed to have a unique solution. This holds in our application.

At each time step $(t_k, t_{k+1}), k = 1, 2, \cdots, t_{k+1} - t_k = \tau$, we solve the equations in the following order:

$$(i) \qquad \frac{d\check{\mathbf{u}}^{(1)}(t)}{dt} = \mathcal{A}^{(1)}(t, \check{\mathbf{u}}^{(1)}(t), \mathbf{u}^{(2)}(t_k)),\ t_k < t < t_{k+1} \tag{44}$$

with initial value $\check{\mathbf{u}}^{(1)}(t_k)$ computed at the previous time step,

$$(ii) \qquad \frac{d\mathbf{u}^{(2)}(t)}{dt} = \mathcal{A}^{(2)}(t, \bar{\mathbf{u}}^{(1)}(t_{k+1/2}), \mathbf{u}^{(2)}(t)),\ t_k < t < t_{k+1} \tag{45}$$

with initial value $\mathbf{u}^{(2)}(t_k)$, where $\bar{\mathbf{u}}^{(1)}(t_{k+1/2}) = \frac{1}{2}(\check{\mathbf{u}}^{(1)}(t_k) + \check{\mathbf{u}}^{(1)}(t_{k+1}))$,

$$(iii) \qquad \frac{d\mathbf{u}^{(1)}(t)}{dt} = \mathcal{A}^{(1)}(t, \mathbf{u}^{(1)}(t), \bar{\mathbf{u}}^{(2)}(t_{k+1/2})),\ t_k < t < t_{k+1} \tag{46}$$

with initial value $\mathbf{u}^{(1)}(t_k)$, where $\bar{\mathbf{u}}^{(2)}(t_{k+1/2}) = \frac{1}{2}(\mathbf{u}^{(2)}(t_k) + \mathbf{u}^{(2)}(t_{k+1}))$.

To find the local truncation errors due to the splitting, we substitute the exact solution

$\mathbf{u}(t)$ into each equation in (47) and, assuming sufficient regularity, we obtain

$$\begin{aligned}
&\frac{d\mathbf{u}^{(1)}(t)}{dt} - \mathcal{A}^{(1)}(t, \mathbf{u}^{(1)}(t), \mathbf{u}^{(2)}(t_k)) \\
&= \mathcal{A}^{(1)}(t, \mathbf{u}^{(1)}(t), \mathbf{u}^{(2)}(t)) - \mathcal{A}^{(1)}(t, \mathbf{u}^{(1)}(t), \mathbf{u}^{(2)}(t_k)) \\
&= \frac{\partial \mathcal{A}^{(1)}}{\partial \mathbf{u}^{(2)}}(\mathbf{u}^{(2)}(t) - \mathbf{u}^{(2)}(t_k)) + O(\|\mathbf{u}^{(2)}(t) - \mathbf{u}^{(2)}(t_k)\|^2) \\
&= (t - t_k)\frac{\partial \mathcal{A}^{(1)}}{\partial \mathbf{u}^{(2)}}\mathcal{A}^{(2)}(t, \mathbf{u}^{(1)}(t), \mathbf{u}^{(2)}(t)) + O(\|\mathbf{u}^{(2)}(t) - \mathbf{u}^{(2)}(t_k)\|^2)
\end{aligned} \tag{47}$$

Similarly, letting now $\bar{\mathbf{u}}^{(i)}(t_{k+1/2}) = \frac{1}{2}(\mathbf{u}^{(i)}(t_k + \mathbf{u}^{(i)}(t_{k+1}), i = 1, 2$, for the second equation, we get

$$\begin{aligned}
&\frac{d\mathbf{u}^{(2)}(t)}{dt} - \mathcal{A}^{(2)}(t, \bar{\mathbf{u}}^{(1)}(t_{k+1/2}), \mathbf{u}^{(2)}(t)) \\
&= \frac{\partial \mathcal{A}^{(2)}}{\partial \mathbf{u}^{(1)}}(\mathbf{u}^{(1)}(t) - \bar{\mathbf{u}}^{(1)}(t_{k+1/2})) + O(\|\mathbf{u}^{(1)}(t) - \bar{\mathbf{u}}^{(1)}(t_{k+1/2})\|^2) \\
&= (t - t_{k+1/2})\frac{\partial \mathcal{A}^{(2)}}{\partial \mathbf{u}^{(1)}}\mathcal{A}^{(1)}(t, \mathbf{u}^{(1)}(t), \mathbf{u}^{(2)}(t)) + O(\|\mathbf{u}^{(1)}(t) - \bar{\mathbf{u}}^{(1)}(t_{k+1/2})\|^2)
\end{aligned} \tag{48}$$

and finally,

$$\begin{aligned}
&\frac{d\mathbf{u}^{(1)}(t)}{dt} - \mathcal{A}^{(1)}(t, \mathbf{u}^{(1)}(t), \bar{\mathbf{u}}^{(2)}(t_{k+1/2})) \\
&= (t - t_{k+1/2})\frac{\partial \mathcal{A}^{(1)}}{\partial \mathbf{u}^{(2)}}\mathcal{A}^{(2)}(t, \mathbf{u}^{(1)}(t), \mathbf{u}^{(2)}(t)) + O(\|\mathbf{u}^{(2)}(t) - \bar{\mathbf{u}}^{(2)}(t_{k+1/2})\|^2.
\end{aligned} \tag{49}$$

To find the corresponding local discretization errors in $\mathbf{u}^{(1)}(t_{k+1})$ and $\mathbf{u}^{(2)}(t_{k+1})$, we integrate (47), (48) and (49). As in the derivation of the stability estimates in Section 2.2, but where we now integrate from t_k to t_{k+1}, instead from 0 to t, we obtain bounds $O((t_{k+1} - t_k)^2) = O(\tau^2)$ for the first equation. Since $\int_{t_k}^{t_{k+1}}(t - t_{k+1/2})dt = 0$, we obtain, however, bounds $O(\tau^3)$ for the second and third equations.

