Chapter 21

Operator Splitting Algorithms for Free Surface Flows: Application to Extrusion Processes

Andrea Bonito, Alexandre Caboussat, and Marco Picasso

Abstract We investigate the benefits of operator splitting methods in the context of computational fluid dynamics. In particular, we exploit their capacity at handling free surface flows and a large variety of physical phenomena in a flexible way. A mathematical and computational framework is presented for the numerical simulation of free surface flows, where the operator splitting strategy allows to separate inertial effects from the other effects. The method of characteristics on a fine structured grid is put forward to accurately approximate the inertial effects while continuous piecewise polynomial finite element associated with a coarser subdivision made of simplices is advocated for the other effects. In addition, the splitting strategy also allows modularity, and in a straightforward manner rheological model change for the fluid. We will emphasize this flexibility by treating Newtonian flows, visco-elastic flows, multi-phase, and multi-density immiscible incompressible Newtonian flows. The numerical framework is thoroughly presented; the test case of the filling of a cylindrical tube with potential die swell in an extrusion process is taken as the main illustration of the advantages of operator splitting.

A. Bonito (✉)
Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA
e-mail: bonito@math.tamu.edu

A. Caboussat
Haute Ecole de Gestion de Genève, University of Applied Sciences Western Switzerland
(HES-SO), Rue de la Tambourine 17, 1227 Carouge, Switzerland
e-mail: alexandre.caboussat@hesge.ch

M. Picasso
MATHICSE, Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland
e-mail: marco.picasso@epfl.ch

1 Introduction

Complex free surface phenomena involving multi-phase Newtonian and/or Non-Newtonian flows are nowadays a topic of active research in many fields of physics, engineering, and bioengineering. Numerous mathematical models and associated numerical approximations for complex liquid-gas free surfaces problems are also available.

The purpose of this chapter is to present a comprehensive review of a computational methodology developed in the group of Jacques Rappaz at *Ecole polytechnique fédérale de Lausanne (EPFL)*, called *cfsFlow* and commercialized by a spin-off company of EPFL named Ycoor Systems S.A. [40]. Originally proposed for two-dimensional cases by Maronnier, Picasso, and Rappaz [25], it evolved to handle three-dimensional flows [26], account for surrounding compressible gas [11, 12] and surface tension [8], allow complex rheology [6], include space adaptive interface tracking [9], and recently integrate multi-phase fluids [19]. Besides the typical fluid flows applications, it is worth noting that these methods have been also applied successfully to predict the evolution of glaciers [20, 21, 33].

Many algorithms are available to approximate free boundary problems, see for instance [2, 29, 31, 37, 38]. The novelty in *cfsFlow* is to use a time splitting approach [15] and a two-grids method to decouple advection and diffusion regimes. This allows the use of well-suited numerical techniques for each of the two regimes separately. In particular, the advection phenomenon describing the evolution of each liquid phases is approximated on structured grids by the forward method of characteristics [34] on the volume-of-fluid representation of each phase. On the other hand, finite element approximations on simplices determined as liquid are implemented to handle diffusion-like phenomena.

We start by discussing in Section 2 the basic model for Newtonian fluids with free surface. The type of operator splitting strategies considered and their applications to free boundary problems are presented in Section 3, the associated numerical algorithms being presented in Section 4. Fluids verifying more complex rheology are discussed in Section 5, where the upper convected Maxwell constitutive relation for the extra stress tensor is chosen as our model problem. Multi-phase fluids are considered in Section 6 and perspectives on emulsion processes are put forward in Section 7.

The filling of a cylindrical tube with potential die swell in an extrusion process is taken as the main illustration of the advantages of the presented numerical algorithm and is used throughout this chapter to evaluate the effect of each component in the final model. We note in passing that the numerical simulations of extrusion is of great importance for instance in industrial processes involving pasta dough [22] or textile products [1].

Acknowledgements

2 Mathematical Modeling of Newtonian Fluids with Free Surfaces

We present in this section the mathematical model used to describe the evolution of an incompressible Newtonian fluid with a free surface, neglecting the effect of the ambient fluid. A simple model for the treatment of the ambient fluid has been proposed in [12] and the addition of surface tension effects has been described in [8, 9].

The computational domain is denoted by $\Lambda \subset \mathbb{R}^d$, $d = 2, 3$, and $T > 0$ stands for the final time. We describe in Section 2.1 the Navier-Stokes equations for fluids subject to free boundaries and detail in Section 2.2 the Eulerian approach used to track the liquid domain evolution.

2.1 Navier-Stokes System

We denote by $\Omega(t) \subset \Lambda$ the domain occupied by the liquid at time $t \in [0, T]$ and by

$$Q := \{(\mathbf{x}, t) \in \Lambda \times (0, T] \mid \mathbf{x} \in \Omega(t)\},$$

the space-time liquid domain. The fluid is assumed to be incompressible and Newtonian so that its velocity $\mathbf{u} : Q \to \mathbb{R}^d$ and pressure $p : Q \to \mathbb{R}$ are the solutions to the Navier-Stokes equations:

$$\begin{cases} \rho \left(\dfrac{\partial}{\partial t} \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} \right) - 2\nabla \cdot (\mu \mathbf{D}(\mathbf{u})) + \nabla p = \mathbf{f} & \text{in} \quad Q, \\ \nabla \cdot \mathbf{u} = 0 & \text{in} \quad Q, \end{cases} \tag{21.1}$$

where $\mathbf{f} : Q \to \mathbb{R}^d$ is a given external force (typically $\mathbf{f} := \rho \mathbf{g}$, where $\mathbf{g}$ is the gravitational acceleration), $\mathbf{D}(\mathbf{u}) := \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^t)$ is the symmetric part of the velocity gradient and $\rho > 0$, $\mu > 0$ are respectively the fluid density and viscosity. Notice that the Navier-Stokes equations are only defined in the liquid domain Q, the effect of the outside fluid being neglected. Hence, the velocity and pressure outside Q are not defined.

We now discuss the boundary/interface conditions associated with the above system and refer to Figure 21.1 for an illustration in the die swell context. We separate the boundary of the computational domain in two disjoint open sets $\Gamma_{\mathscr{D}}$ and $\Gamma_{\mathscr{N}}$ such that $\partial \Lambda = \overline{\Gamma_{\mathscr{D}}} \cup \overline{\Gamma_{\mathscr{N}}}$. The velocity is prescribed on $\Gamma_{\mathscr{D}}$ (Dirichlet boundary condition), i.e. for a given $\mathbf{g}_{\mathscr{D}} : \Gamma_{\mathscr{D}} \times [0,T] \to \mathbb{R}^d$, we have

$$\mathbf{u} = \mathbf{g}_{\mathscr{D}} \quad \text{on} \quad \partial Q_{\mathscr{D}} := \{(\mathbf{x},t) \in \Gamma_{\mathscr{D}} \times [0,T] \mid \mathbf{x} \in \partial \Omega(t)\}. \tag{21.2}$$

On the other hand, a force is applied on $\Gamma_{\mathscr{N}}$ (Neumann boundary condition), i.e. for a given $\mathbf{g}_{\mathscr{N}} : \Gamma_{\mathscr{N}} \times [0,T] \to \mathbb{R}^d$, we have

$$(2\mu \mathbf{D}(\mathbf{u}) - p\mathbf{I})\mathbf{n} = \mathbf{g}_{\mathscr{N}} \quad \text{on} \quad \partial Q_{\mathscr{N}} := \{(\mathbf{x},t) \in \Gamma_{\mathscr{N}} \times [0,T] \mid \mathbf{x} \in \partial \Omega(t)\}, \tag{21.3}$$

where $\mathbf{n}(.,t)$ is the outward pointing unit vector normal to $\partial \Omega(t)$ and $\mathbf{I}$ is the identity tensor. More general boundary conditions such as slip boundary conditions could be considered in a similar way but are not included here to keep the presentation as simple as possible.

Regarding the free interface condition, we assume that no force is exerted to the liquid, that is

$$(2\mu \mathbf{D}(\mathbf{u}) - p\mathbf{I})\mathbf{n} = \mathbf{0} \quad \text{on} \quad \mathscr{F} := \{(\mathbf{x},t) \in \Lambda \times (0,T] \mid \mathbf{x} \in \partial \Omega(t) \setminus \partial \Lambda\}, \tag{21.4}$$

and, since the interface evolves with the fluid velocity, that the interface velocity $\mathbf{u}_{\mathscr{F}}$ satisfies

$$\mathbf{u}_{\mathscr{F}} = \mathbf{u} \quad \text{on} \quad \mathscr{F}. \tag{21.5}$$

Finally, an initial condition $\mathbf{u}_0 : \Omega(0) \to \mathbb{R}^d$ is provided for the velocity

$$\mathbf{u}(.,0) = \mathbf{u}_0 \quad \text{on} \quad \Omega(0). \tag{21.6}$$

2.2 Implicit Representation of the Liquid Domain

The liquid domain $\Omega(t)$ is mathematically represented during the evolution via its characteristic function $\phi : \Lambda \times [0,T] \to \{0,1\}$, implying that

$$\Omega(t) = \{\mathbf{x} \in \Lambda \mid \phi(\mathbf{x},t) = 1\}, \qquad t \in [0,T]. \tag{21.7}$$

In view of the interface velocity condition (21.5), we interpret the evolution of $\Omega(t)$ as the transport of its characteristic function with the fluid velocity:
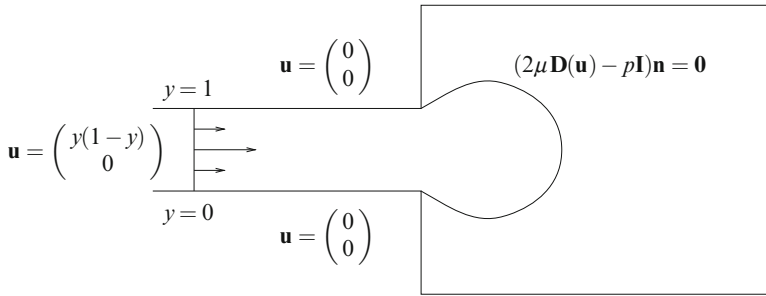
Fig. 21.1 Boundary conditions in the context of die swell. The liquid enters the cavity with an initial velocity **u**, the horizontal component is a parabolic profile and the vertical component vanishes. The velocity is imposed to vanish on the rest of the boundary. Another setup would be to enforce slip boundary conditions on the lateral walls of the extruder, implying a constant, instead of parabolic, profile of velocities in the tube.

$$\frac{\partial}{\partial t}\phi + \mathbf{u} \cdot \nabla\phi = 0 \quad \text{in} \quad Q, \qquad \phi = 0 \quad \text{in} \quad \Lambda \setminus Q, \tag{21.8}$$

where **u** is the fluid velocity only defined on the space-time fluid domain Q as noted in Section 2.1. An illustration is provided in Figure 21.2.

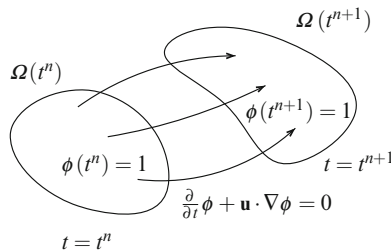**Fig. 21.2** Deformation of the liquid domain $\Omega(t)$ for $t \in [t^n, t^{n+1}]$ deduced from the transport of the characteristic function ϕ with the liquid velocity **u** according to (21.8).

Remark 1 (Ambient Fluid and Computational Cost). As the effect of the outside fluid is neglected, the Navier-Stokes relations (21.1) for the velocity-pressure pair are only considered in the space-time liquid domain Q. As a consequence, the velocity is only defined on Q and so is the transport equation for the characteristic function in (21.8). A possible equivalent alternative would consist in finding an extension of the velocity field to $\Lambda \times (0,T]$, thereby extending the transport relation to the entire space-time computational domain $\Lambda \times (0,T]$. However, the numerical scheme described in Section 4 takes full advantage of the representation (21.8) in order to reduce the overall computational cost.

We supplement the transport equation in (21.8) by the value of the characteristic function ϕ at the inflow boundary

$$\partial Q_{\text{inflow}} := \{ (\mathbf{x},t) \in \partial \Lambda \times (0,T] \mid \mathbf{x} \in \partial \Omega(t) \quad \text{and} \quad \mathbf{u} \cdot \mathbf{n} < 0 \}, \qquad (21.9)$$

namely,

$$\phi = 1 \quad \text{on} \quad \partial Q_{\text{inflow}}. \qquad (21.10)$$

The initial value of the characteristic equation is chosen to match the initial given domain $\Omega(0) := \Omega$,

$$\phi(.,0) = 1 \quad \text{on} \quad \Omega(0) \qquad \text{and} \qquad \phi(.,0) = 0 \qquad \text{elsewhere.} \qquad (21.11)$$

3 Operator Splitting Algorithm

We take advantage of an operator splitting scheme to separate the numerical issues inherent to the approximation of the diffusion and advection operators in the approximation of the system of equations (21.1) and (21.8). In this context, it allows to treat separately Stokes systems on given non-moving liquid domains and transport equations for the velocity and the liquid characteristic function.

Several operator splitting algorithms are available in the literature, starting from the early works of Peaceman and Rachford [32], Douglas and Rachford [14], Marchuk [23, 24], and Yanenko [39]. We refer to Glowinski [15] for a survey of these methods. In Section 3.1, we review a particular version of the so-called *Lie scheme* and we detail in Section 3.2 its application to free boundary problems.

3.1 The Lie Scheme

We advocate in this work a particular version of the *Lie scheme* and follow the description provided in [15, 16] (see also Chapters 1 and 2 in this book). Assume that we are interested in the solution of the Cauchy problem

$$\begin{cases} \dfrac{d}{dt}\mathbf{v} + A(\mathbf{v},t) = 0, \, t \in (0,T], \\ \mathbf{v}(0) = \mathbf{v}_0, \end{cases}$$

where the operator A can be decomposed as the sum of q operators

$$A = \sum_{i=1}^{q} A_i. \qquad (21.12)$$

The scheme starts with a subdivision $0 =: t^0 < t^1 < \ldots < t^N := T$ of the time interval $[0, T]$. Over each sub-interval $I^{n+1} := (t^n, t^{n+1}]$ the approximation of $v(t^{n+1})$ (an approximation of $v(t^n)$ being given) is obtained in q steps (corresponding to q alternating "directions"):

> set $v^0 = v_0$ and $w^0(t) \equiv v^0$ for $t \in [0, t^1]$;
> for $n = 0, \ldots, N-1$
>   for $i = 1, \ldots, q$
>   find $w^{n+i/q}(t)$ as the solution of
> $$\frac{d}{dt} v + A_i(v, t) = 0 \text{ on } (t^n, t^{n+1}]$$
>   and satisfying the initial condition $v(t^n) = w^{n+(i-1)/q}(t^{n+1})$;
>   end for
>   set $v^{n+1} := w(t^{n+1})$
> end for

It turns out that if the operators A_i are *linear*, *time independent*, and they *commute*, then $v^n = v(t^n)$ for $n = 0, \ldots, N$. However, in generic situations, the above scheme is, at most, first order accurate [15]. Nevertheless, this motivates the introduction of first order discretizations in time and space for each sub-step of the algorithm, as described in Sections 4.1 and 4.2.

3.2 Application to Free Surface Flows

In the context of fluid flows with free boundaries, we set up a splitting of type (21.12) using two alternating "directions" ($q = 2$). We call these two steps the *prediction* and *correction* steps which are now described on each time subinterval $I^{n+1} := (t^n, t^{n+1}]$. They consist in separating the hyperbolic regime from the parabolic regime in order to apply numerical methods well suited to each situation; see Section 4.

We assume that an approximation of the liquid characteristic function ϕ^n is given, and therefore so is an approximation of the liquid domain Ω^n via the relation

$$\Omega^n := \{\mathbf{x} \in \Lambda \mid \phi^n(\mathbf{x}) = 1\}.$$

This relation corresponds to (21.7) after approximating the liquid characteristic function at time $t = t^n$. We also assume to be given a velocity approximation $\mathbf{u}^n(\mathbf{x})$ of $\mathbf{u}(\mathbf{x}, t^n)$. The prediction step determines an approximation of the liquid domain at time t^{n+1} together with a prediction of the velocity on the new domain. The correction step provides an update of the velocity and pressure on the liquid domain that remains unchanged. Figure 21.3 provides an illustration of the process for the die swelling example.
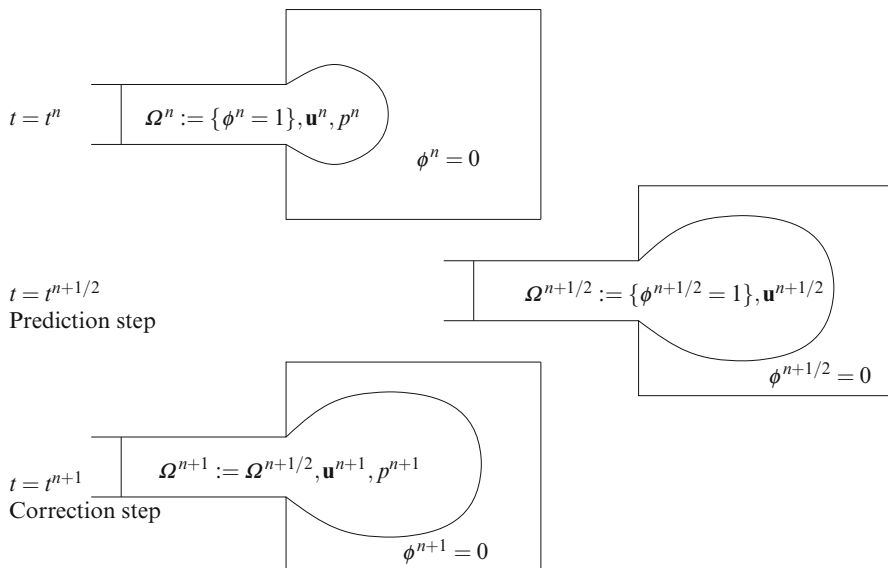
Fig. 21.3 Alternating direction splitting applied to free boundary problems. Given approximations ϕ^n and $\mathbf{u}^n$ of the liquid domain characteristic function ϕ and velocity $\mathbf{u}$ at time $t = t^n$, the first step consists in finding updated approximations of the characteristic function ϕ (and thus of the liquid domain Ω) as well as of the fluid velocity $\mathbf{u}$. On the new liquid domain, the second step determines a velocity correction together with its associated pressure. In particular, the liquid domain does not change during the correction step.

3.2.1 The Prediction Step

The *prediction step* encompasses the advection components of (21.1) and (21.8). It consists in simultaneously finding approximations of the characteristic function ϕ and the velocity field $\mathbf{u}$ satisfying

$$\frac{\partial}{\partial t}\phi + \mathbf{u}\cdot\nabla\phi = 0 \quad \text{and} \quad \frac{\partial}{\partial t}\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} = \mathbf{0} \quad \text{in} \quad Q^{n+1} := Q \cap \left(\Lambda \times I^{n+1}\right).$$

$$(21.13)$$

The numerical scheme proposed here relies on the so-called method of characteristics and is detailed now. For any point $\bar{\mathbf{x}} \in \Omega^n$ in the liquid domain, we define the characteristic trajectory $\mathbf{y}(.;\bar{\mathbf{x}})$ starting at $\bar{\mathbf{x}}$ by

$$\frac{d}{dt}\mathbf{y}(t;\bar{\mathbf{x}}) = \mathbf{u}(\mathbf{y}(t;\bar{\mathbf{x}}),t), \quad \text{for} \quad t \in I^{n+1}, \quad \text{and} \quad \mathbf{y}(t^n;\bar{\mathbf{x}}) = \bar{\mathbf{x}}. \qquad (21.14)$$

Along this characteristic trajectory, the transport relations in (21.8) read

$$\frac{d}{dt}\phi(\mathbf{y}(t;\bar{\mathbf{x}}),t) = 0 \qquad \text{and} \qquad \frac{d}{dt}\mathbf{u}(\mathbf{y}(t;\bar{\mathbf{x}}),t) = \mathbf{0}.$$

Hence, from the initial conditions $\phi(t^n) = \phi^n$ and $\mathbf{u}(t^n) = \mathbf{u}^n$ on Ω^n, we obtain

$$\phi(\mathbf{y}(t;\overline{\mathbf{x}}),t) = \phi^n(\overline{\mathbf{x}}) = 1 \qquad \text{and} \qquad \mathbf{u}(\mathbf{y}(t;\overline{\mathbf{x}}),t) = \mathbf{u}^n(\overline{\mathbf{x}}) \qquad (21.15)$$

as long as $\mathbf{y}(t;\overline{\mathbf{x}}) \in \Lambda$. We set $\phi(\mathbf{x},t) = 0$ whenever $\mathbf{x} \in \Lambda \setminus \{\mathbf{y}(t;\overline{\mathbf{x}}) \mid \overline{\mathbf{x}} \in \Omega^n\}$ so that these relations defines ϕ on $\Lambda \times I^{n+1}$ (and the associated liquid domain) as well as the velocity $\mathbf{u}$ on the liquid domain. As pointed out earlier, the algorithm does not need the velocity $\mathbf{u}(\mathbf{x},t)$ whenever $\mathbf{x} \in \Lambda \setminus \{\mathbf{y}(t;\overline{\mathbf{x}}) \mid \overline{\mathbf{x}} \in \Omega^n\}$. The *prediction step* ends upon setting

$$\phi^{n+\frac{1}{2}} := \phi(t^{n+1}) \qquad \text{in} \quad \Lambda,$$

and consequently

$$\Omega^{n+\frac{1}{2}} := \left\{ \mathbf{x} \in \Lambda \mid \phi^{n+\frac{1}{2}}(\mathbf{x}) = 1 \right\} := \left\{ \mathbf{y}(t^{n+1},\overline{\mathbf{x}}) \mid \overline{\mathbf{x}} \in \Omega^n \right\} \cap \Lambda \qquad (21.16)$$

as well as

$$\mathbf{u}^{n+\frac{1}{2}} := \mathbf{u}(t^{n+1}) \qquad \text{in} \quad \Omega^{n+\frac{1}{2}}.$$

3.2.2 The Correction Step

After the prediction step, the approximation of the liquid domain remains unchanged. In the framework of the splitting scheme described in Section 3.1, the "corrected" characteristic function satisfies

$$\frac{\partial}{\partial t}\phi = 0 \quad \text{in} \quad \Omega^{n+\frac{1}{2}} \times I^{n+1} \qquad \text{with} \qquad \phi(t^n) = \phi^{n+\frac{1}{2}} \quad \text{in} \quad \Omega^{n+\frac{1}{2}}.$$

As a consequence, we set $\phi^{n+1} := \phi^{n+\frac{1}{2}}$, $\Omega^{n+1} := \Omega^{n+\frac{1}{2}}$ and we note that the predicted velocity is now defined over Ω^{n+1}, i.e., $\mathbf{u}^{n+\frac{1}{2}} : \Omega^{n+1} \to \mathbb{R}^d$.

Then, the updated velocity $\mathbf{u} : \Omega^{n+1} \times I^{n+1} \to \mathbb{R}^d$ as well as the associated pressure $p : \Omega^{n+1} \times I^{n+1} \to \mathbb{R}$ are defined as the solution to the following Stokes system on a given non-moving domain:

$$\begin{cases} \rho\dfrac{\partial}{\partial t}\mathbf{u} - 2\nabla \cdot (\mu \mathbf{D}(\mathbf{u})) + \nabla p = \mathbf{f} \\[2mm] \nabla \cdot \mathbf{u} = 0 \end{cases} \qquad \text{in} \quad \Omega^{n+1} \times I^{n+1}, \qquad (21.17)$$

supplemented by the boundary conditions

$$\mathbf{u} = \mathbf{g}_D \quad \text{on} \quad \partial\Omega^{n+1} \cap \Gamma_{\mathcal{D}}, \quad (2\mu\mathbf{D}(\mathbf{u}) - p\mathbf{I})\mathbf{n}^{n+1} = \mathbf{g}_N \quad \text{on} \quad \partial\Omega^{n+1} \cap \Gamma_{\mathcal{N}},$$

and the free interface condition

$$(2\mu\mathbf{D}(\mathbf{u}) - p\mathbf{I})\mathbf{n}^{n+1} = \mathbf{0} \quad \text{on} \quad \partial\Omega^{n+1} \setminus \partial\Lambda,$$

where $\mathbf{n}^{n+1}$ is the outer pointing unit vector normal to $\partial\Omega^{n+1}$. Finally, we define the corrected velocity approximation $\mathbf{u}^{n+1} : \Omega^{n+1} \to \mathbb{R}^d$ by $\mathbf{u}^{n+1} := \mathbf{u}(t^{n+1})$ and the associated pressure by $p^{n+1} : \Omega^{n+1} \to \mathbb{R}$ by $p^{n+1} := p(t^{n+1})$.

4 Numerical Approximation of Free Surface Flows

We are now in a position to describe the numerical algorithm for the approximation of the solution to the free boundary problem (21.1) and (21.8). It takes full advantage of the splitting into *prediction* and *correction* steps discussed in Section 3.2. The time and space discretizations are presented in Sections 4.1 and 4.2 respectively. This section ends with Section 4.3, where numerical illustrations are given, in particular, in the context of die swell.

4.1 Time Discretization

We recall that the time interval $[0, T]$ is decomposed in N subintervals $I^n := (t^n, t^{n+1}]$, $n = 0, \dots, N-1$ and we denote the associated time steps by $\delta t^{n+1} := t^{n+1} - t^n$. In what follows, we discuss the algorithm over the time interval I^n.

4.1.1 Prediction Step

An *explicit Euler* approximation $\mathbf{Y}^{n+1}$ of the characteristic curve $\mathbf{y}(t^{n+1}; \bar{\mathbf{x}})$ in (21.14) is advocated for the prediction step. For all $\bar{\mathbf{x}} \in \Omega^n$, we set

$$\mathbf{Y}^{n+1}(\bar{\mathbf{x}}) := \bar{\mathbf{x}} + \delta t^{n+1} \mathbf{u}^n(\bar{\mathbf{x}}). \tag{21.18}$$

In view of (21.16), the approximation of the liquid domain $\Omega^{n+\frac{1}{2}}$, denoted $\Omega_N^{n+\frac{1}{2}}$, is defined as

$$\Omega_N^{n+\frac{1}{2}} := \left\{ \mathbf{Y}^{n+1}(\bar{\mathbf{x}}) \mid \bar{\mathbf{x}} \in \Omega^n \right\} \cap \Lambda.$$

The characteristic curves $\mathbf{Y}^{n+1}$ determine also the approximations $\Phi^{n+\frac{1}{2}}$ and $\mathbf{U}^{n+\frac{1}{2}}$ of $\phi^{n+\frac{1}{2}}$ and $\mathbf{u}^{n+\frac{1}{2}}$ according to the relations

$$\Phi^{n+\frac{1}{2}}(\mathbf{Y}^{n+1}(\bar{\mathbf{x}})) = \phi^n(\bar{\mathbf{x}}) := 1, \qquad \mathbf{U}^{n+\frac{1}{2}}(\mathbf{Y}^{n+1}(\bar{\mathbf{x}})) = \mathbf{u}^n(\bar{\mathbf{x}}), \tag{21.19}$$

whenever $\mathbf{Y}^{n+1}(\bar{\mathbf{x}}) \in \Lambda$. In addition, we set $\Phi^{n+\frac{1}{2}}(\mathbf{x}) = 0$ for $\mathbf{x} \in \Lambda \setminus \Omega_N^{n+\frac{1}{2}}$.

4.1.2 Correction Step

The approximation of the liquid domain characteristic function is not modified in this step, i.e.,

$$\Phi^{n+1} := \Phi^{n+\frac{1}{2}} \qquad \text{and} \qquad \Omega_N^{n+1} := \Omega_N^{n+\frac{1}{2}}.$$

An *implicit Euler* method is advocated for the solution of the Stokes system (21.17). This consists in seeking $\mathbf{U}^{n+1} : \Omega_N^{n+1} \to \mathbb{R}^d$ and $P^{n+1} : \Omega_N^{n+1} \to \mathbb{R}$ satisfying

$$\begin{cases} \rho \dfrac{\mathbf{U}^{n+1} - \mathbf{U}^{n+\frac{1}{2}}}{\delta t^{n+1}} - 2\nabla \cdot (\mu \mathbf{D}(\mathbf{U}^{n+1})) + \nabla P^{n+1} = \mathbf{f}(.,t^{n+1}), \\ \nabla \cdot \mathbf{U}^{n+1} = 0, \end{cases} \qquad (21.20)$$

in Ω_N^{n+1}, subject to the boundary conditions

$$\mathbf{U}^{n+1} = \mathbf{g}_D(.,t^{n+1}) \quad \text{on} \quad \partial\Omega_N^{n+1} \cap \Gamma_{\mathscr{D}},$$
$$(2\mu\mathbf{D}(\mathbf{U}^{n+1}) - P^{n+1}\mathbf{I})\mathbf{n}_N^{n+1} = \mathbf{g}_N(.,t^{n+1}) \quad \text{on} \quad \partial\Omega_N^{n+1} \cap \Gamma_{\mathscr{N}},$$

and to the free interface condition

$$(2\mu\mathbf{D}(\mathbf{U}^{n+1}) - P^{n+1}\mathbf{I})\mathbf{n}_N^{n+1} = \mathbf{0} \quad \text{on} \quad \partial\Omega_N^{n+1} \setminus \partial\Lambda.$$

Here $\mathbf{n}_N^{n+1}$ is the outer pointing unit vector normal to $\partial\Omega_N^{n+1}$.

4.2 Two-Grid Spatial Discretization

The space discretization takes also full advantage of the alternating splitting described above. The prediction and correction steps are approximated using different subdivisions and numerical techniques. On the one hand, a subdivision made of structured cells is advocated for the characteristic relation (21.18) coupled with a Simple Linear Interface Calculation (SLIC) [28] procedure in order to limit the numerical diffusion when approximating the volume fraction of liquid in (21.19). On the other hand, a standard stabilized finite element method is proposed for the approximation of the solution of the Stokes system (21.20). We start with the description of the two subdivisions and define the associated discrete approximation spaces. Then we detail the numerical techniques tailored to each discrete spaces.

4.2.1 Two Subdivisions and Associated Discrete Spaces

The prediction and correction steps rely on two different subdivisions. A volume-of-fluid type method on a structured grid is advocated for the prediction step consisting

of two transport equations (21.19). The computational domain Λ is bounded and therefore can be included into a structured grid of cells C_i, $i = 1, \ldots, M$. We denote by $\mathscr{P}^S := \{C_i, \ i = 1, \ldots, M\}$ the collection of all those structured cells and by $h := \max_{C \in \mathscr{P}^S} \operatorname{diameter}(C)$ the typical size of the elements. An example of such mesh is shown in Figure 21.4.
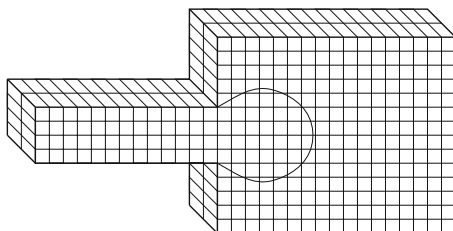

Fig. 21.4 Structured subdivision used for the space discretization during the prediction step.

We denote by $\mathbb{V}^S$ the approximation space which consists of all piecewise constant functions associated with the partition $\mathscr{P}^S$:

$$\mathbb{V}^S := \left\{ v : \Lambda \to \mathbb{R} \ \mid \ v|_C \text{ is constant } \forall C \in \mathscr{P}^S \right\}.$$

Note that $\mathbb{V}^S$ will be used as the approximation space for the liquid characteristic function Φ^n and the predicted velocity $\mathbf{U}^{n+\frac{1}{2}}$. In particular, the approximation of the former does not necessarily take values in $\{0, 1\}$ but in $\mathbb{R}$.

The second discretization considered is a typical conforming finite element subdivision made of triangles when $d = 2$ or tetrahedra when $d = 3$. The collection of these elements is denoted $\mathscr{T}^{FEM}$ and we denote by $H := \max_{T \in \mathscr{T}^{FEM}} \operatorname{diameter}(T)$ the typical size of the elements. An example of such a discretization is shown in Figure 21.5.
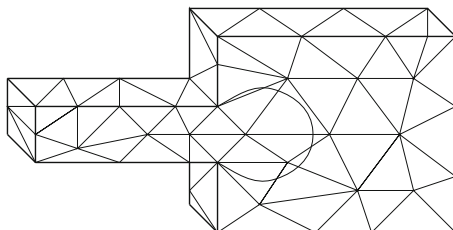

Fig. 21.5 Finite element subdivision used for the space discretization during the correction step.

For any subset $\tau \subset \mathscr{T}^{FEM}$, we denote by $\mathscr{V}(\tau)$ the collection of vertices in τ and by $\mathbb{V}^{FEM}(\tau)$ the space of globally continuous, piecewise polynomials of degree ≤ 1 associated with the subdivision τ:

$$\mathbb{V}^{FEM}(\tau) := \left\{ V : \bigcup_{T \in \tau} T \to \mathbb{R} \mid V \text{ continuous}, \ V|_T \text{ is a polynomial of degree } 1, \forall T \in \tau \right\}$$

and

$$\mathbb{V}_0^{FEM}(\tau) := \left\{ V \in \mathbb{V}^{FEM}(\tau) \mid V|_{\Gamma_{\mathscr{D}} \cap \partial T} = 0 \quad \forall T \in \tau \right\}.$$

In the sequel, the subset τ will represent the "liquid" elements, i.e., an approximate subdivision of $\Omega(t)$ at a given time t.