Further, for the convection equation, we obtain the local bound

$$| O(\tau) | \int_{t_k}^{t_{k+1}} e^{-\alpha\beta(t_{k+1}-s)}ds = \frac{1}{\alpha\beta} | O(\tau) | (1 - e^{-\alpha\beta\tau}) = O(\tau^2), \tau \to 0,$$

which, hence, does not depend on α and β. For the second application of this equation, corresponding to $\mathbf{u}^{(1)}(t)$, we obtain then the splitting error $O(\tau^3)$. In this way we obtain the local discretization errors $O(\tau^3)$ for both $\mathbf{u}^{(2)}(t_{k+1})$ and the final computation of $\mathbf{u}^{(1)}(t_{k+1})$.

We then continue in the same way for the next interval, (t_{k+1}, t_{k+2}), etc. The global errors due to the operator splitting become now $O(\tau^2)$. This is the same order as a proper convex combination of the Euler backward and Crank-Nicholson methods. Clearly, we do not need to keep the time-step constant.

In the problem we deal with, the major computational effort is to solve the Navier-Stokes equations. Using the above version of a symmetric operator splitting methods we only need to solve it once at each time step but still obtain a second order splitting error. Similarly, for the combination of Navier-Stokes with Cahn-Hilliard equations, used in the numerical section, we solve the equations in the order CH–NS–CH. Hereby, to lower the error due to linearization in the CH equation we may repeat the linearization step once. For the NS equation we can also use a symmetric operator splitting method. Thereby we split off the advection part from the rest. Since the initial pressure is not known we must keep the saddle point structure intact and start with the diffusion-pressure equation in the operator splitting method.

The operator splitting method can be applied also for several, more coupled equations, such as fluid flow coupled with poroelasticity, modelled by the Darcy flow and elasticity equations for a deformable porous media. Such decouplings can enable the use of legacy codes available for each separate problem, simplifying the implementation of the method.

7 Numerical illustrations

We consider two test problems.

Problem 7.1 (Problem with a known analytical solution) *We solve the variable density Navier-Stokes equations in the square domain* $[-0.5, 0.5]^2$*, having the following analytical solution,*

$$\begin{aligned} \rho(x,y,t) &= 2 + xcos(sin(t)) + ysin(sin(t)), \\ \mathbf{u}(x,y,t) &= \left[-ycos(t), xcos(t)\right]^T, \\ p(x,y,t) &= sin(x)sin(y)sin(t). \end{aligned} \tag{50}$$

These functions satisfy the mass conservation equation. Correspondingly, the right-hand side to the momentum equation has the form

$$\mathbf{f} = \begin{bmatrix} \rho(x,y,t)(ysin(t) - xcos(t)^2) + cos(x)sin(y)sin(t) \\ -\rho(x,y,t)(xsin(t) + ycos(t)^2) + sin(x)cos(y)sin(t) \end{bmatrix}.$$

We note that this problem does not fully describe the properties of the physical processes we aim to study. For instance, the assumed (volume-wise) incompressibility of the fluid implies that the density is not varying much, which is not satisfied by the function, describing ρ*. Despite of that, the problem has been used as a benchmark test in several other related works, such as [17, 39].*

Problem 7.2 (Rayleigh-Taylor instability) *We compute the development of a Rayleigh-Taylor instability (RTI) in the rectangular domain* $[0, 1] \cup [0, 4]$*, and this problem consists of two immiscible liquids. At* $T = 0$ *the heavier is located above the lighter, and for* $T > 0$*, the system is driven by the action of the downward gravity force, i.e., the force term in the momentum equation is directed downward and is equal to* $\rho\,\mathbf{g}$*.*

Both problems are discretized in space using a $Q_2 - Q_2 - Q_1$ finite element discretization for the density-velocity-pressure. Here $Q1$ and $Q2$ stand for 'bilinear' and 'biquadratic'

Table 2: Error in time, BDF2

	Velocity		Density		Pressure	
τ	Inf norm	Rate	Inf norm	Rate	Inf norm	Rate
0.0312	3.28e-4	3.96	4.66e-4	4.16	1.38e-3	3.81
0.0156	8.28e-5	3.96	1.12e-4	4.34	3.63e-4	3.85
0.0078	2.09e-5	3.93	2.58e-5	4.44	9.42e-5	5.12
0.0039	5.32e-6	-	5.80e-6	-	1.84e-5	-
τ	L^2 norm	Rate	L^2 norm	Rate	L^2 norm	Rate
0.0312	3.47e-2	4.08	3.53e-2	4.36	2.67e-2	3.88
0.0156	8.50e-3	4.05	8.90e-3	4.04	6.89e-3	3.81
0.0078	2.10e-3	4.01	2.20e-3	3.91	1.81e-3	3.76
0.0039	5.24e-4	-	5.63e-4	-	4.81e-4	-

basis functions, correspondingly. The FEM pair Q_2-Q_1 is one of the Taylor-Hood elements and is LBB-stable.

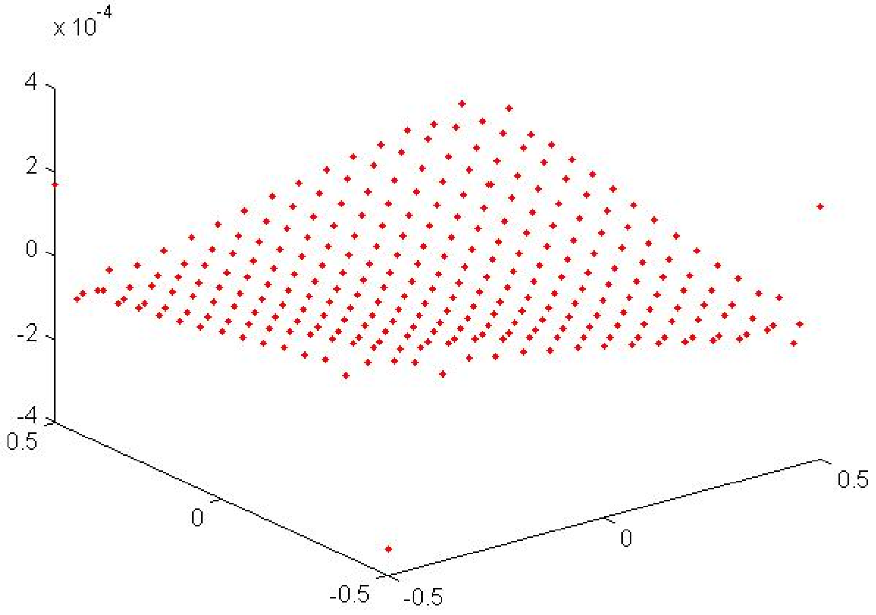
We first present some tests for Problem 7.1. The discrete problem is solved using the regularized splitting schemes of Sections 3 and 4. The mesh size is chosen small enough so that the error from the discretization in space is negligible compared to the time stepping error.

Since when using BDF2 (Algorithm 2) the overall discretization error is $O(\tau^2)+O(h^2)$, the time steps tested are in the range $8h \geq \tau \geq h$, where h is the mesh size. The results are shown in Table 2, where the parameters are fixed as $h = 0.0039$, $\mu = 0.01$, $T = 3.14$, and the error is measured using the maximum norm. As can be seen, the time discretization error is of second-order for all the quantities. To check whether Algorithm 2 can reasonably resolve the pressure, we plot in Figure 1 the difference between the analytical and the computed pressure. Figure 1(a) shows that the pressure is captured quite well and the small difference between the computed and analytical pressure holds everywhere in the domain. This is a contrast to the plot in [19] where an artificial boundary condition is imposed for the pressure unknowns causing a larger error along the boundaries of the computational domain.

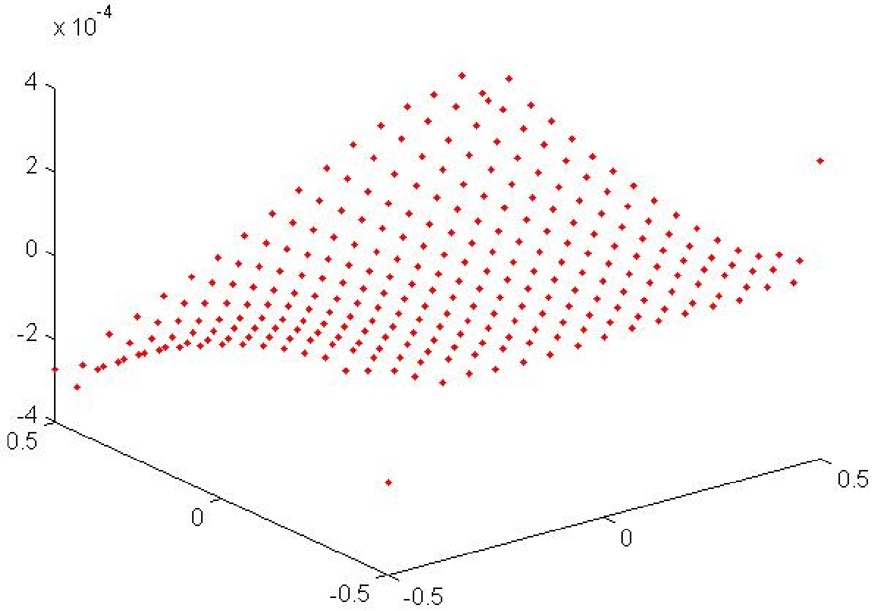
After discretizing in space and time, we rewrite the system in (25) into a block matrix form as follows:

$$\mathcal{A}\begin{bmatrix}\mathbf{v}_h(t+\tau)\\ p_h(t+\tau)\end{bmatrix} = \text{rhs}, \quad \text{where} \quad \mathcal{A} = \begin{bmatrix}A & B^T\\ B & -\tau^2 L_p\end{bmatrix}. \tag{51}$$

Here, $\mathbf{v}_h$, p_h denote the corresponding approximations of $\mathbf{v}$ and p, discretized in space. For simplicity, we assume that the viscosity μ is constant. The system (51) is solved by a generalized conjugate gradient method, such as GMRES ([36]) or GCGMR ([3]). The matrix block A has the form $A = \frac{1}{\tau}M + E$, where M is a mass matrix and E originates from the discrete diffusion and convection terms. The block B arises from the negative



(a) BDF2



(b) OLTM

Figure 1: Difference between exact and computed pressure, $\tau = h = 0.0156$ and $T = 1.57$

divergence operator. The matrix L_p is computed as $L_p = L_p^0 + h^2 I$, where L_p^0 is the discrete Laplacian operator for the pressure unknowns perturbed by the scaled identity matrix. Thus, the matrix L_p is nonsingular.