In order to fully exploit the potentialities of this two-grid method, we consider a structured grid $\mathscr{T}^S$ that is finer than the finite element mesh $\mathscr{T}^{FEM}$. This allows us to improve the accuracy on the approximation of the transport equations (21.18), without having a computationally prohibitive approximation of the diffusion problem (21.20). As it turns out, this allows choices of relatively large CFL, see Section 4.3. Typically the value of H is between $5h$ and $10h$, namely the structured grid is five to ten times finer than the finite element mesh. Further comments about the choice of the sizes of both discretizations can be found in [12, 26].

To alternate between the prediction and correction steps, we need projection operators to map functions in $\mathbb{V}^S$ into functions in $\mathbb{V}^{FEM}(\tau)$ and vice versa. We start with the projection $\pi_{S \to FEM} : \mathbb{V}^S \to \mathbb{V}^{FEM}(\tau)$ mapping the structured grid into the finite element mesh. Note that a function in $\mathbb{V}^{FEM}(\tau)$ is uniquely determined by its values on the vertex set $\mathscr{V}(\tau)$ and it is thus sufficient for the projection operator to set these values. Hence, for any $V \in \mathbb{V}^S$ and any $\mathbf{v} \in \mathscr{V}(\tau)$, we define

$$(\pi_{S \to FEM} V)(\mathbf{v}) := \frac{\sum_{T \in \tau \,:\, \mathbf{v} \in \mathscr{V}(T)} \sum_{C \in \mathscr{P}^S, \, C \subset T} \phi_{\mathbf{v}}(\text{center}(C)) V(\text{center}(C))}{\sum_{T \in \tau \,:\, \mathbf{v} \in \mathscr{V}(T)} \sum_{C \in \mathscr{P}^S, \, C \subset T} \phi_{\mathbf{v}}(\text{center}(C))}, \quad (21.21)$$

where $\text{center}(C)$ denotes the (barycentric) center of the cell C and $\phi_{\mathbf{v}}$ denotes the Lagrange piecewise linear basis function associated with the vertex $\mathbf{v}$. The notation $C \subset T$ indicates that $\text{center}(C) \in T$. We denote identically the projection of a scalar-valued function or of a vector-valued function for which the projection is applied component-wise. In Figure 21.6, we have depicted a sketch in two dimensions of the set of cells of $\mathscr{T}^S$ appearing in the above summation for the calculation of the value at a vertex of $\mathscr{V}(\tau)$ in a typical structured subdivision.

The projection from the finite element subdivision to the structured mesh is defined for any $V \in \mathbb{V}^{FEM}(\tau)$ as follows. First, we extend the function V to the entire computational domain Λ by 0. Then, for each cell $C \in \mathscr{T}^S$, we denote by $T(C) \in \mathscr{T}^{FEM}$ the element in the finite element subdivision containing the center of the cell C and define

$$(\pi_{FEM \to S} V)|_C := \sum_{\mathbf{v} \in \mathscr{V}(T(C))} \phi_{\mathbf{v}}(\text{center}(C)) V(\mathbf{v}). \quad (21.22)$$

If the cell center is exactly at the boundary of several elements of the finite element mesh, then one arbitrary (but fixed) element is chosen among the possible elements. Again, we denote identically the projection of a scalar-valued function or of a vector-valued function (computed component-wise).
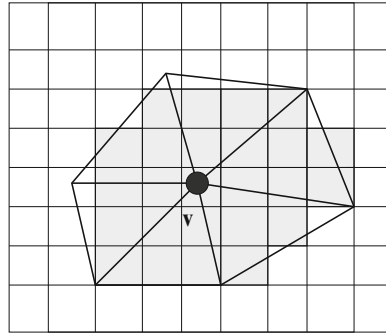
Fig. 21.6 The shaded cells correspond to all the cells appearing in the average projection (21.21) in order to determine the value of the function in $\mathbb{V}^{FEM}$ at the node **v**. The Lagrange basis function $\phi_{\mathbf{v}}$ is the piecewise linear function (subordinate to the finite triangular subdivision) with value 1 at **v** and 0 at the other vertices.

Remark 2 (Implementation). The two projection operators (21.21) and (21.22) require a data structure mapping each cell of the structured mesh to an element of the finite element subdivision (the element containing the cell center). This array of indices is computed once and for all at the beginning of the simulation. However, when allowing for mesh adaptations an updated array is required after each mesh modification.

4.2.2 Prediction Step

The prediction steps start with given approximations $\Phi_M^n \in \mathbb{V}^S$ and $\mathbf{U}_M^n \in (\mathbb{V}^S)^d$ of the liquid fraction and velocity respectively, on the structured grid of cells (recall that M denotes the number of structured cells in the subdivision). As noted earlier, although $\phi(\mathbf{x},t) \in \{0,1\}$, its approximation takes values in $\mathbb{R}$. However, the resulting numerical diffusion is counter-balanced by the SLIC and decompression algorithms described below.

We define the approximation $\mathbf{Y}_M^{n+1} \in (\mathbb{V}^S)^d$ of the characteristic trajectories $\mathbf{Y}^{n+1}$ as follows. As $\mathbf{Y}_M^{n+1}$ is constant over each cell, it suffices to determine its values at the centers $\mathbf{x}_i$ of each cell C_i, $i = 1,\ldots,M$ and we set

$$\mathbf{Y}_M^{n+1}(\mathbf{x}_i) := \mathbf{x}_i + \delta t^{n+1} \mathbf{U}_M^n(\mathbf{x}_i). \tag{21.23}$$

The image via $\mathbf{Y}_M^{n+1}$ of each cell C_i is denoted $\widetilde{C}_i$, i.e., $\widetilde{C}_i := \mathbf{Y}_M^{n+1}(C_i)$, so that

$$\left\{ C \in \mathscr{P}^S \ : \ C_i \cap \widetilde{C} \neq \emptyset \right\}$$

corresponds to all the cells (at least partially) transported to the cell C_i. As a consequence, the approximation $\Phi_M^{n+\frac{1}{2}}(\mathbf{x}_i)$ of the liquid characteristic function $\Phi^{n+\frac{1}{2}}(\mathbf{x}_i)$

defined by (21.15) is obtained by adding the (weighted) contribution from cells transported to C_i, that is

$$\Phi_M^{n+\frac{1}{2}}(\mathbf{x}_i) := \sum_{C \in \mathscr{P}^S} \Phi_M^n(\text{center}(C))|C_i \cap \widetilde{C}|, \qquad (21.24)$$

where $|C_i \cap \widetilde{C}|$ denotes the measure of $C_i \cap \widetilde{C}$. Due to the Cartesian properties of the structured grid $\mathscr{P}^S$, this measure is straightforward to compute as $C_i \cap \widetilde{C}$ are parallelepiped rectangles. Figure 21.7 illustrates the transport of one (two-dimensional) cell C into $\widetilde{C}$, which overlaps four other cells.
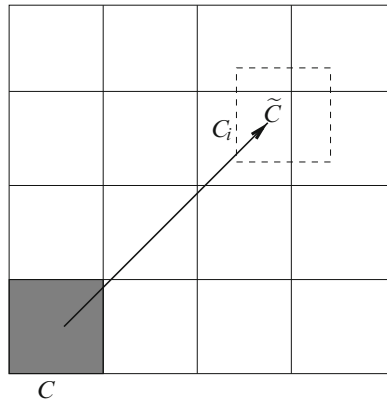
Fig. 21.7 Approximation of $\Phi^{n+\frac{1}{2}}$ using the method of characteristics. The cell C is transported to $\widetilde{C}$ and the quantity $\Phi_M^n(\mathbf{x}_j)$ is distributed among the intersecting cells. The contribution to $\Phi_M^{n+\frac{1}{2}}(\mathbf{x}_i)$ from $\Phi_M^n(\text{center}(C))$ is $\Phi_M^n(\text{center}(C))|C_i \cap \widetilde{C}|$ according to relation (21.24). The elements $C_i \cap \widetilde{C}$ are rectangles, making the computation of $|C_i \cap \widetilde{C}|$ straightforward.

We emphasize again that in view of relation (21.24) the liquid domain characteristic approximation $\Phi_M^{n+\frac{1}{2}}$ values are thus not necessarily 0 or 1 but could be any positive real number. In fact, this numerical diffusion (values strictly between 0 and 1) and numerical compression (values strictly larger than 1) are the two drawbacks of the projection formula (21.7), and are addressed now.

Numerical diffusion manifests itself when cells are partially filled, i.e., $0 < \Phi_M^{n+\frac{1}{2}}(\text{center}(C)) < 1$ for some $C \in \mathscr{P}^S$. Since the exact volume fraction of liquid ϕ is a step function and discontinuous at the free surface, numerical diffusion around the interfaces has to be controlled by the numerical scheme. It is reduced by the so-called SLIC algorithm [28], where the contribution to $\Phi_M^{n+\frac{1}{2}}(\text{center}(C_i))$ of partially filled cell C_i is still proportional to $|\widetilde{C} \cap C_i|$ but depends in addition on the values of the Φ_M^n on the neighboring cells of C. More precisely, before being transported along the characteristics, the quantities $\Phi_M^n(\text{center}(C))$ are concentrated

near the boundary of the cell C instead of being spread out in the entire cell. This procedure is illustrated in Figure 21.8 (bottom), and allows to reduce the error due to the projection of the transported quantity in $\widetilde{C}$ across several cells C_i. The way the quantity $\Phi_M^n(\text{center}(C_i))$ is pushed towards the boundary of the cell depends on the neighboring values of the volume fraction. Examples in two dimensions of space are illustrated in Figure 21.8 (top). We refer to [25, 26] for a more detailed description of the algorithm.
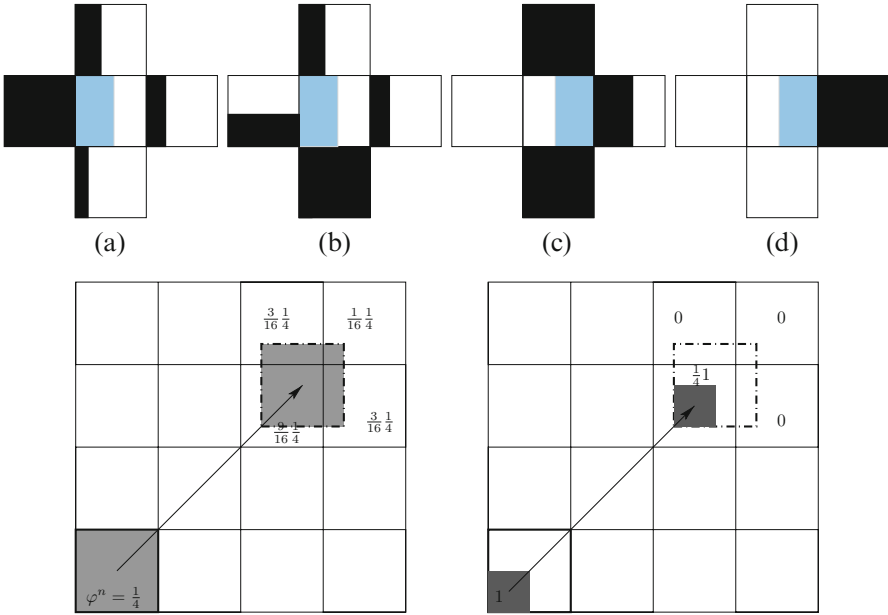
Fig. 21.8 (Top) Effect of the two dimensional SLIC algorithm on the cell center for four possible interfaces. The quantity $\Phi_M^n(\mathbf{x}_j)$, in blue, is pushed back to the sides of C depending on the values of Φ_M^n in the neighboring cells, in black. (Bottom) An example of two dimensional advection and projection when the volume fraction of liquid in the cell is $\Phi_M^n(\mathbf{x}_j) = \frac{1}{4}$. Left: without SLIC, the volume fraction of liquid is advected and projected on four cells, with contributions (from the top left cell to the bottom right cell) $\frac{3}{16}\frac{1}{4}$, $\frac{1}{16}\frac{1}{4}$, $\frac{9}{16}\frac{1}{4}$, $\frac{3}{16}\frac{1}{4}$. Right: with SLIC, the volume fraction of liquid is first pushed at one corner, then it is advected and projected on one cell only, with contribution $\frac{1}{4}$.

Let us discuss the compression case, i.e., $\Phi_M^{n+\frac{1}{2}}(C) > 1$ for some cell C. In that case, the excess $\Phi_M^{n+\frac{1}{2}}(C) - 1$ is stored in a buffer and redistributed into partially filled cells in order to *decompress* the field $\Phi_M^{n+\frac{1}{2}}$. Our algorithm redistributes the excess of liquid in a global way into cells that are in a neighborhood of the interface first (cells that are partially filled). Therefore it allows to conserve the mass in a global sense, in a way that is similar to *global repair* algorithms [35].

More precisely, we proceed in two steps: first, we compute the excess of liquid $\Phi_M^{n+\frac{1}{2}}(C) - 1$ in each cell C after advection and projection onto $\mathscr{P}^\mathscr{S}$; second, we redistribute these amounts into partially filled cells, starting with cells that are nearly full. The detailed algorithm can be found in [25, 26] and is illustrated in Figure 21.9 when $\mathscr{P}^\mathscr{S}$ is a single layer of cells. Although this figure represents a one-dimensional, over-simplified situation, it illustrates the rebalancing principle that allows to conserve the mass at each time step.

$\Phi_M^{n+\frac{1}{2}}$ before the decompression algorithm

| 0.5 | 0.9 | 1.7 | 0.8 | 0.4 |

| 0.9 | 1.0 | 1.0 | 1.0 | 0.4 |

$\Phi_M^{n+\frac{1}{2}}$ after the decompression algorithm
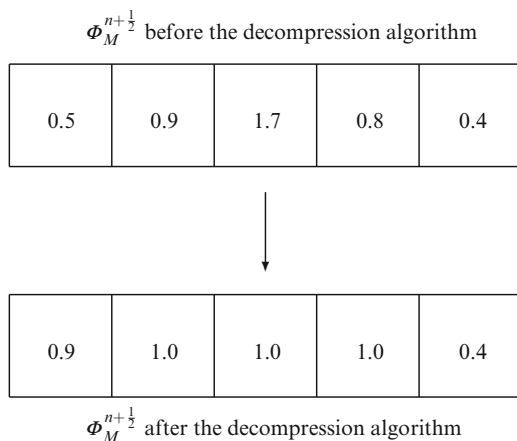
Fig. 21.9 Decompression algorithm. The volume fraction in excess in some cells is redistributed into the partially filled cells. Here the excess of 0.7 in the middle cell is redistributed in the partially filled cells, starting with the ones that are nearly full (0.9, 0.8 and 0.5 in order).

Similarly to (21.24), the velocity approximation $\mathbf{U}_M^{n+\frac{1}{2}}$ is given by the formula

$$\mathbf{U}_M^{n+\frac{1}{2}}(\mathbf{x}_i) := \sum_{C \in \mathscr{P}^\mathscr{S}} \mathbf{U}_M^n(\text{center}(C)) |C_i \cap \widetilde{C}|; \tag{21.25}$$

but the SLIC and decompression algorithm are not applied to the approximation of the velocity.

At the end of the prediction step, the projection onto the finite element space $\Phi_K^{n+\frac{1}{2}} \in \mathbb{V}^{FEM}$ of $\Phi_M^{n+\frac{1}{2}}$ is computed using the operators defined in Section 4.2.1

$$\Phi_K^{n+\frac{1}{2}} := \pi_{S \to FEM} \Phi_M^{n+\frac{1}{2}} \in \mathbb{V}^{FEM}. \tag{21.26}$$

The latter is used to define the liquid domain: An element $T \in \mathscr{T}^{FEM}$ is said to be liquid if $\max_{\mathbf{x} \in T} \Phi_K^{n+1/2}(\mathbf{x}) \geq 1/2$, the set of all liquid elements is then denoted by $\tau_K^{n+\frac{1}{2}}$, and the liquid domain $\Omega_K^{n+\frac{1}{2}}$ is the union of all liquid elements. The choice of the value $1/2$ for the threshold is arbitrary. It has been empirically discussed in

[26], but results have shown little sensitivity with respect to the value of this parameter. However, this definition of the liquid domain $\Omega_K^{n+\frac{1}{2}}$ implies an approximation error of the order $O(H)$ on the approximation of the free surface. Mesh refinement techniques have been designed to address this drawback [9], but are not developed further here.

The velocity is not directly projected onto the finite element space as the projection would depend on the values of the velocity outside $\Omega_K^{n+\frac{1}{2}}$ (which do not exist). Instead, we project $\Phi_M^{n+\frac{1}{2}} \mathbf{U}_M^{n+\frac{1}{2}}$ and recover the velocity $\mathbf{U}_K^{n+\frac{1}{2}} \in \mathbb{V}^{FEM}(\tau_K^{n+\frac{1}{2}})^d$ at each vertex $\mathbf{v} \in \mathscr{V}(\tau_K^{n+\frac{1}{2}})$ as follows:

$$\mathbf{U}_K^{n+\frac{1}{2}}(\mathbf{v}) := \frac{(\pi_{S \to FEM}(\Phi_M^{n+\frac{1}{2}} \mathbf{U}_M^{n+\frac{1}{2}}))(\mathbf{v})}{\Phi_K^{n+\frac{1}{2}}(\mathbf{v})} \qquad (21.27)$$

if $\mathbf{v} \notin \Gamma_{\mathscr{D}}$ and $\mathbf{U}_K^{n+\frac{1}{2}}(\mathbf{v}) = \mathbf{g}_{\mathscr{D}}(\mathbf{v})$ otherwise. Notice that the above expression defines $\mathbf{U}_K^{n+\frac{1}{2}}$ only on $\Omega_K^{n+\frac{1}{2}}$ but only these values are needed in the correction step.