The preconditioner for the matrix $\mathcal{A}$ in (51), used in this paper, has a block lower-triangular form as follows:

$$\mathcal{P} = \begin{bmatrix} \widetilde{A} & O \\ B & \widetilde{S} \end{bmatrix}, \tag{52}$$

where $\widetilde{S}$ is the approximation of the exact Schur complement of $\mathcal{A}$. As already described, we let

$$\widetilde{S}^{-1} = -\frac{1}{\tau} L_p^{-1} - \mu \widetilde{M}_p^{-1},$$

where $\widetilde{M_p}$ is the diagonal part of the mass matrix for the pressure unknowns. It is trivial to solve linear systems with the diagonal matrix $\widetilde{M}_p$, and for the linear systems with L_p we use an off-the-shelf aggregation-based multigrid solver, for instance, the AGMG method from [33].

The matrix $\widetilde{A}$ denotes the approximation of the pivot matrix A, and the approximation can be defined as an inner iterative solution with proper stopping tolerance. In this paper, we also use the AGMG method as the inner solver.

To check the quality of the preconditioner $\mathcal{P}$ for the Problem 7.1, Table 3 presents the iteration counts when solving the system (51) by GCGMR and the iterations when solving systems with L_p and A by the AGMG method. All three stopping tolerances are relative and chosen to be 10^{-6}. For a fixed ratio τ/h, i,e., looking through the rows, one can see that the AGMG and the GCGMR iterations are essentially independent of the mesh refinement. If we fix the mesh size h, i.e., looking through the columns, one can see that AGMG and the GCGMR iterations are slightly dependent of the time steps τ. Table 3 shows that the preconditioner $\mathcal{P}$ with AGMG as the inner solver for the sub-blocks has an optimal order of computational complexity.

Table 3 is obtained by setting all three stopping tolerances as 10^{-6}. The purpose of choosing such a tight tolerance for AGMG is to illustrate its efficiency. In practice, however, it is not necessary to choose such a small tolerance for the AGMG solver. For the L_p^{-1}, only one V-cycle AGMG is enough to obtain an accurate solution.

To have a fair comparison with respect to the BDF2 scheme, for OLTM in (27)-(30) the parameters are the same and the results of the test runs are shown in Table 4. As is seen, the time discretization error is also second-order for all the quantities. Figure 1(b) illustrates the difference between the exact and the computed pressure in the case when OLTM is used. Based on Figure 1(b) we see that OLTM resolves the pressure as well as the BDF2 scheme.

We turn next to Problem 7.2. RTI describes the phenomena of mixing of two materials, one lighter than the other, seeking to reduce their combined potential energy. This is a potentially unstable hydrodynamic configuration, occurring in linear, nonlinear and turbulent regimes. RTI, namely, the occurrence of interfacial instabilities, has been first considered

Table 3: iterations

	its			its			its		
τ	L_p	A	$\mathcal{A}$	L_p	A	$\mathcal{A}$	L_p	A	$\mathcal{A}$
	$h = 0.031$			$h = 0.015$			$h = 0.007$		
h	8	5	5	8	5	5	8	7	7
$2h$	8	7	7	8	7	7	8	8	9
$4h$	8	9	8	8	9	10	8	8	8
$8h$	8	7	11	8	10	10	8	10	10

Table 4: Error in time for OLTM.

	Velocity		Density		Pressure	
τ	Inf norm	Rate	Inf norm	Rate	Inf norm	Rate
0.0312	3.79e-3	3.91	8.41e-5	3.95	5.69e-3	3.79
0.0156	9.70e-4	3.85	2.13e-5	3.88	1.50e-3	3.80
0.0078	2.52e-4	4.01	5.49e-6	3.92	3.99e-4	3.77
0.0039	6.28e-5	-	1.40e-6	-	1.06e-4	-

τ	L^2 norm	Rate	L^2 norm	Rate	L^2 norm	Rate
0.0312	2.12e-1	3.91	4.89e-3	3.79	4.67e-2	3.74
0.0156	5.42e-2	3.88	1.29e-3	3.89	1.25e-2	3.69
0.0078	1.40e-2	4.37	3.32e-4	4.01	3.40e-3	3.80
0.0039	3.20e-3	-	8.29e-5	-	8.95e-4	-

in 1883 by Lord Rayleigh in the idealized setting of two incompressible immiscible fluids in a constant gravitational field,

Due to the fact that over longer time the processes become turbulent and need to be resolved down to the Kolmogorov scale, some studies take the approach to consider the materials compressible (cf. [16]). We confine ourselves to simulations of relatively short time intervals, before reaching the turbulent mixing stage. The references we have used are [44, 16, 17, 27, 23]. In some of the works, the simulations are performed within the sharp interface framework, referred also as 'direct numerical simulations' (DNS) of the NS equations, as in [16, 17]. The advantage of DNS, compared to the other option introduced below, is its simplicity since one does not have to couple other mathematical models with the Navier-Stokes equations. The main drawback is that DNS method highly depends on the computer resources. Because the Navier-Stokes equations do not straightforwardly describe the interface and the surface tension force is not considered at all, the mesh must be sufficiently fine to guarantee the resolution, especially within the interface.

In some other works, such as in [44], Euler equations are solved. A paper rich with experimental observations and numerical simulations, based on NS but assuming compressibility, is [23].

The framework we follow here is to couple NS to CH. The CH equations are one of the modelling tools within the diffuse interface approach and their solution replaces the need to solve the density equation. This approach is followed by [27, 40, 21, 13] and others.

The coupled NS-CH system in dimensionless form reads as follows

$$\begin{aligned}
\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{v} - \frac{1}{Re}\Delta\mathbf{u} + \nabla p &= \frac{\rho}{Fr}\mathbf{g}\\
\nabla\cdot\mathbf{v} &= \mathbf{u}\cdot\nabla\rho\\
\eta - \Psi'(C) + \epsilon^2\Delta C &= 0\\
-\frac{1}{Pe}\Delta\eta + \frac{\partial C}{\partial t} + (\mathbf{u}\cdot\nabla)C &= 0\\
\frac{\partial C}{\partial n} = 0, \quad \frac{\partial \eta}{\partial n} &= 0, \ \mathbf{x}\in\partial\Omega\\
C(\mathbf{x},0) &= C_0(\mathbf{x}).
\end{aligned} \tag{53}$$

together with some appropriate boundary and initial conditions for the NS equations.

In Problem 7.2, we assume that the two immiscible and incompressible phases share the same viscosity value but di sities. In the above equations, the term C is referred to as the concentration and the term η denotes the chemical potential. The function $C(\mathbf{x},t)$ models the interface, which attains a different constant value in each phase flow and rapidly but smoothly changes within the interface between the phase flows. In this paper we let C vary between -1 and 1 and, accordingly, choose the chemical potential $\Psi(C)$ as $\Psi(C) = \frac{1}{4}(C^2-1)^2$. The dimensionless physical parameters are the Reynolds number Re, Froude number Fr, Peclet number Pe and the Cahn number ϵ. The derivation of dimensionless form (53) and the expressions of these parameters are clearly given in [27].

System (53) is solved again using an operator splitting technique. One solves NS to determine the velocity, then solves CH to determine the position of the interfaces and, respectively, ρ and μ, and repeats. In [21], convergence of the NS-CH splitting is derived. We point out also that, in general, the coupling between NS and CH is not only via velocity but also via an additional term in the right hand side of NS, that depends on the concentration variable and describes the so-called surface tension forces. As explained in [44], and implemented in [17], that term can be ignored when RTI is simulated.

As can be seen, in contrast to the DNS method (19)–(21), in the coupled system (53) there is no need to solve the density equation anymore. The reason is that when the interface is known, the density can be straightforwardly obtained. In [40, 27], after computing C at the next time step, the density is recovered as an arithmetic average of the densities of the different phases (two in this case),

$$\rho(C) = \rho_1(1 - C)/2 + \rho_2(1 + C)/2, \tag{54}$$

where ρ_1, ρ_2 denote the constant density values within each phase. A more correct relation to recover the density is as a weighted harmonic average

$$\rho(C) = 2 \Bigg/ \left(\frac{1 - C}{\rho_1} + \frac{1 + C}{\rho_2} \right). \tag{55}$$

The expression (55) follows naturally since $(1 - C)/2$ and $(1 + C)/2$ are the mass fractions of the fluids and a unit volume $1/\rho$ consists of the corresponding volumes of the two fluids.

The numerical scheme to compute (53) is described as follows:

Algorithm 4 (Coupled NS-CH) Given initial $(C_0, \eta_0, \mathbf{u}_0)$, for $n \geq 1$, proceed:

A4-1: Compute (C^{n+1}, η^{n+1}) by solving the Cahn-Hilliard equations

$$\begin{aligned} \eta^{n+1} - \Psi'(C^{n+1}) + \epsilon^2 \Delta C^{n+1} &= 0, \\ -\frac{1}{Pe}\Delta \eta^{n+1} + \frac{\partial C^{n+1}}{\partial t} + (\mathbf{u}^n \cdot \nabla) C^{n+1} &= 0, \end{aligned} \tag{56}$$

A4-2: Recover ρ^{n+1} as (54) or (55).

A4-3: Compute $(\mathbf{v}^{n+1}, p^{n+1})$ by solving

$$\begin{aligned} \frac{3\mathbf{v}^{n+1} - 4\mathbf{v}^n + \mathbf{v}^{n-1}}{2\tau} + (\mathbf{u}^\star \cdot \nabla)\mathbf{v}^{n+1} - \frac{1}{Re}\Delta \frac{\mathbf{v}^{n+1}}{\rho^{n+1}} + \nabla p^{n+1} &= \frac{\rho^{n+1}}{Fr}\mathbf{g} \\ \nabla \cdot \mathbf{v}^{n+1} - \tau^2 \Delta p^{n+1} &= \mathbf{u}^\star \cdot \nabla \rho^{n+1} \end{aligned} \tag{57}$$

where $\mathbf{u}^\star = 2\mathbf{u}^n - \mathbf{u}^{n-1}$.

A4-4: Recover $\mathbf{u}^{n+1} = \mathbf{v}^{n+1}/\rho^{n+1}$.

Here τ is the time step. For $n = 1$, $(\mathbf{v}^1, \mathbf{u}^1, p^1)$ can be computed by using one step of the first-order scheme as in Algorithm 1, Step A1-2.