4.2.3 Correction Step

As already mentioned, the liquid characteristic function is not modified during this step; so we set

$$\Phi_K^{n+1}(\mathbf{v}) := \Phi_K^{n+\frac{1}{2}}(\mathbf{v})$$

for all $\mathbf{v} \in \mathscr{V}(\tau_K^{n+\frac{1}{2}})$, and

$$\Omega_K^{n+1} := \Omega_K^{n+\frac{1}{2}} \qquad \text{and} \qquad \tau_K^{n+1} := \tau_K^{n+\frac{1}{2}}.$$

Then, the Stokes system (21.20) on the *fixed* liquid domain Ω_K^{n+1} and with vanishing Dirichlet boundary condition $\mathbf{g}_{\mathscr{D}} \equiv 0$[1] reads as follows.

Seek $\mathbf{U}_K^{n+1} \in \mathbb{V}_0^{FEM}(\tau_K^{n+1})^d$ and $P_K^{n+1} \in \mathbb{V}^{FEM}(\tau_K^{n+1})$ satisfying

$$B^{n+1}((\mathbf{U}_K^{n+1}, P_K^{n+1}), (\mathbf{V}, R)) + S^{n+1}((\mathbf{U}_K^{n+1}, P_K^{n+1}), (\mathbf{V}, R)) = L^{n+1}(\mathbf{V}) \qquad (21.28)$$

for any $(\mathbf{V}, R) \in \mathbb{V}_0^{FEM}(\tau_K^{n+1})^d \times \mathbb{V}^{FEM}(\tau_K^{n+1})$. The bilinear functional $B^{n+1} : (\mathbb{V}_0^{FEM}(\tau_K^{n+1})^d \times \mathbb{V}^{FEM}(\tau_K^{n+1})) \times (\mathbb{V}_0^{FEM}(\tau_K^{n+1})^d \times \mathbb{V}^{FEM}(\tau_K^{n+1})) \to \mathbb{R}$ is defined as

$$B^{n+1}((\mathbf{U}, P), (\mathbf{V}, R)) := \frac{\rho}{\delta t^{n+1}} \int_{\Omega_K^{n+1}} \mathbf{U} \cdot \mathbf{V} \, d\mathbf{x} + 2\mu \int_{\Omega_K^{n+1}} \mathbf{D}(\mathbf{U}) :: \mathbf{D}(\mathbf{V}) \, d\mathbf{x}$$
$$- \int_{\Omega_K^{n+1}} R \, \nabla \cdot \mathbf{V} \, d\mathbf{x} + \int_{\Omega_K^{n+1}} P \, \nabla \cdot \mathbf{V} \, d\mathbf{x},$$

---

[1] The case of non-vanishing Dirichlet boundary conditions reads similarly upon defining a lifting of the boundary conditions.

where $A :: B := \sum_{i,j=1}^{d} A_{ij}B_{ij}$, for $A, B \in \mathbb{R}^{d \times d}$. The right-hand side $L^{n+1} : \mathbb{V}_0^{FEM}$ $(\tau_K^{n+1})^d \to \mathbb{R}$ is given by

$$L^{n+1}(\mathbf{V}) := \frac{\rho}{\delta t^{n+1}} \int_{\Omega_K^{n+1}} \mathbf{U}^n \cdot \mathbf{V} \, d\mathbf{x} + \int_{\Omega_K^{n+1}} \mathbf{f}(t^{n+1}) \cdot \mathbf{V} \, d\mathbf{x}.$$

The functionals S^{n+1} in (21.28) are the Galerkin Least-Square stabilization terms to cope with the fact that the choice of the finite element spaces is not inf-sup stable. They are given by:

$$S^{n+1}((\mathbf{U}, P), (\mathbf{V}, R)) := \sum_{T \subset \Omega_K^{n+1}} \alpha_T \int_T \left(\rho \frac{\mathbf{U} - \mathbf{U}^n}{\delta t^{n+1}} + \nabla P - \mathbf{f}(t^{n+1}) \right) \cdot \nabla R \, d\mathbf{x},$$

where α_T is the stabilization coefficient locally defined as:

$$\alpha_T := \begin{cases} C_{SUPG} \dfrac{\text{diam}(T)^2}{12\mu} & \text{if } 0 \leq Re_T \leq 3 \\[2mm] C_{SUPG} \dfrac{\text{diam}(T)^2}{4Re_T\mu} & \text{if } 3 \leq Re_T \end{cases}$$

where the local Reynolds number is defined by $Re_T := \frac{\rho \, \text{diam}(T) \, \max_{\mathbf{x} \in T} |\mathbf{U}^n|}{2\mu}$ and C_{SUPG} is an dimensionless constant typically set to 1.0.

At the end of the correction step, the velocity is projected onto the structured grid

$$\mathbf{U}_M^{n+1} := \pi_{FEM \to S} \mathbf{U}_K^{n+1} \in (\mathbb{V}^S)^d,$$

while the volume fraction of liquid remains unchanged

$$\Phi_M^{n+1} := \Phi_M^{n+\frac{1}{2}} \in \mathbb{V}^S.$$

4.3 Numerical Results for Newtonian Flows

The example that serves as a guideline in this work is the experiment of the die swell with contraction in an extrusion process. This benchmark not only illustrates the advantages of the splitting approach presented in this work, but it is also worth noting that numerical simulation of extrusion is of great importance in industrial processes, for instance for pasta dough in food engineering [22].

We consider an axisymmetric capillary die with a contraction at the entrance. The fluid is injected into the die and then expands at the exit. The behavior of the fluid depends strongly on the fluid rheology. In this section, we consider Newtonian fluids and refer to Sections 5 and 6 for more complex fluids.

4.3.1 Extrusion with Initial Contraction: Computational Domain

We describe the computational domain used for the subsequent extrusion experiments with initial contraction. The computational domain Λ is depicted in Figures 21.10 and 21.11, together with the finite element mesh used for (most of) the simulations presented in this work. It consists of three cylinder as depicted in Figure 21.10. The first cylinder of diameter 0.010 m and length 0.005 m is where the liquid is injected. Then, the liquid enters the die, a second cylinder of diameter 0.001 m (contraction) and length 0.010 m. When, exiting the die, the liquid enters the third cylinder of diameter 0.010 m and length 0.015 m. The total length of the domain is therefore 0.030 m. The size of the finite elements in the die is characterized by $H = 0.0001$ m. The structured grid consists of a subdivision made of cubic cells of length 0.000025 m.
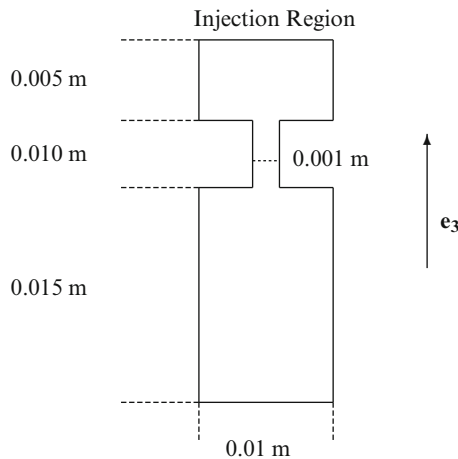
Fig. 21.10 Extrusion with Initial Contraction: Dimensions of the computational domain.

4.3.2 Slip Boundary Conditions

We consider a Newtonian fluid with density $\rho = 1300$ kg m^{-3}, and viscosity $\mu = 10$ kg(ms)$^{-1}$. The fluid is injected with a constant speed of 0.00023 ms^{-1}, such that the speed in the die is approximately 0.05 ms^{-1}. No-slip boundary conditions are imposed at the bottom of the domain and slip boundary conditions are imposed on the other parts of $\partial\Lambda$. We postpone to Section 4.3.3 for a discussion on the effect of different types of boundary conditions. Gravity forces, $\mathbf{g} = -9.81\mathbf{e_3}$ ms^{-2} are oriented along the die (see Figure 21.10). The time step is constant and equal to $\delta t = 0.005$ s, which implies a CFL number of about 10 during the prediction step and of about 2.5 during the correction step. At time $t = 0.9$ s approximately, the
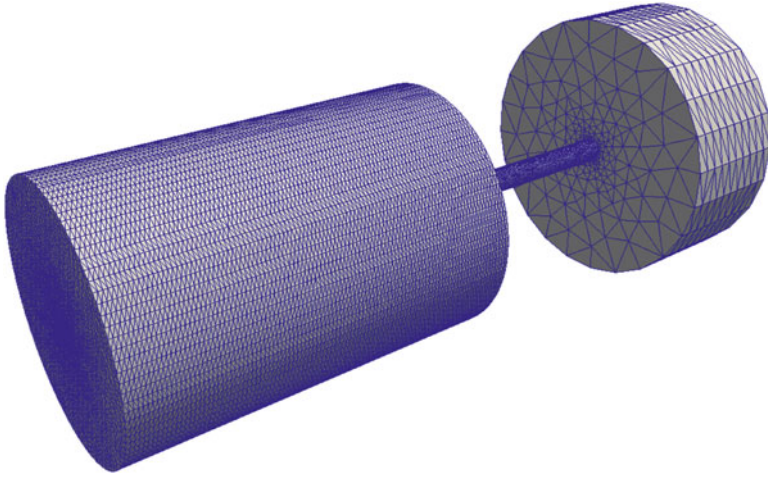
Fig. 21.11 Extrusion with Initial Contraction: Computational domain and finite element mesh.

jet hits the bottom of the computational domain and the fluid buckles. Figure 21.12 visualizes, in a medium plane inside the tube, snapshots of the volume fraction of liquid Φ and of the corresponding velocity field **U**. Figure 21.13 visualizes the buckling of the jet of Newtonian fluid once it hits the bottom of the computational domain.

In this case, we observe that the operator splitting scheme does not introduce any additional error as long as the flow is laminar and does not touch the bottom of the domain. This allows to consider large time steps if needed, without any CFL condition. Little oscillations in the jet are observed due to the spatial discretization and the unstructured finite element mesh. The buckling effect when the flow touches the boundary requires smaller time steps to retain accuracy.

4.3.3 No-Slip Boundary Conditions

When enforcing slip boundary conditions on the lateral side as in the previous test case, the liquid has a constant velocity (until it hits the bottom) so that the operator splitting produces the exact solution for any value of the time step; see Figure 21.12. For this simulation, we impose no-slip boundary condition on $\partial \Lambda$ except at the inflow where we keep the constant velocity profile of magnitude 0.00023 ms^{-1}. A Poiseuille profile is observed for the velocity in the die with a slight perturbation due to the contraction. Figure 21.14 shows the results obtained with the setup described in Sections 4.3.1 and 4.3.2 but with no-slip boundary conditions in the cavity before the die and in the die; compare with Figure 21.12.

The effect of boundary conditions is amplified for liquids with larger viscosities. Figure 21.15 provides a similar simulation when the viscosity is 10 times larger ($\mu = 100$ kg(ms)$^{-1}$). It demonstrates the effect of no-slip boundary conditions on the shape of the liquid front and on the free surface front velocity, which decreases as the viscosity increases.
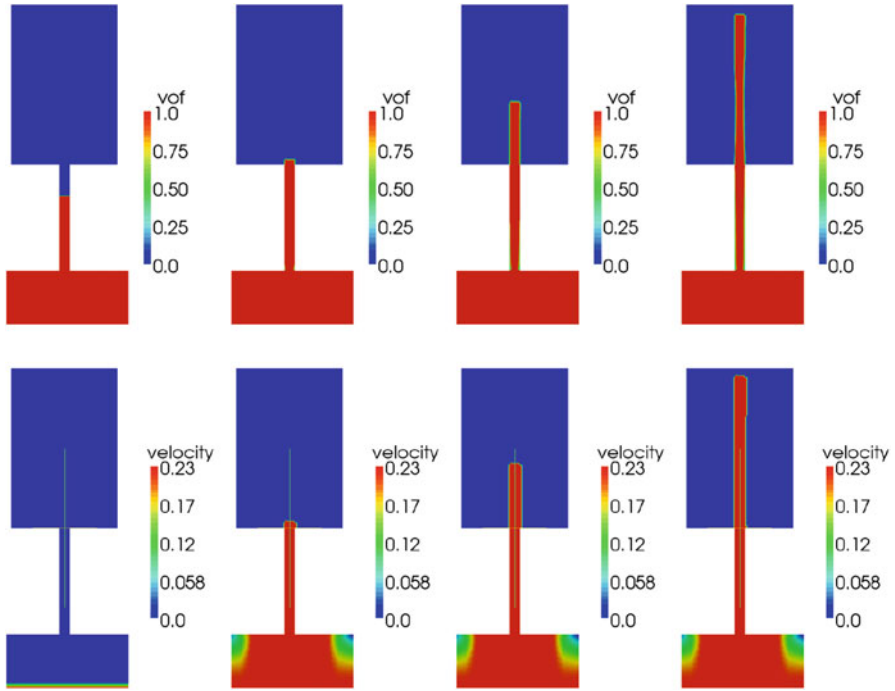
Fig. 21.12 Die swell with extrusion of a Newtonian fluid. Snapshots of the solution at times $t = 0$, 0.3, 0.6, and 0.9 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ; bottom: speed $|\mathbf{U}|$.
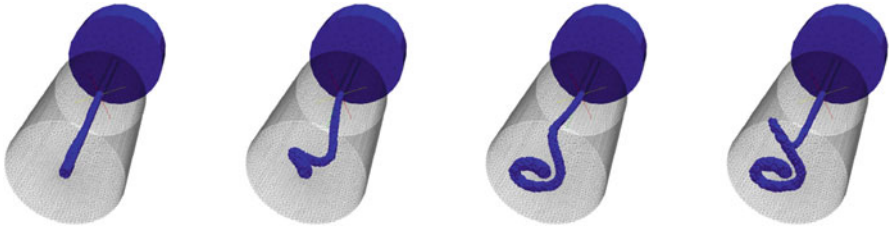

Fig. 21.13 Die swell with extrusion of a Newtonian fluid. Snapshots of the buckling of the jet at times $t = 1.0, 1.2, 1.4$, and 1.6 s (left to right).