How to handle the nonlinear term $\Psi'(C^{n+1})$ and how to efficiently solve the linear systems arising from the Cahn-Hilliard equations (56) falls out of the scope of the present paper. We follow the details in [12]. For the solution of the Navier-Stokes equations (57) we use BDF2 and the already discussed preconditioning techniques.

The setting of the problem parameters for Problem 7.2 are fixed as follows. The density difference is represented by the Atwood number $At = (\rho_2 - \rho_1)/(\rho_2 + \rho_1)$. Here we choose the density ratio to be $\rho_2/\rho_1 = 3$, i.e., we consider a low Atwood number $At = 0.5$. At $T = 0$, the initial location of the two immiscible and incompressible flows are imposed as

$$C(\mathbf{x}, 0) = tanh(\frac{y - 2 - 0.1cos(2\pi x)}{\sqrt{2}\epsilon}),$$

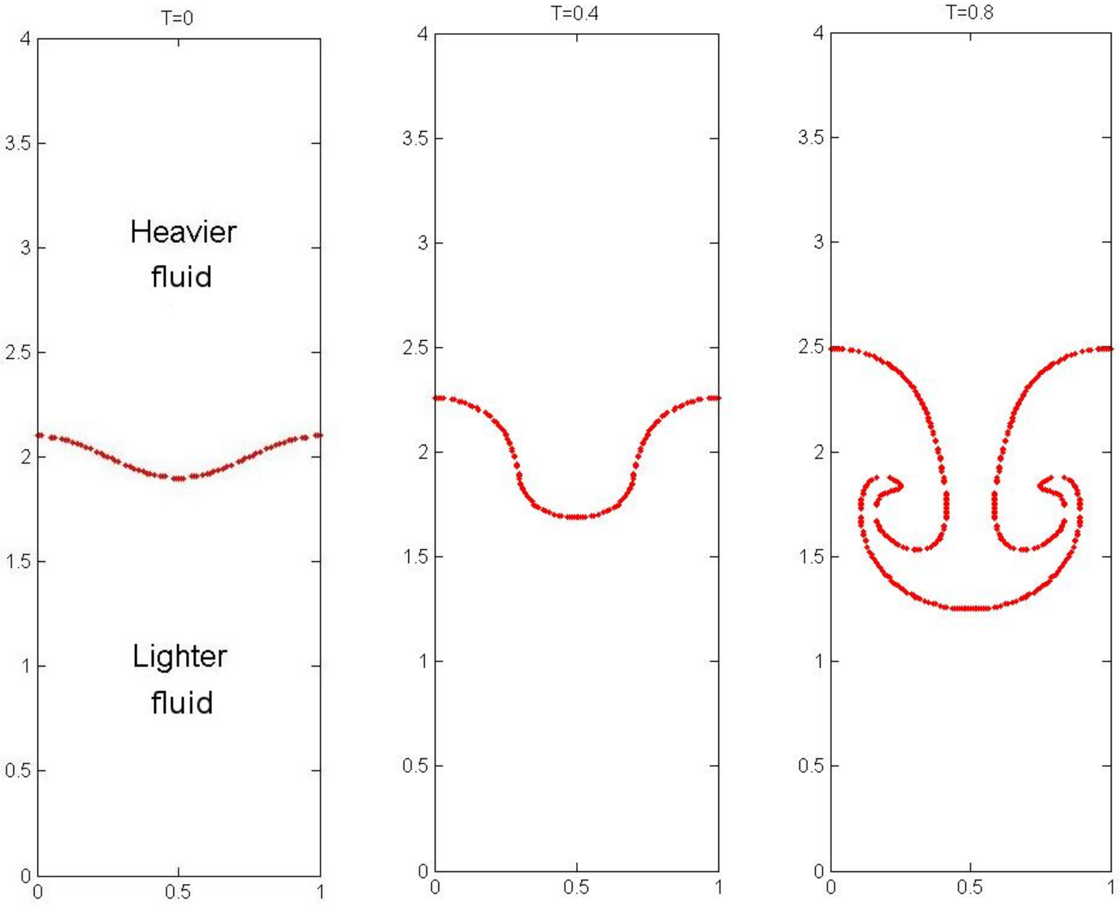
where $C = 1$ denotes the flow with density ρ_2, $C = -1$ corresponds to the flow with density ρ_1, and $-1 < C < 1$ describe the location of the interface. The computational domain is $[0, 1] \cup [0, 4]$ and we simulate on a space mesh 128×512, i.e., the mesh size is $h = 1/128$. The Cahn number ϵ, that determines the thickness of the interface, is set to be $\epsilon = 2h$. The reason is that we want to have a fixed number of grid points across the interface. The Peclet number is chosen to be $Pe = 1/\epsilon$ and the effect of the Peclet number is numerically illustrated in [27]. The Reynolds number is set to be $Re = 3000$ and the time step is chosen as $\tau = 0.25 \times h$.

The evolution of the interface is plotted in Figure 2. As can be seen, at $T = 0.8$ the rolling-up of the falling flow can be clearly seen and this plot resembles well that in [27]. Even though the time evolution is relatively short, it illustrates well the numerical behaviour of the solution methods, that constitute the focus of the present work.

8 Concluding remarks

We have studied various aspects of the numerical solution of the variable density Navier-Stokes equations - discretization, operator splitting and linearization, and the interplay between the related errors. We have also considered preconditioning techniques, suitable for the arising linear systems, in order to enable fast and robust numerical simulations of the underlying flow phenomena.