Figures 21.16 and 21.17 show snapshots of the buckling effects for $\mu = 10$ kg(ms)$^{-1}$ and $\mu = 100$ kg(ms)$^{-1}$ respectively, and no-slip boundary conditions. The boundary conditions change drastically the shape of the liquid during the buckling. In addition, larger viscosities slow the liquid front propagation and reduce the buckling effect as it was already noted in the simple cavity setting [6, 36].
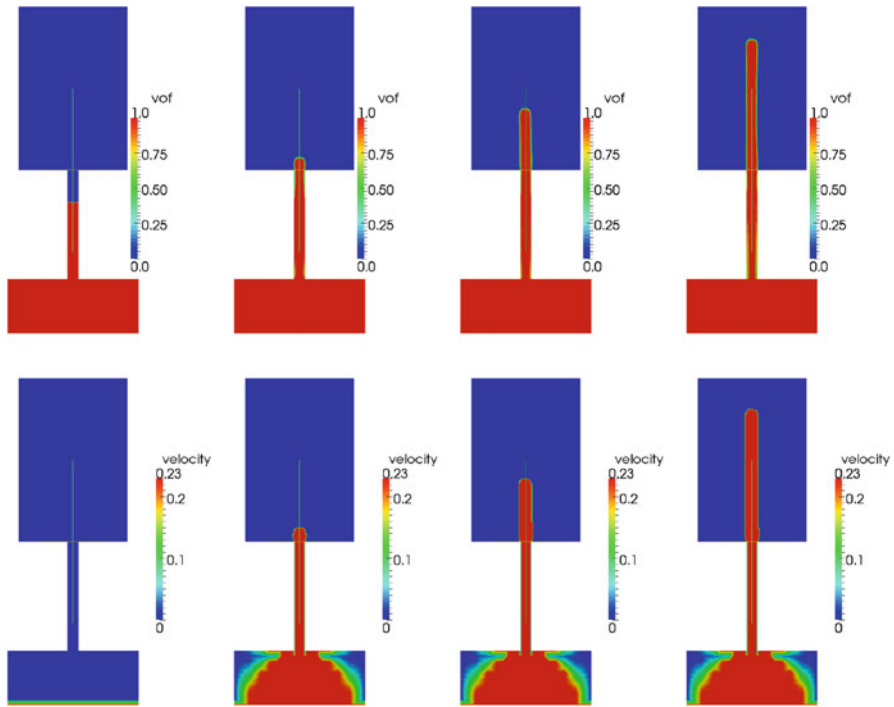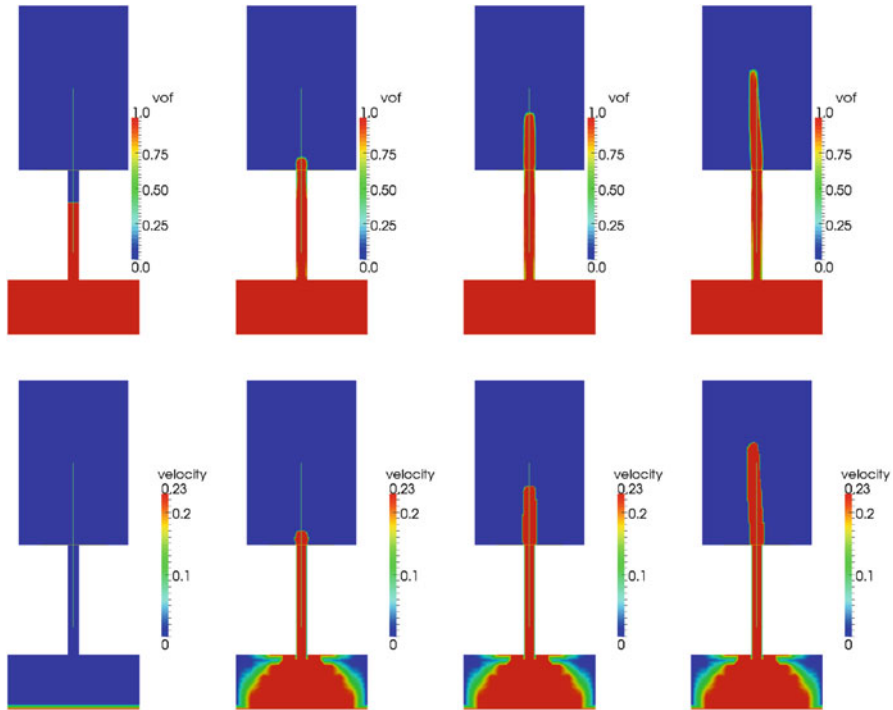
Fig. 21.14 Die swell with extrusion of a Newtonian fluid, with no-slip boundary conditions in the die ($\mu = 10$ kg(ms)$^{-1}$). Snapshots of the solution at times $t = 0, 0.3, 0.6$, and 0.9 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ; bottom: speed $|\mathbf{U}|$.

5 Visco-Elastic Flows with Free Surfaces

We now discuss an extension to liquids with more complex rheology and in particular the modification of the Navier-Stokes system (21.1) to account for visco-elastic effects. The upper-convected Maxwell model is chosen to describe the complex rheology but the algorithm presented here is not restricted to specific models.

5.1 Mathematical Modeling of Visco-Elastic Flows with Free Surfaces

Visco-elastic fluids are characterized by the presence of an extra-stress tensor denoted by $\sigma \in \mathbb{R}^{\frac{d(d+1)}{2}} \simeq \mathbb{R}^{d \times d}_{sym}$, the space of $d \times d$ symmetric tensors, supplementing the Cauchy stress tensor $2\mu \mathbf{D}(\mathbf{u}) - p\mathbf{I}$ in (21.1). Hence, the velocity $\mathbf{u}$, pressure p and visco-elastic stress σ satisfy in Q:

Fig. 21.15 Die swell with extrusion of a Newtonian fluid, with no-slip boundary conditions in the die ($\mu = 100$ kg(ms)$^{-1}$). Snapshots of the solution at times $t = 0, 0.3, 0.6$, and 0.9 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ; bottom: speed $|\mathbf{U}|$. Compare with Figure 21.14 representing the same setting but with a fluid of smaller viscosity.
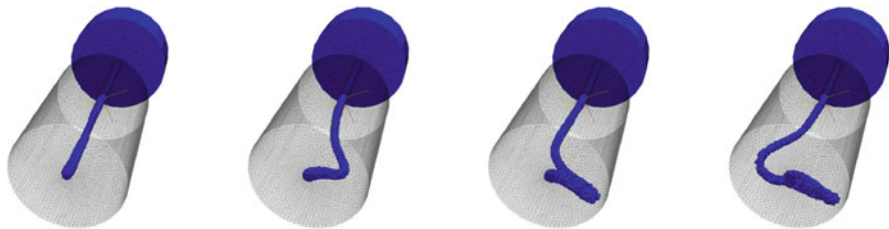

Fig. 21.16 Die swell with extrusion of a Newtonian fluid, with no-slip boundary conditions in the die ($\mu = 10$ kg(ms)$^{-1}$). Snapshots of the buckling of the jet at times $t = 1.0, 1.2, 1.4$, and 1.6 s (left to right).

$$\begin{cases} \rho \left(\dfrac{\partial}{\partial t}\mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} \right) - \nabla \cdot (2\mu \mathbf{D}(\mathbf{u})) + \nabla p - \nabla \cdot \sigma = \mathbf{f}, \\ \nabla \cdot \mathbf{u} = 0. \end{cases} \qquad (21.29)$$
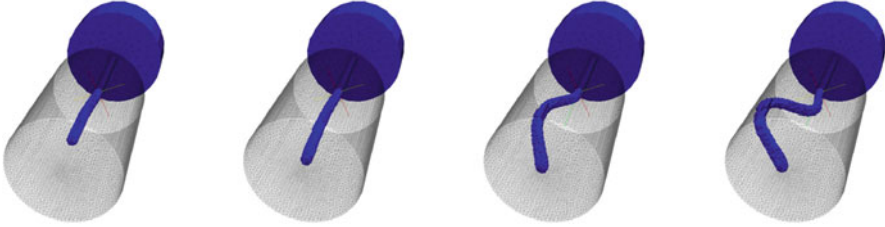
Fig. 21.17 Die swell with extrusion of a Newtonian fluid, with no-slip boundary conditions in the die ($\mu = 100$ kg(ms)$^{-1}$). Snapshots of the buckling of the jet at times $t = 1.0, 1.2, 1.4$, and 1.6 s (left to right).

The Dirichlet condition (21.2) on the velocity field remains unchanged but the Neumann (21.3) as well as the interface (21.4) conditions are modified to account for the presence of the visco-elastic stress

$$(2\mu \mathbf{D}(\mathbf{u}) - p\mathbf{I} + \sigma)\mathbf{n} = \mathbf{g}_{\mathscr{N}} \quad \text{on} \quad \partial Q_{\mathscr{N}}, \tag{21.30}$$

$$(2\mu \mathbf{D}(\mathbf{u}) - p\mathbf{I} + \sigma)\mathbf{n} = \mathbf{0} \quad \text{on} \quad \mathscr{F}. \tag{21.31}$$

As model problem, we consider the upper-convected Maxwell model to provide the constitutive relation for σ, namely the extra-stress σ satisfies:

$$\sigma + \lambda \left(\frac{\partial}{\partial t}\sigma + (\mathbf{u} \cdot \nabla)\sigma - \nabla \mathbf{u} \, \sigma - \sigma \nabla \mathbf{u}^t \right) = 2\mu_p \mathbf{D}(\mathbf{u}) \quad \text{in} \quad Q, \tag{21.32}$$

where λ is the fluid relaxation time, μ_p is the so-called polymer viscosity [4, 5, 30]. The problem is thus coupled via the introduction of the extra-stress σ in the Navier-Stokes equations, and reciprocally, the velocity $\mathbf{u}$ in the constitutive equation for σ. The values of the stress tensor are set to a given tensor $\mathbf{G} : \partial Q_{inflow} \to \mathbb{R}^{\frac{d(d+1)}{2}}$ on the inflow boundary of the domain:

$$\sigma = \mathbf{G}, \quad \text{on} \quad \partial Q_{inflow}.$$

Similar to the initial conditions (21.6) for the velocity field, the initial viscoelastic stress is set to be

$$\sigma(0) = \sigma_0 \quad \text{on} \quad \Omega(0)$$

for a given $\sigma_0 : \Omega(0) \to \mathbb{R}^{\frac{d(d+1)}{2}}$.

5.2 Extension of the Operator Splitting Strategy

The prediction and correction steps described in Section 3.2 extend naturally. The constitutive relation (21.32) for σ also contains a transport relation which is accounted for in the prediction step.

5.2.1 The Prediction Step

The prediction step (21.13) becomes: find the characteristic function ϕ, the velocity field $\mathbf{u}$, and the extra-stress σ satisfying

$$\frac{\partial}{\partial t}\phi + \mathbf{u} \cdot \nabla\phi = 0$$

$$\frac{\partial}{\partial t}\mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} = 0 \qquad \text{in} \quad Q^{n+1} := Q \cap \left(\Lambda \times I^{n+1}\right). \tag{21.33}$$

$$\frac{\partial}{\partial t}\sigma + (\mathbf{u} \cdot \nabla)\sigma = 0$$

It ends upon setting

$$\sigma^{n+\frac{1}{2}} := \sigma(t^{n+1}) \qquad \text{in} \quad \Omega^{n+\frac{1}{2}},$$

in addition to the values for $\phi^{n+\frac{1}{2}}$ and $\mathbf{u}^{n+\frac{1}{2}}$. As for the velocity, the method of characteristics transports each component of the symmetric tensor (namely six fields when $d = 3$ and three fields when $d = 2$). Their values are obtained from the characteristics lines as in (21.15).

After space discretization, and using the notations introduced in Section 4, the prediction $\Sigma_M^{n+\frac{1}{2}} \in (\mathbb{V}^S)^{\frac{d(d+1)}{2}}$ of $\sigma(t^{n+1})$ is given at each cell center $\mathbf{x}_i$ by

$$\Sigma_M^{n+\frac{1}{2}}(\mathbf{x}_i) := \sum_{C \in \mathscr{F}^S} \Sigma_M^n(\text{center}(C))|C_i \cap \widetilde{C}|;$$

compare with (21.24).

At the end of the prediction step, the projection of the tensor $\Sigma_M^{n+\frac{1}{2}}$ into the finite element space into the finite element space $\mathbb{V}^{FEM}(\tau_K^{n+1})^{\frac{d(d+1)}{2}}$ is computed according to a formula similar to (21.27): i.e., for every $\mathbf{v} \in \mathscr{V}(\tau_K^{n+\frac{1}{2}})$

$$\Sigma_K^{n+\frac{1}{2}}(\mathbf{v}) := \frac{(\pi_{S \to FEM}(\Phi_M^{n+\frac{1}{2}} \Sigma_M^{n+\frac{1}{2}}))(\mathbf{v})}{\Phi_K^{n+\frac{1}{2}}(\mathbf{v})} \tag{21.34}$$

if $\mathbf{v}$ is not a vertex at the inflow boundary, and $\Sigma_K^{n+\frac{1}{2}}(\mathbf{v}) = \mathbf{G}(\mathbf{v})$ otherwise.

5.2.2 The Correction Step

After incorporation of the extra-stress related terms, the correction step reads as follows:

$$
\begin{cases}
\rho \dfrac{\partial}{\partial t}\mathbf{u} - \nabla \cdot (2\mu \mathbf{D}(\mathbf{u})) + \nabla p - \nabla \cdot \sigma = \mathbf{f} \\[2mm]
\nabla \cdot \mathbf{u} = 0 & \text{in} \quad \Omega^{n+1} \times I^{n+1}, \quad (21.35) \\[2mm]
\sigma + \lambda \left(\dfrac{\partial}{\partial t}\sigma - \nabla \mathbf{u}\,\sigma - \sigma \nabla \mathbf{u}^t \right) = 2\mu_p \mathbf{D}(\mathbf{u})
\end{cases}
$$

supplemented by the appropriate boundary conditions and free interface conditions.

As in Section 4, the volume fraction and liquid domain remain unchanged during the correction step. Problem (21.35) allows to obtain a correction of the velocity $\mathbf{u}$, the extra-stress tensor σ, and the pressure p. This correction step consists in two sub-steps decoupling the velocity-pressure corrections and the extra-stress correction. The first sub-step consists of solving a Stokes problem of the (21.28) type with a modified functional $L^{n+1}(.)$ accounting for the extra-stress tensor term:

$$
L^{n+1}(\mathbf{V}) := \frac{\rho}{\delta t^{n+1}} \int_{\Omega_K^{n+1}} \mathbf{U}_K^n \cdot \mathbf{V}\,d\mathbf{x} + \int_{\Omega_K^{n+1}} \mathbf{f}(t^{n+1}) \cdot \mathbf{V}\,d\mathbf{x} - \int_{\Omega_K^{n+1}} \Sigma_K^{n+\frac{1}{2}} :: \mathbf{D}(\mathbf{V})\,d\mathbf{x}.
$$

This corresponds to an explicit treatment of the visco-elastic effect $\Sigma_K^{n+\frac{1}{2}}$ in the first equation in (21.35). We then solve the third relation of (21.35) to update the extra-stress tensor $\Sigma_K^{n+\frac{1}{2}}$. The time discretization considered consists of an explicit treatment of the nonlinear terms, while continuous piecewise linear finite elements are used for the space discretization: Seek $\Sigma_K^{n+1} \in \left(\mathbb{V}^{FEM}(\tau_K^{n+1}) \right)^{\frac{d(d+1)}{2}}$, the subspace of $\left(\mathbb{V}^{FEM}(\tau_K^{n+1}) \right)^{d \times d}$ consisting in those symmetric matrices, satisfying

$$
\int_{\Omega_K^{n+1}} \left(\delta t^{n+1} \Sigma_K^{n+1} + \lambda \Sigma_K^{n+1} \right) :: \Theta\,d\mathbf{x}
$$

$$
= \int_{\Omega_K^{n+1}} \left(\Sigma_K^{n+\frac{1}{2}} + \delta t^{n+1} \Sigma_K^{n+\frac{1}{2}} \right.
$$

$$
\left. + \delta t^{n+1} \nabla \mathbf{U}_K^{n+1}\, \Sigma_K^{n+\frac{1}{2}} + \delta t^{n+1} \Sigma_K^{n+\frac{1}{2}} (\nabla \mathbf{U}_K^{n+1})^t \right) :: \Theta\,d\mathbf{x}
$$

$$
+ 2\mu_p \delta t^{n+1} \int_{\Omega_K^{n+1}} \mathbf{D}(\mathbf{U}_K^{n+1}) :: \Theta\,d\mathbf{x}, \qquad \forall \Theta \in \mathbb{V}_M^{FEM}(\tau_K^{n+1})^{\frac{d(d+1)}{2}}.
$$

In addition, the Elastic Viscous Stress Splitting (EVSS) stabilization procedure could be activated to compensate for possible small viscosities μ. Details can be found in [6].

5.3 Numerical Results for Visco-Elastic Flows

We first consider again the extrusion with initial contraction experiment presented in Section 4.3.1. The goal of this section is to discuss the visco-elastic influence, via the presence of the extra-stress σ_{33}.

5.3.1 Extrusion with Die Swell and Contraction

Let us consider a visco-elastic fluid, which has density $\rho = 1300$ kg m^{-3}, and viscosity $\mu = 0$. Its relaxation time is $\lambda = 0.1$ s and the polymer viscosity is $\mu_p = 10$ kg(ms)$^{-1}$. The boundary condition at the inflow boundary is a Poiseuille flow with velocity given by $\mathbf{u} = (0, 0, u_z)$ and $u_z(r) = -100(r^2 - 0.01^2)$ ms^{-1} (where r is the radial distance to the central axis of the die). Slip boundary conditions are imposed on the lateral sides of $\partial \Lambda$ and no-slip boundary conditions are applied on the bottom plate. Gravity forces, $\mathbf{g} = -9.81\mathbf{e_3}$ ms^{-2} are oriented along the die. The time step is constant and equal to $\delta t = 0.005$ s.

Figure 21.18 provides the volume fraction Φ, extra-stress σ_{33}, and speed $|\mathbf{U}|$ fields in a median cut in the middle of the domain at various times. Figure 21.19 visualizes the buckling effect. Since slip boundary conditions are applied along the die, no die swell occurs after exiting the die. However, when the jet hits the wall we observe a different buckling behavior compared to the Newtonian case; compare Figures 21.13 and 21.19.

5.3.2 Influence of the Polymer Viscosity and Relaxation Time

The influence of the polymer viscosity μ_p and relaxation time λ is now investigated, keeping slip boundary conditions along the die. Figures 21.20 and 21.21 represent the volume fraction Φ, extra-stress σ_{33}, and speed $|\mathbf{U}|$ fields in a median cut in the middle of the domain at various times, as well as the buckling effect of the liquid domain, for a relaxation time $\lambda = 1$ s. and a polymer viscosity $\mu_p = 10$ kg(ms)$^{-1}$ (larger relaxation time compared to the simulations in Section 5.3.1). Figures 21.20 and 21.21 illustrate the same quantities, for a relaxation time $\lambda = 0.1$ s. and a polymer viscosity $\mu_p = 100$ kg(ms)$^{-1}$ (larger viscosity compared to the simulations in Section 5.3.1). Clearly, both polymer viscosity μ_p and the relaxation time λ have a significant influence on the jet shape during buckling.

5.3.3 Influence of Boundary Conditions

In this section, we discuss the influence of the boundary conditions (typically slip vs. no-slip boundary conditions) on the die boundary, on the buckling phenomena, and on the visco-elastic fluid behavior. The polymer viscosity and relaxation time are kept as in Figures 21.20–21.23, $\mu_p = 100$ kg(ms)$^{-1}$, $\lambda = 0.1$ s, whereas no-slip
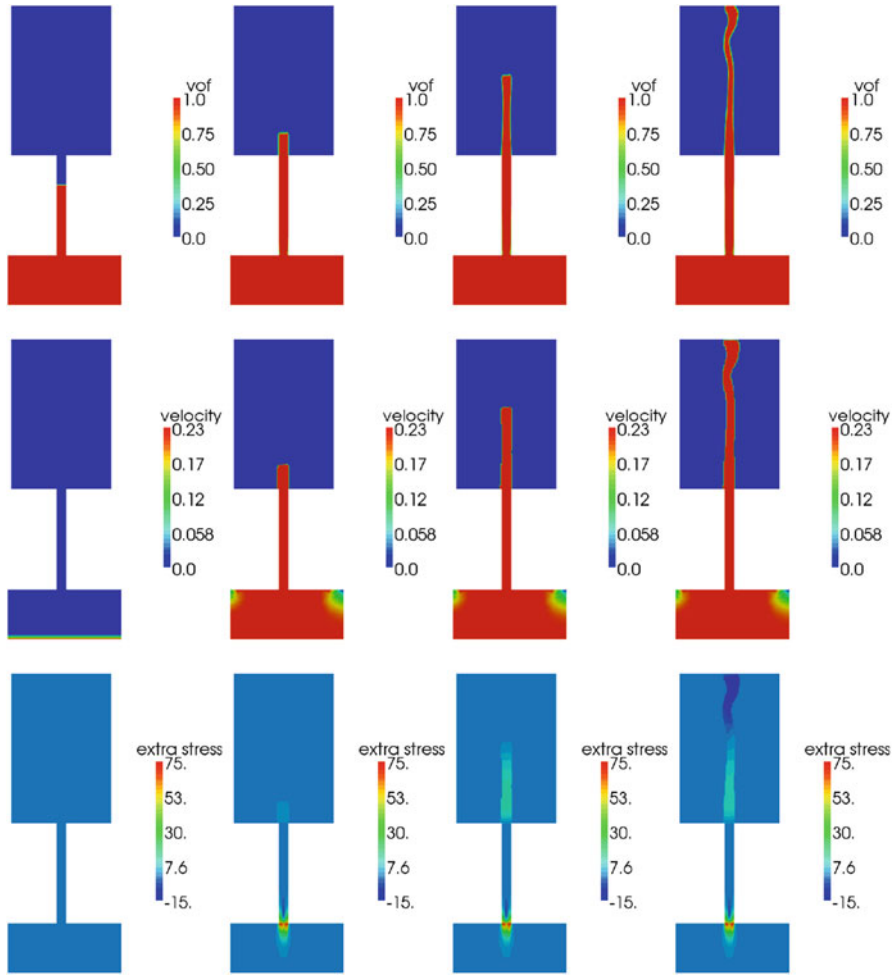
Fig. 21.18 Die swell with extrusion of a visco-elastic fluid, with slip boundary conditions in the die ($\mu_p = 10$ kg(ms)$^{-1}$, $\lambda = 0.1$ s.). Snapshots of the solution at times $t = 0, 0.4, 0.6$, and 0.8 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ; middle: speed $|\mathbf{U}|$; bottom: representation of extra-stress σ_{33}.

boundary conditions now apply along the die. The shape of the jet is significantly different as shown in Figures 21.24 and 21.25. The die swell is significant, therefore we decrease the value of the relaxation time to $\lambda = 0.005$ s, still keeping the same polymer viscosity $\mu_p = 100$ kg(ms)$^{-1}$. The swelling of the die is now much smaller as shown in Figures 21.26 and 21.27. Figures 21.28 and 21.29 illustrate the same quantities, for a smaller relaxation time, $\lambda = 0.002$ s, still keeping the same polymer viscosity $\mu_p = 100$ kg(ms)$^{-1}$. We therefore conclude that the type of boundary conditions applied along the die has a significant impact on the extrusion process.
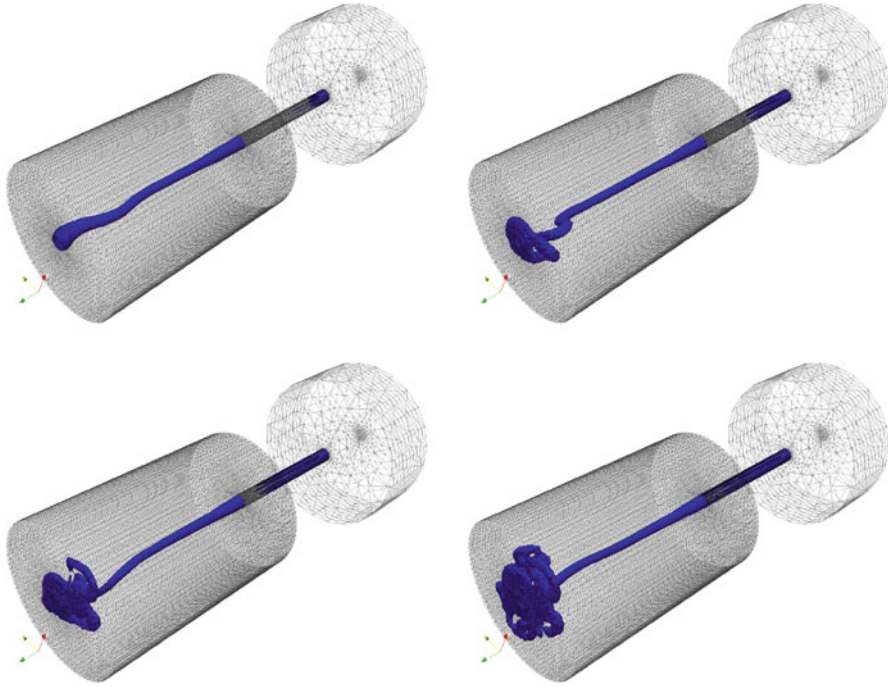
Fig. 21.19 Die swell with extrusion of a visco-elastic fluid, with slip boundary conditions in the die ($\mu_p = 10$ kg(ms)$^{-1}$, $\lambda = 0.1$ s.). Snapshots of the solution at times $t = 0.8, 1.0, 1.2$, and 1.6 s. Representation of the liquid domain and buckling effect.

5.3.4 Bended Die

Finally, to conclude the discussion on viscoelastic effects, we study quantitatively the influence of the bend of the die on the extrusion. In particular, the distribution of the extra-stress and the differences of amplitude are fundamental in industrial processes, as they induce a different behavior of the visco-elastic material at the exit of the die, and thus a different final production.

We consider a geometry that is similar to the one before, with the addition of a ninety degree angle bend in the die. All the other geometrical dimensions remain the same. Figure 21.30 illustrates the geometry together with the associated finite element mesh.

As before, the visco-elastic fluid has density $\rho = 1300$ kg m^{-3}, and viscosity $\mu = 0$. Its relaxation time is $\lambda = 0.1$ s and the polymer viscosity is $\mu_p = 10$ kg(ms)$^{-1}$. The boundary condition at the inflow boundary is a Poiseuille flow, slip boundary conditions are imposed along the die, as well as at the exit of the domain. Gravity forces with amplitude $|\mathbf{g}| = 9.81$ ms^{-2} are oriented along the inflow. The time step is constant and equal to $\delta t = 0.005$ s.
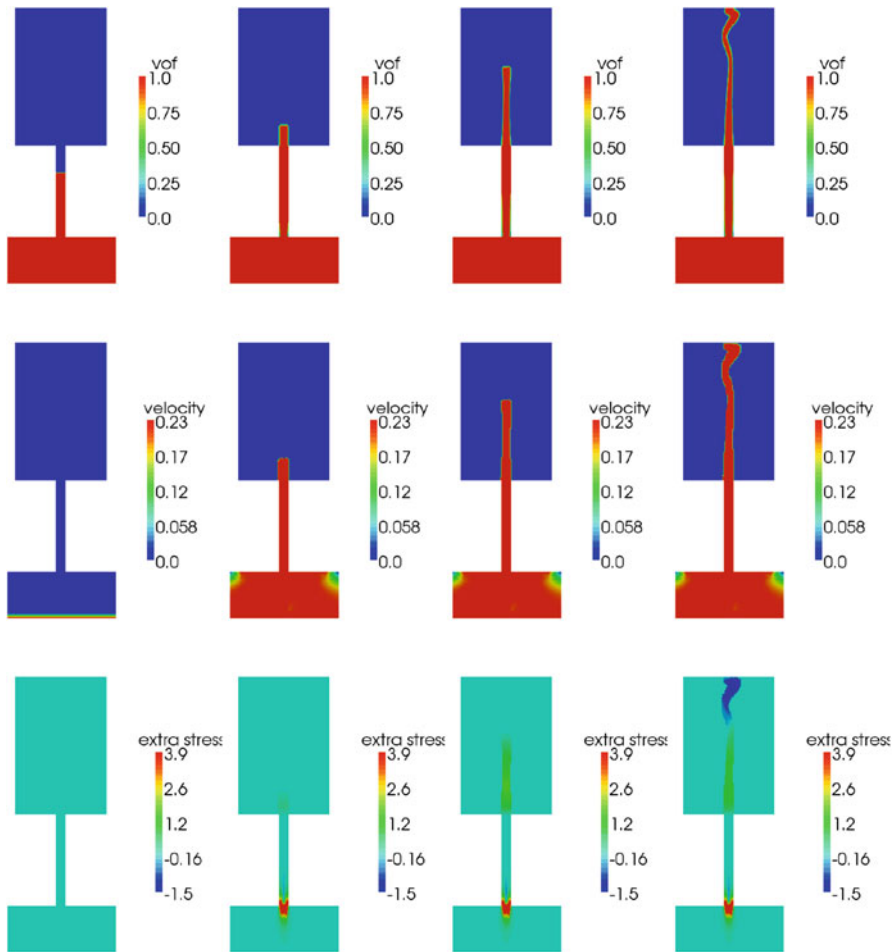
Fig. 21.20 Die swell with extrusion of a visco-elastic fluid, with slip boundary conditions in the die ($\mu_p = 10$ kg(ms)$^{-1}$, $\lambda = 1$ s.). Snapshots of the solution at times $t = 0, 0.4, 0.6$, and 0.8 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ; middle: speed $|\mathbf{U}|$; bottom: representation of extra-stress σ_{33}.

Figure 21.31 illustrates representations of the volume fraction Φ, speed $|\mathbf{U}|$, and extra-stress σ_{33} fields in a median cut in the middle of the domain at various instants of time. Figure 21.32 illustrates snapshots of the liquid domain. These results should be compared with those of Figures 21.18 and 21.19 which correspond to a straight die. One can observe a significant buckling effect as the liquid is switching directions. This behavior is caused by the variation of the extra stress inside the curved die. Figure 21.33 illustrates snapshots of the liquid domain in the same situation but for a relaxation time $\lambda = 0.02$ s. One can observe larger effects due to the shorter

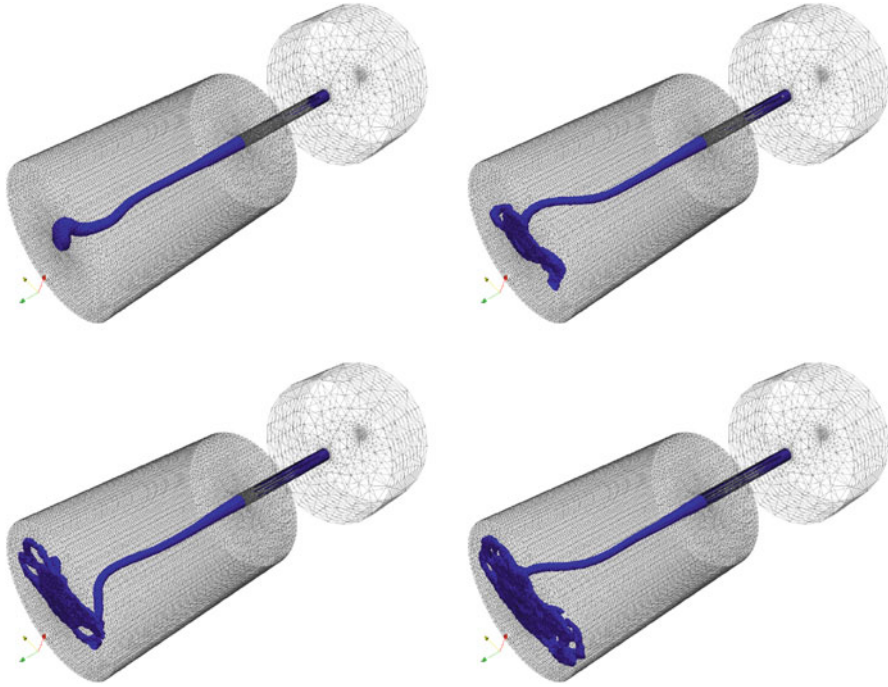**Fig. 21.21** Die swell with extrusion of a visco-elastic fluid, with slip boundary conditions in the die ($\mu_p = 10$ kg(ms)$^{-1}$, $\lambda = 1$ s.). Snapshots of the solution at times $t = 0.8, 1.0, 1.2$, and 1.6 s. Representation of the liquid domain and buckling effect.

relaxation time. We therefore conclude that memory effects due to the shape of the cavity before the die may strongly affect the shape of the jet after the die. This observation can be very important in industrial applications such as pasta processing for instance.

6 Multiphase Flows with Free Surfaces

6.1 Mathematical Modeling of Multiphase Flows with Free Surfaces

We extend here the previous model to the case of multiple liquid phases with a free surface. More precisely, we consider P liquid phases, and the ambient gas is the phase numbered $P + 1$. We assume that the liquid phases are incompressible and immiscible, and thus rely on the *density-dependent Navier-Stokes equations* for the
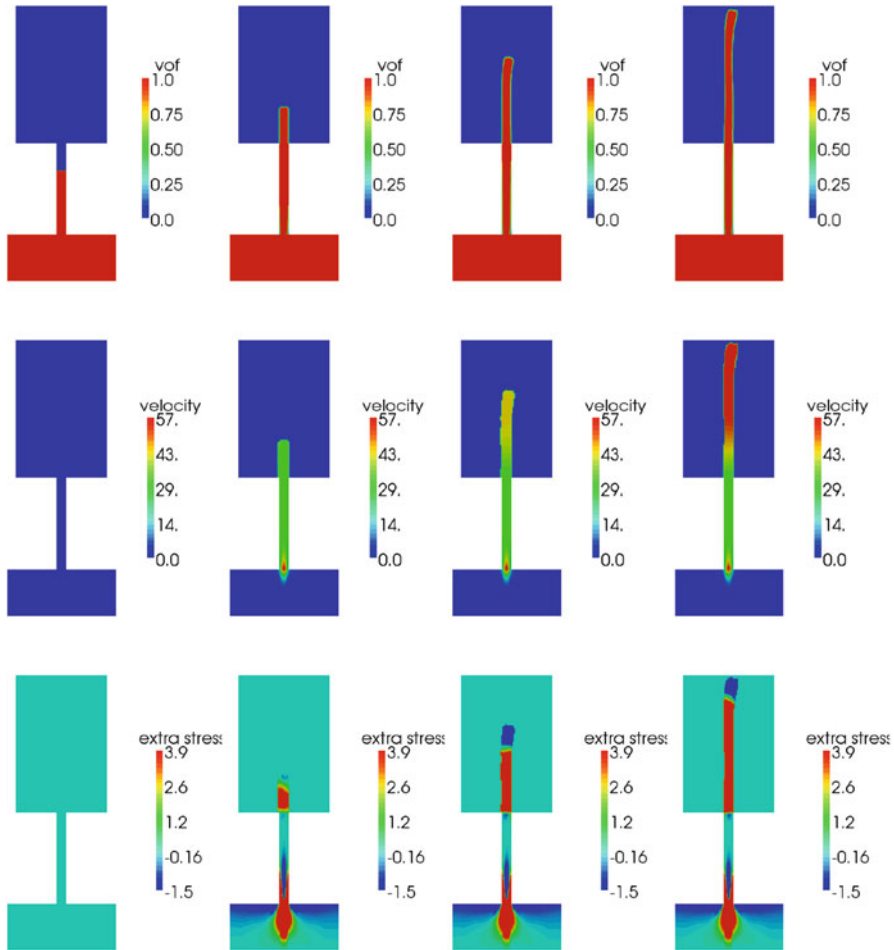
Fig. 21.22 Die swell with extrusion of a visco-elastic fluid, with slip boundary conditions in the die ($\mu_p = 100$ kg(ms)$^{-1}$, $\lambda = 0.1$ s.). Snapshots of the solution at times $t = 0, 0.4, 0.6$, and 0.8 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ; middle: speed $|\mathbf{U}|$; bottom: representation of extra-stress σ_{33}.

modeling of the global set of liquid phases. This model is based on [10, 19]. In [19], the emphasis has been put on the simulation of landslide-generated impulse waves. Here we show that the applications are numerous and that our algorithm can apply at different time and space scales.