We have investigated the impact of using the momentum instead of the velocity as a primal unknown variable, based on the argument that the momentum is better behaved than the velocity and is, thus easier to compute numerically. We have also considered the efficiency of an operator splitting, where no artificial boundary conditions have to be imposed for the pressure, enabling in this way the use of well-known high quality preconditioners for saddle point matrices. The divergence free vector field, is computed without the need to use projections onto the divergence-free vector space. No lower-order errors due to splitting and linearization arise.



(a) T=0

(b) T=0.4

(c) T=0.8

Figure 2: The evolution of the interface

It is argued to use the regularized matrix, described in Section 4, as a block matrix preconditioner for the fully discretized version of the coupled problem (8)-(10). Since this is a nonlinear problem, the use of the regularized block matrix based on (32), can be considered as a nonlinear iteration method. It is similar to the case of Oseen equations as preconditioner to the Navier-Stokes equations and also similar to the method used to solve two-phase flow problem of Cahn-Hilliard type in [7]. We can use several such iteration steps at each time step. Each iteration includes the case of the regularized matrix in Section 4 as outer iteration method, which itself includes inner iterations. This means that we have three levels of embedded iterations:

(i) nonlinear iterations,

(ii) outer block matrix iteration,

(iii) inner iterations for the top-left pivot block.

Since the iterations multiply up, clearly for such a method to be efficient one must use very few steps, perhaps only one or two in (i) and (ii). This is confirmed by numerical experiments.

A complete testing and comparison of the performance of the proposed solution scheme with other projection-based operator splitting schemes remains to be done.

The methods used in this paper can be applied also for more coupled equations, such as Navier-Stokes equations for a free fluid coupled with poro-elasticity equations, modelled by Darcy flow equations and equations for a porous elastic medium.

Acknowledgements

The work of the third author is (partly) supported by the Swedish Research Council (VR) via the grant *Finite element preconditioners for algebraic problems as arising in modelling of multiphase microstructures*, 2009-2011. The support is hereby gratefully acknowledged.

References

[1] E. D'Agnillo, *Importance of discrete mass conservation in incompressible flows.* M.Sc. Thesis, Department of Mathematical Sciences, Clemson University, April 12, 2012.

[2] O. Axelsson, Error estimates over infinite intervals of some discretizations of evolution equations, *BIT*, 24 (1984), 413–424.

[3] O. Axelsson. *Iterative Solution Methods.* Oxford University Press, 1994.

[4] O. Axelsson, V.A. Barker, M. Neytcheva, B. Polman, Solving the Stokes Problem on a Massively Parallel Computer, *Math. Modelling and Analysis*, 6 (2001), 7–27.

[5] O. Axelsson, E. Glushkov, N. Glushkova. The local Green's function method in singularly perturbed convection-diffusion problems. *Math. Comp.* 78 (2009), no. 265, 153–170.

[6] O. Axelsson, S.V. Gololobov. Stability and error estimates for the θ-method for strongly monotone and infinitely stiff evolution equations. *Numer. Math.* 89 (2001), 31–48.

[7] O. Axelsson, P. Boyanova, M. Kronbichler, M. Neytcheva, X. Wu, Numerical and computational efficiency of solvers for two-phase problems. *Computers & Mathematics with Appl.*, 2012. Published online at `http://dx.doi.org/10.1016/j.camwa.2012.05.020`

[8] O. Axelsson, M. Neytcheva, A general approach to analyse preconditioners for two-by-two block matrices *Numer. Lin. Alg. Appl.* Article first published online: 14 DEC 2011, DOI: 10.1002/nla.830

[9] O. Axelsson, P.S. Vassilevski. A black box generalized conjugate gradient solver with inner iterations and variable-step preconditioning. *SIAM J. Matrix Anal. Appl.* 12 (1991), no. 4, 625–644.

[10] J.B. Bell, D.L. Marcus, A second-order projection method for variable-density flows, *Journal of Computational Physics* 101 (1992), 334–348.

[11] M. Benzi, G. Golub and J. Liesen. Numerical solution of saddle point problmes. *Acta Numer.*, 14:1-137, 2005.

[12] P. Boynanova. On numerical solution methods for block-structure discrete systems. PhD thesis, Department of Information Technology, Uppsala University, Sweden, 2012. Avaliavle online, ISSN 1651-6214; 942.

[13] P. Boyanova, M. Do-Quang, M. Neytcheva. Efficient preconditioners for large scale binary Cahn-Hilliard models, *Computational Methods in Applied Mathematics*, 12 (2012), ISSN (Online) 1609-9389, DOI: 10.2478/cmam-2012-0001.

[14] A.J. Chorin. Numerical solution of the Navier-Stokes equations, *Mathematics of Computation*, 22 (1968), 745–762.

[15] H.-J.G. Diersch, O. Kolditz. Variable-density flow and transport in porous media: approaches and challenges, *Advances in Water Resources*, 25 (2002), 899–944.

[16] A. Gauthier, B. Le Creurer, Compressibility effects in Rayleigh-Taylor instability-induced flows. *Phi. Trans. R. Soc.*, A 368 (2010), 1681–1704.

[17] J.-L. Guermond, A. Salgado, A splitting method for incompressible flows with density based on a pressure Poisson equation. *J. Comput. Phys.* 228 (2009), 2834–2846.

[18] J.-L. Guermond, L. Quartapelle, A projection FEM for variable density incompressible flows. *J. Comput. Phys.* 165 (2000), no. 1, 167–188.

[19] J.-L. Guermond, P. Minev, J. Shen, An overview of projection methods for incompressible flows. *Comput. Methods Appl. Mech. Eng.*, 195 (2006), 6011–6045.

[20] J.-L. Guermond, R. Pasquetti, B. Popov, Entropy viscosity method for nonlinear conservation laws. *J. Comput. Phys.*, 230 (2011), 4248–4267.

[21] X. Feng, Fully discrete finite element approximations of the Navier-Stokes–Cahn-Hilliard diffuse interface model for two-phase fluid flows. *SIAM J. Numer. Anal.*, 44 (2006), 1049–1072.

[22] C. Johnson, U. Nävert, J. Pitkäranta, Finite element methods for linear hyperbolic problems. *Comput. Methods Appl. Mech. Eng.* 45 (1984), no. 1-3, 285–312.

[23] D.D. Joseph, Fluid dynamics of two miscible liquids with diffusion and gradient stresses. *Eur. J. Mech. B*, 9 (1990), 565–596.

[24] X. He, M. Neytcheva, S. Serra Capizzano. On an augmented Lagrangian-based preconditioning of Oseen type problems *BIT Numerical Analysis*, 51 (2011), 865–888.

[25] X. He, M. Neytcheva, Preconditioning the incompressible Navier-Stokes equations with variable viscosity. *Journal of Computational Mathematics*, 2012. In press.

[26] W. Hundsdorfer, J. Verwer, Numerical solution of time-dependent advection-diffusion-reaction equations. Springer Series in Computational Mathematics, 33. Springer-Verlag, Berlin, 2003.

[27] H.G. Lee, K. Kim, J. Kim, On the long term simulation of the Rayleigh-Taylor instability. *Int. J. Numer. Meth. Eng.*, 00 (2010), 1–25.

[28] A. Linke, Collision in a cross-shaped domain – a steady 2D Navier-Stokes example, demonstrating the importance of mass conservation in CFD. *Comp. Meth. Appl. Math. Eng.*, 198 (2009), 3278–3286.

[29] P. Lions, *Mathematical Topics in Fluid Mechanics, Vol 1: Incompressible models.* Clarendon Press, Oxford, 1996.

[30] Ch. Liu, N.J. Walkington. Convergence of numerical approximations of the incompressible Navier-Stokes equations with variable density and viscosity. *SIAM J. Numer. Anal.* 45 (2007), no. 3, 1287–1304.

[31] G.I. Marchuk, Some application of splitting–up methods to the solution of mathematical physics problems, *Appl. Mat.* 13(2) (1968), 103–132.

[32] K.-A. Mardal, R. Winther Uniform preconditioners for the time dependent Stokes problem. *Numer. Math.*, 98 (2004), 305–327.

[33] Y. Notay, The software package AGMG, `http://homepages.ulb.ac.be/~ynotay/`

[34] M. Olshanskii, A. Reusken, Grad-div stabilization for Stokes equations. *Math. Comp.*, 73 (2004), 713–742.

[35] R. Pierre, Simple C^0 Approximations for the Computation of Incompressible Flows, *Comp. Meth. Appl. Mech. Eng.*, 68 (1988), 205–227.

[36] Y. Saad and M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Statist. Comput.*, 7 (1986), 856–869.

[37] R. Rannacher, On Chorin's projection method for the incompressible Navier-Stokes equations. The Navier-Stokes equations II – theory and numerical methods (Oberwolfach, 1991), 167-183, Lecture Notes in Math., 1530, Springer, Berlin, 1992.

[38] Y. Saad, A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.* 14 (1993), no. 2, 461–469.

[39] A. Salgado, *Approximation Techniques for Incompressible Flows with Heterogeneous Properties.* PhD Dissertation, Texas A& M University, 2010.

[40] J. Shen, X. Yang, A phase-field model and its numerical approximation for two-phase incompressible flows with different densities and viscosities. *SIAM J. Sci. Comp.*, 32 (2010), 1159–1179.

[41] C. Simmons, Variable density groundwater flow: From current challenges to future possibilities, *Hydrogeology Journal*, 13 (2005), 116–119.

[42] G. Strang, On the construction and comparison of difference schemes, *SIAM J. Numer. Anal.* 5(3)(1968), 506–517.

[43] R. Temam, Une méthode d'approximation de la solution des équations de Navier-Stokes. *Bull. Soc. Math. France*, 96 1968, 115–152.

[44] G. Tryggvason, Numerical simulations of the Rayleigh-Taylor instability. Journal of Computational Physics, 75 (1988), 253–282.

[45] S. Turek, *Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach*, Springer, ISBN 3-540-65433-X, 1999

[46] P.S. Vassilevski. *Multilevel Block Factorization Preconditioners*, Springer-Verlag New York Inc., 2008.

[47] W. Villanueva, G. Amberg, Some generic capillary-driven flows, *International Journal of Multiphase Flow*, 32 (2006), 1072–1086.

[48] G. Wanner, A short proof on nonlinear A-stability, Scientific Notes, *BIT* 16 (1976), 226–227.

[49] S. Zahedi, *Numerical Methods for Fluid Interface Problems*. Doctoral Thesis in Applied and Computational Mathematics, TRITA-CSC-A 2011:07, ISSN 1653-5723 (2011).