We denote by $\Omega_\ell(t) \subset \Lambda$, $\ell = 1, \ldots, P$, the domain occupied by the ℓth liquid phase at time $t \in [0, T]$ and by $\Omega(t) = \bigcup_{\ell=1}^{P} \Omega_\ell(t)$ the global liquid domain. The subdomain $\Omega_\ell(t)$ is defined by its characteristic function $\phi_\ell : \Lambda \times [0, T] \rightarrow \{0, 1\}$:

$$\Omega_\ell(t) = \{\mathbf{x} \in \Lambda \mid \phi_\ell(\mathbf{x}, t) = 1\}, \quad \ell = 1, \ldots, P. \tag{21.36}$$

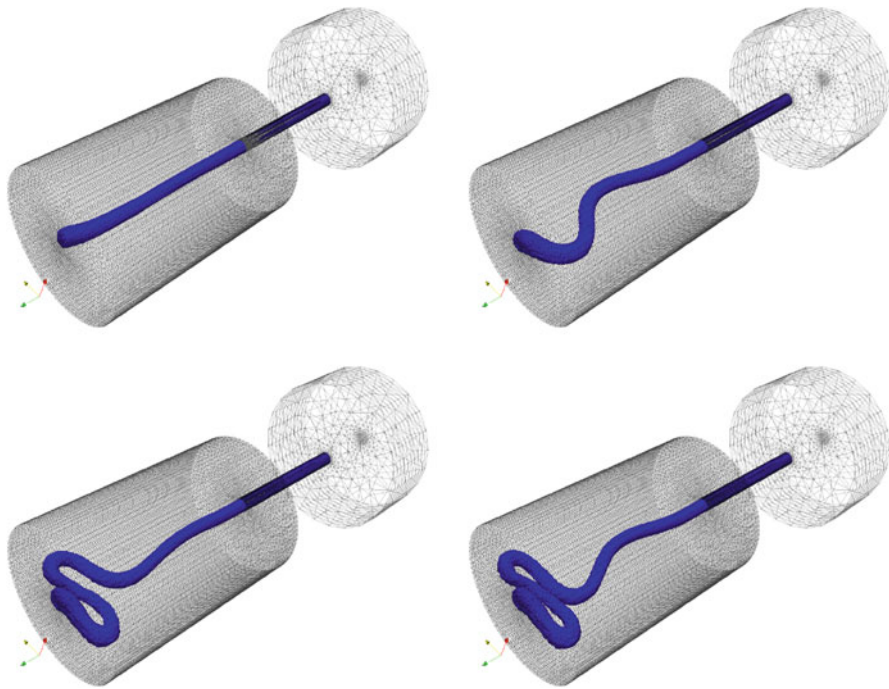As a consequence, $\phi := \sum_{\ell=1}^{P} \phi_\ell$ is the characteristic function of $\Omega(t)$.

Fig. 21.23 Die swell with extrusion of a visco-elastic fluid, with slip boundary conditions in the die ($\mu_p = 100$ kg(ms)$^{-1}$, $\lambda = 0.1$ s.). Snapshots of the solution at times $t = 0.8, 1.0, 1.2$, and 1.3 s. Representation of the liquid domain and buckling effect.

All phases being considered to be Newtonian, incompressible, and immiscible, the Navier-Stokes equations are satisfied for each of them, with physical properties such as density and viscosity varying from one liquid phase to the other. We denote by ρ_l and μ_l, $l = 1, \ldots, P$, the respective densities and viscosities. In this setting the velocity and pressure are related through the Navier-Stokes relations (21.1), where the mass density is recovered as $\rho := \sum_{\ell=1}^{P} \phi_\ell \, \rho_\ell$, and similarly for the viscosity $\mu := \sum_{\ell=1}^{P} \phi_\ell \, \mu_\ell$.

The boundary conditions, interface conditions, and initial conditions are imposed in a similar way as in the single liquid phase case. At the interfaces between liquid phases, natural continuity conditions are imposed so that no forces are applied. Figure 21.34 illustrates a 2D sketch of multiple liquid phases in the case of die swell extrusion.

The evolution of each domain $\Omega_\ell(t)$ is governed by the transport of its characteristic function with the fluid velocity, that is:

$$\frac{\partial}{\partial t}\phi_\ell + \mathbf{u} \cdot \nabla \phi_\ell = 0 \quad \text{in} \quad Q_\ell, \qquad \phi_\ell = 0 \quad \text{in} \quad \Lambda \setminus Q_\ell, \tag{21.37}$$

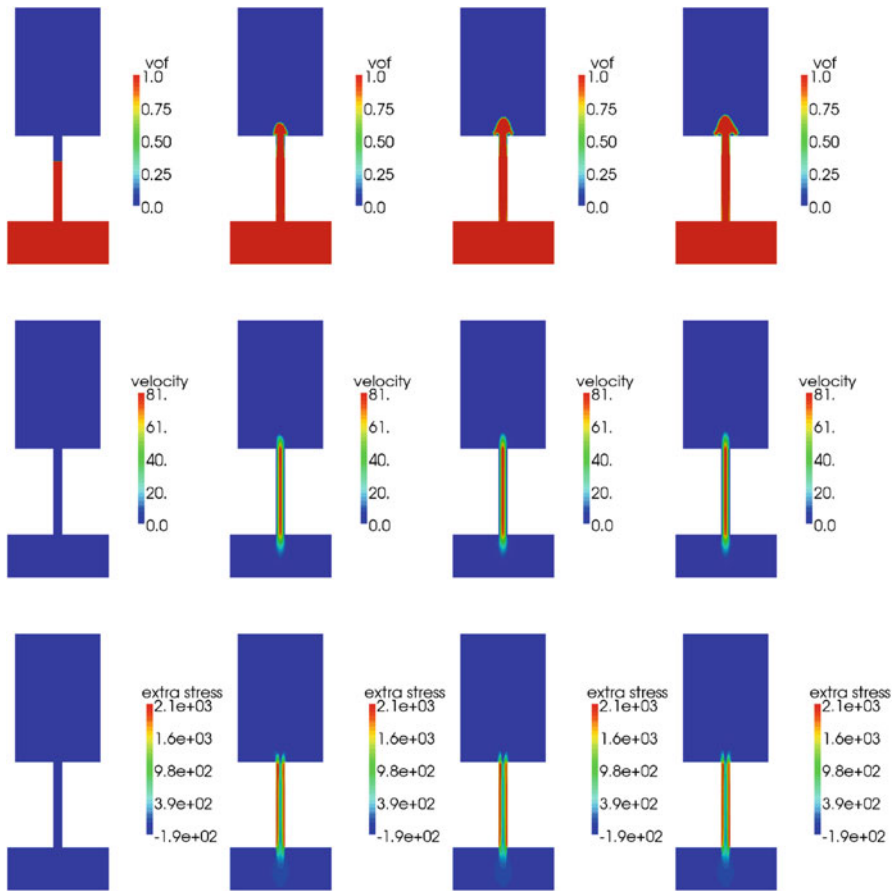**Fig. 21.24** Die swell with extrusion of a visco-elastic fluid, with no-slip boundary conditions in the die ($\mu_p = 100$ kg(ms)$^{-1}$, $\lambda = 0.1$ s.). Snapshots of the solution at times $t = 0, 0.4, 0.6$, and 0.8 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ; middle: speed $|\mathbf{U}|$; bottom: representation of extra-stress σ_{33}.

where

$$Q_\ell := \{(\mathbf{x}, t) \in \Lambda \times (0, T] \mid \mathbf{x} \in \Omega_\ell(t)\}$$

and where $\mathbf{u}$ is the fluid velocity only defined on the space-time fluid domain Q.

The inflow boundary conditions supplementing the equations (21.37) have to be imposed for each liquid phase on the boundary of Λ, the same way it is imposed for one liquid phase. The initial value of the characteristic functions ϕ_ℓ are chosen to match the initial given domains $\Omega_\ell(0)$,

$$\phi_\ell(.,0) = 1 \quad \text{on} \quad \Omega_\ell(0) \quad \text{and} \quad \phi_\ell(.,0) = 0 \quad \text{otherwise.} \quad (21.38)$$
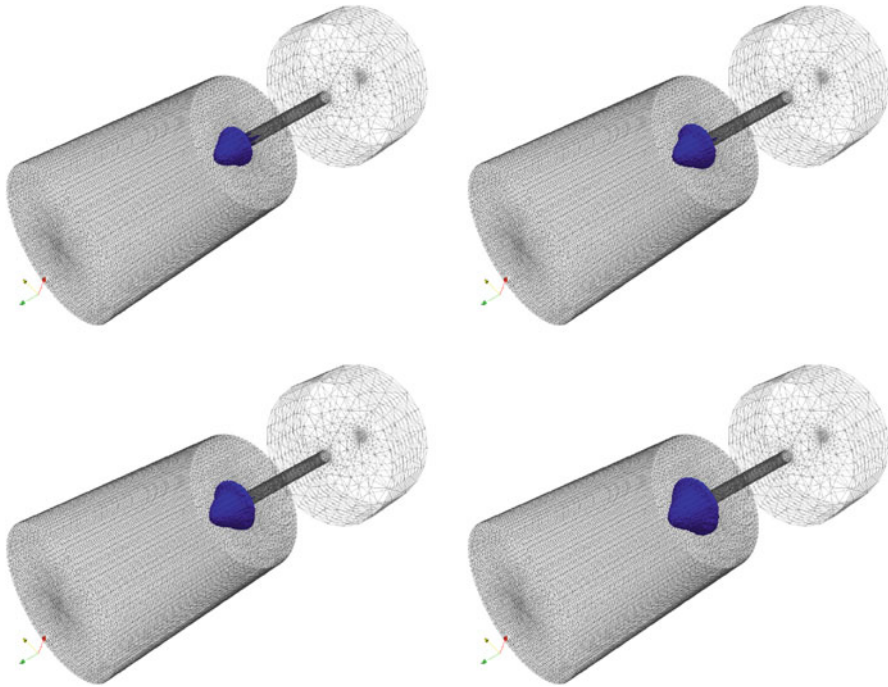
Fig. 21.25 Die swell with extrusion of a visco-elastic fluid, with no-slip boundary conditions in the die ($\mu_p = 100$ kg(ms)$^{-1}$, $\lambda = 0.1$ s.). Snapshots of the solution at times $t = 0.8, 1.0, 1.2,$ and 1.6 s. Representation of the liquid domain and buckling effect.

6.2 Extension of the Operator Splitting Strategy

The extension of the operator splitting method to multiphase flows includes mainly the transport of multiple volume fractions, which is a natural extension of the transport of a single phase. However a significant step of the algorithm involves the reconstruction of the interfaces and the numerical methods to avoid artificial diffusion and compression, which have to be re-designed in the context of multiphase flows.

The operator splitting algorithm to approximate the system of equations (21.1) and (21.37) again decouples the approximation of the diffusion and advection operators. In this case, the diffusion operators correspond to a Stokes problem on a stationary domain with piecewise constant density and viscosity fields. The advection operator includes the transport equations for the Navier part of the incompressible fluid, as well as for the transport of the P characteristic functions.

The time splitting scheme reads as follows. We assume to be given an approximation of the liquid domain characteristic functions ϕ_ℓ^n, $\ell = 1, \ldots, P$ at time t^n. This entails an approximation of the liquid domains Ω_ℓ^n and of the global liquid domain Ω^n via the relations:
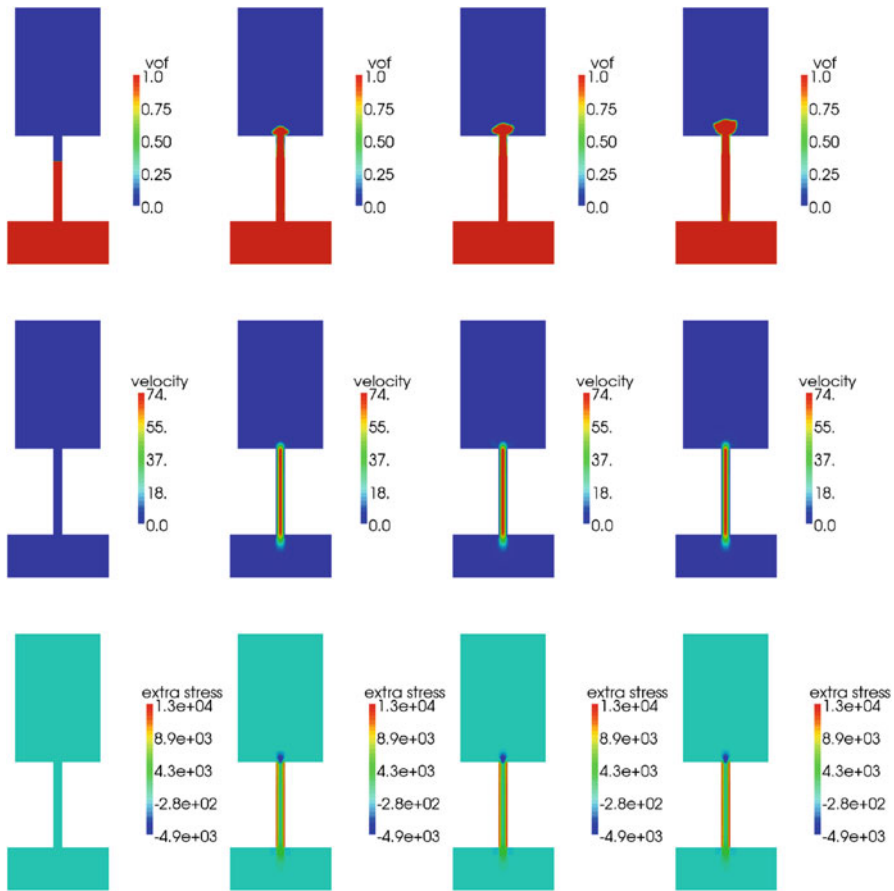
Fig. 21.26 Die swell with extrusion of a visco-elastic fluid, with no-slip boundary conditions in the die ($\mu_p = 100$ kg(ms)$^{-1}$, $\lambda = 0.005$ s.). Snapshots of the solution at times $t = 0, 0.4, 0.6$, and 0.8 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ; middle: speed $|\mathbf{U}|$; bottom: representation of extra-stress σ_{33}.

$$\Omega_\ell^n := \{\mathbf{x} \in \Lambda \mid \phi_\ell^n(\mathbf{x}) = 1\}, \quad \Omega^n := \bigcup_{\ell=1}^{P} \Omega_\ell^n.$$

We also assume to be given a velocity approximation $\mathbf{u}^n(\mathbf{x})$ of $\mathbf{u}(\mathbf{x}, t^n)$. The prediction step determines the new approximation of the liquid domain at time t^{n+1}, together with a prediction of the velocity on the new domain. The correction step provides an update of the velocity and pressure while the liquid domain remains unchanged.
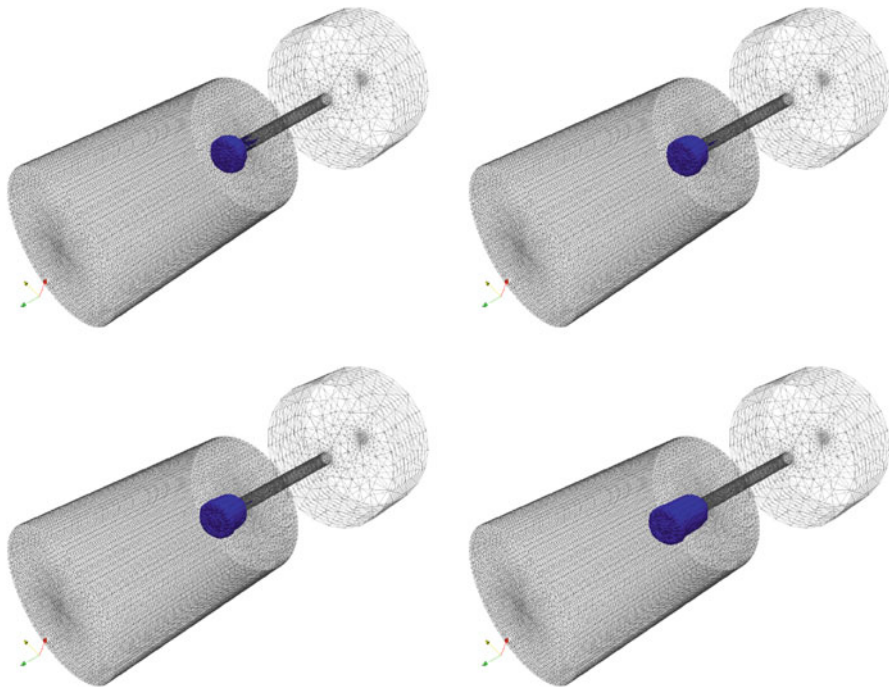
Fig. 21.27 Die swell with extrusion of a visco-elastic fluid, with no-slip boundary conditions in the die ($\mu_p = 100$ kg(ms)$^{-1}$, $\lambda = 0.005$ s.). Snapshots of the solution at times $t = 0.8, 1.0, 1.2$, and 1.6 s. Representation of the liquid domain and buckling effect.

6.2.1 The Prediction Step

The *projection step* encompasses the advection components of (21.1) and (21.37). It consists in solving the $P + 1$ transport equations:

$$\frac{\partial}{\partial t}\phi_\ell + \mathbf{u} \cdot \nabla\phi_\ell = 0, \quad \ell = 1,\dots,P,$$

$$\frac{\partial}{\partial t}\mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} = \mathbf{0} \tag{21.39}$$

in $Q^{n+1} := Q \cap (\Lambda \times I^{n+1})$. Outside the liquid domain Ω_ℓ^n, we set $\phi_\ell(\mathbf{x}, t) = 0$ whenever $\mathbf{x} \in \Lambda \setminus \{\mathbf{y}(t; \bar{\mathbf{x}}) \mid \bar{\mathbf{x}} \in \Omega_\ell^n\}$ and $\mathbf{u}$ is not required outside Ω^n. Eventually, we end up setting $\phi_\ell^{n+1} := \phi_\ell(t^{n+1})$ in Λ, and consequently

$$\Omega_\ell^{n+1} := \{\mathbf{x} \in \Lambda \mid \phi_\ell^{n+1}(\mathbf{x}) = 1\}, \quad \Omega^{n+1} = \bigcup_{\ell=1}^{P} \Omega_\ell^{n+1}, \tag{21.40}$$
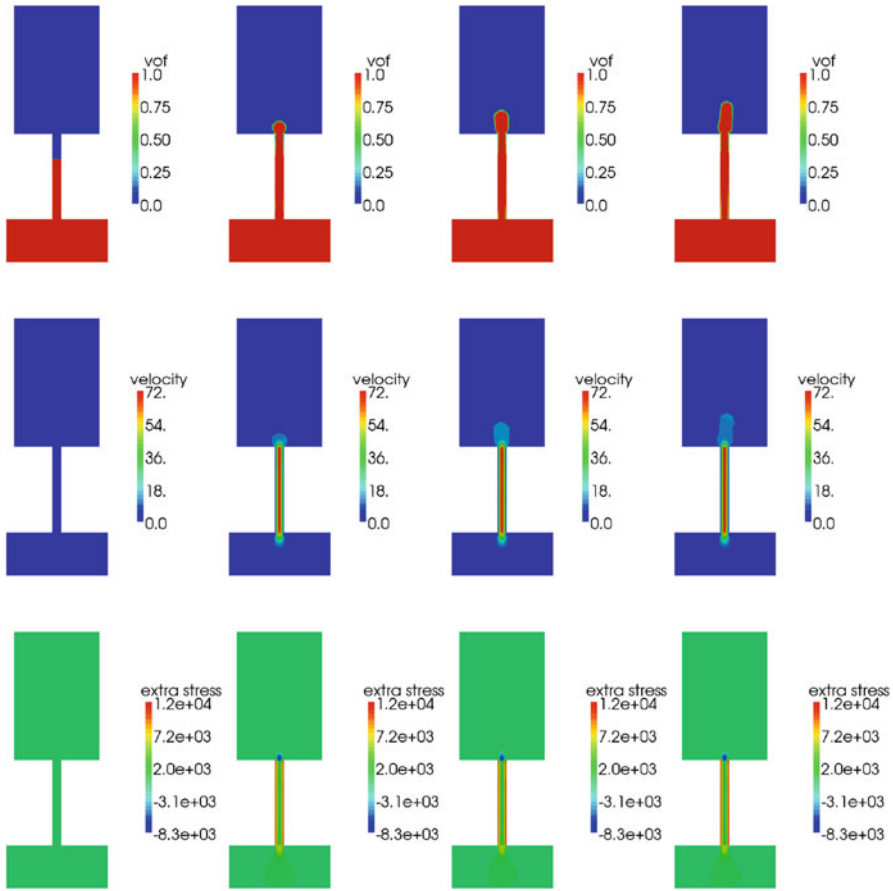
Fig. 21.28 Die swell with extrusion of a visco-elastic fluid, with no-slip boundary conditions in the die ($\mu_p = 100$ kg(ms)$^{-1}$, $\lambda = 0.002$ s.). Snapshots of the solution at times $t = 0, 0.4, 0.6$, and 0.8 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ; middle: speed $|\mathbf{U}|$; bottom: representation of extra-stress σ_{33}.

as well as $\mathbf{u}^{n+\frac{1}{2}} := \mathbf{u}(t^{n+1})$ in Ω^{n+1}. At the continuous level, these problems are highly similar to those encountered for one single phase. However, after space discretization, the complexity is quite different.

Indeed, the prediction steps starts with given approximations $\Phi^n_{\ell,M} \in \mathbb{V}^S$, $\ell = 1, \ldots, P$, and $\mathbf{U}^n_M \in (\mathbb{V}^S)^d$ of the liquid fractions and velocity respectively. The predictions $\Phi^{n+\frac{1}{2}}_{\ell,M}$ and $\mathbf{U}^{n+\frac{1}{2}}_M$, in $\mathbb{V}^S$ and $(\mathbb{V}^S)^d$ respectively, are computed as in (21.24) by transport of the quantities in each cell C and projection on the grid $\mathscr{P}$. Notice that each of the transport equations for $\Phi^n_{\ell,M}$ are solved in parallel for each liquid phase, and the redistribution is achieved sequentially. Details can be found in [19].
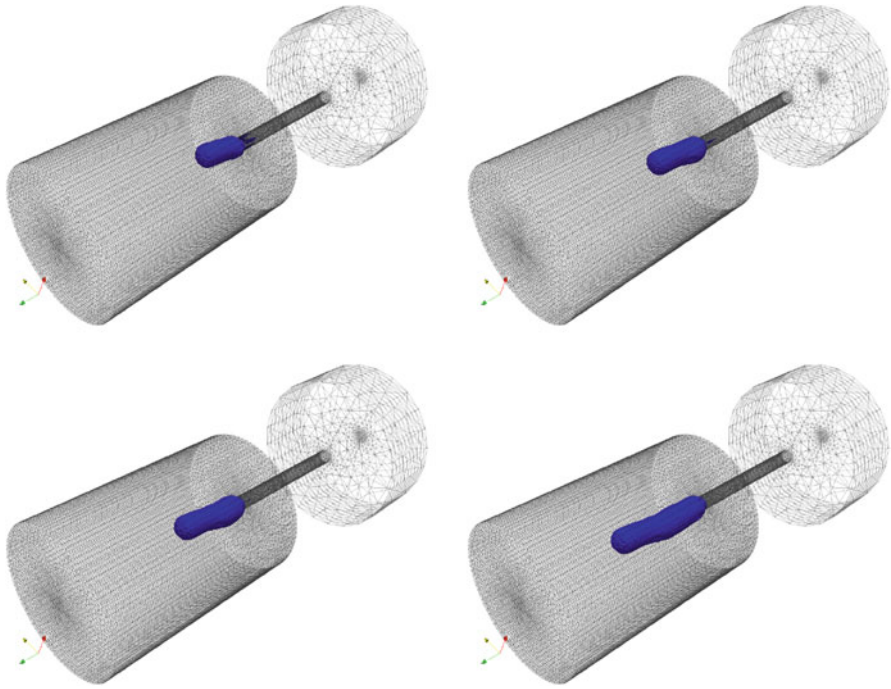
Fig. 21.29 Die swell with extrusion of a visco-elastic fluid, with no-slip boundary conditions in the die ($\mu_p = 100$ kg(ms)$^{-1}$, $\lambda = 0.002$ s.). Snapshots of the solution at times $t = 0.8, 1.0, 1.2$, and 1.6 s. Representation of the liquid domain and buckling effect.
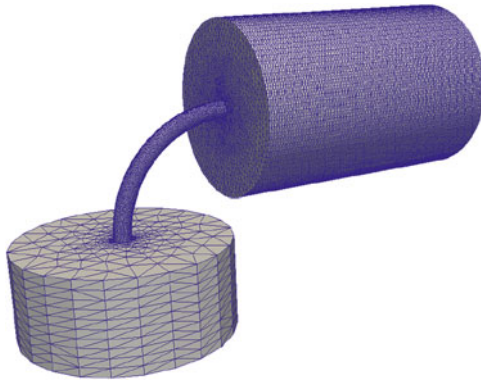

Fig. 21.30 Die swell with a bend, for the extrusion of a visco-elastic fluid. Visualization of the geometrical domain and finite element mesh.
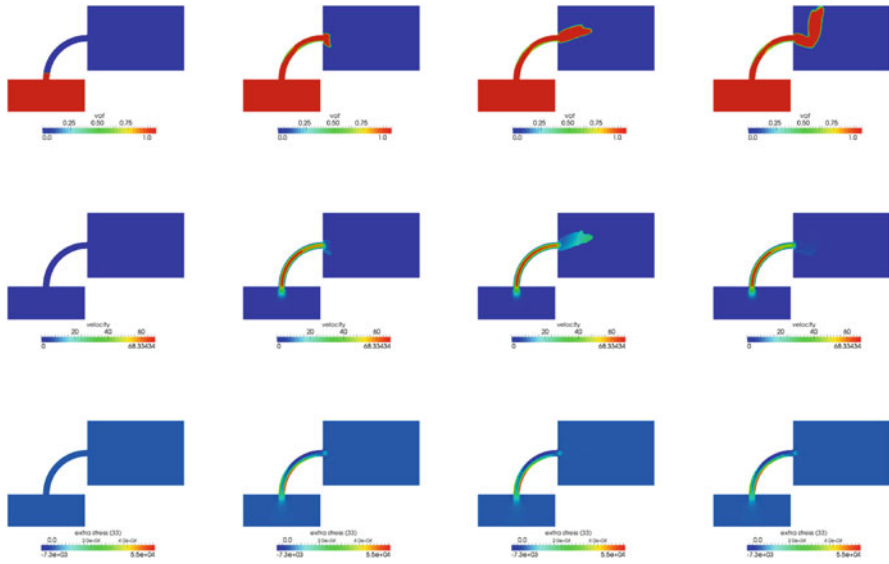
Fig. 21.31 Die swell of a visco-elastic fluid for a die with a 90 degrees bend. Snapshots of the solution at times $t = 0, 0.4, 0.8$, and 1.2 s (left to right). Top: volume fraction of liquid Φ in a median cut in the middle of the domain; middle: speed $|\mathbf{U}|$ in a median cut in the middle of the domain; bottom: Extra-stress field σ_{33} in a median cut in the middle of the domain.

Fig. 21.32 Die swell of a visco-elastic fluid for a die with a 90 degrees bend. Snapshots of the solution at times $t = 0, 0.4, 0.8$, and 1.2 s (left to right).

Fig. 21.33 Die swell of a visco-elastic fluid for a die with a 90 degrees bend. Snapshots of the solution at times $t = 0, 0.4, 0.8$, and 1.2 s (left to right) for $\lambda = 0.02$ s.
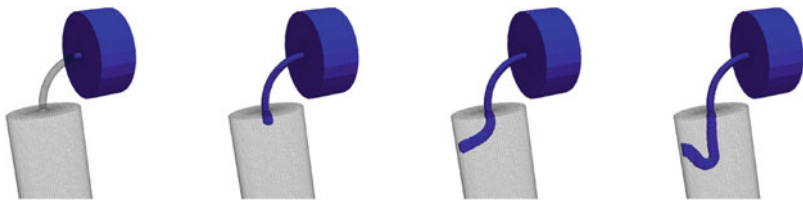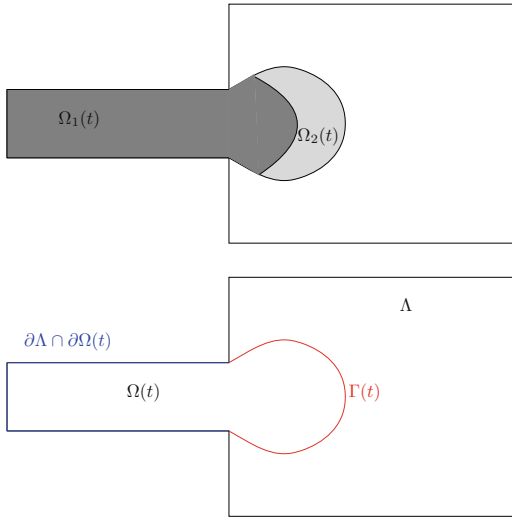
Fig. 21.34 Die swell with two liquid phases (phase 1 pushing phase 2). Geometrical notation for the VOF formulation for two liquid phases with free surface included into the computational domain Λ.

6.2.2 Numerical Diffusion vs Numerical Compression

As in the single phase case, the advected fields $\Phi_{\ell,M}^{n+\frac{1}{2}}$ (and as a matter of fact $\Phi_M^{n+\frac{1}{2}}$ as the sum of all liquid fractions) do not necessarily have values that are exactly zero or one. To cope with this numerical diffusion and compression, we use *multiphase* versions of SLIC and decompression algorithms [12, 26, 28].

The multiphase version of the SLIC algorithm consists of a sequential use of the SLIC algorithm for each liquid phase. It is illustrated in Figure 21.35 (middle and right, for one liquid phase or two liquid phases). Each of the liquid phases is pushed against the sides/corners of the cell to be transported. The transport and projection of the cell are then made for each phase independently and sequentially. Thus the numerical diffusion can be reduced for each phase in parallel. More precisely, on the example illustrated in Figure 21.35 (right), the advected quantity of the first liquid phase lies in one cell only, thus no numerical diffusion is introduced for that particular phase. The advected quantity for the second liquid phase is redistributed over two cells, which means that some diffusion is introduced but limited over two cells instead of four.

Remark 3 (SLIC vs PLIC). The SLIC procedure has been preferred for instance over the higher order PLIC procedure for its handling simplicity within the two-grid framework. The rationale behind this approach is to use a low order interface reconstruction technique, like SLIC, *on a very fine mesh*. The mesh size guarantees the accuracy of the algorithm and compensates for the low order of the reconstruction technique. Replacing the SLIC algorithm with a PLIC algorithm on the structured
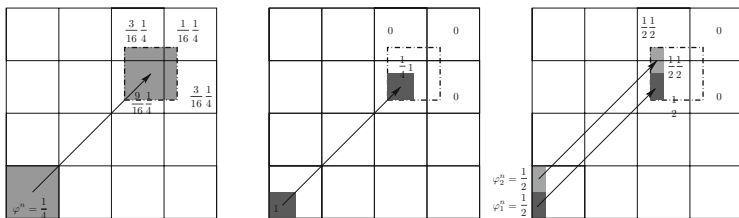
Fig. 21.35 An example of two dimensional advection and projection when the volume fraction of liquid in the cell is $\Phi_M^n = \frac{1}{4}$. Left: without SLIC and with one liquid phase, the volume fraction of liquid is advected and projected on four cells, with contributions (from the top left cell to the bottom right cell) $\frac{3}{16}\frac{1}{4}$, $\frac{1}{16}\frac{1}{4}$, $\frac{9}{16}\frac{1}{4}$, $\frac{3}{16}\frac{1}{4}$. Middle: with SLIC and with one liquid phase, the volume fraction of liquid is first pushed at one corner, then it is advected and projected on one cell only, with contribution $\frac{1}{4}$. Right: with SLIC and with two liquid phases, the volume fractions of liquid are first pushed along one side of the cell, then they are advected. The first liquid phase (corresponding to a volume of $\frac{1}{8}$) is projected on one cell only, with contribution $1\frac{1}{8}$; the second liquid phase (corresponding also to a volume of $\frac{1}{8}$) is projected on two cells, with contribution $1\frac{1}{16}$ and $1\frac{1}{16}$.

grid of small cells is not a fundamental problem, but a technical difficulty. Moreover, the PLIC procedure applied before transport of a cell, and coupled with a projection operator on the finite element mesh would be of little benefit, and expensive from the computational viewpoint.

Note that the sequential treatment of liquid phases implicitly requires them to be sorted; the arbitrary phase ordering influences the reconstruction of the interfaces, as already stated in [13] for three phases. However, numerical experiments show that the effect of the ordering of phases is not a crucial factor for the final results, especially at the limit when the mesh size tends to zero.

After the interface reconstruction and advection steps, it may happen that some cell C_i in the grid $\mathscr{P}^{\mathcal{S}}$ is over-filled, i.e., $\Phi_M^{n+\frac{1}{2}} = \sum_\ell \Phi_{\ell,M}^{n+\frac{1}{2}} > 1$. Such physically non-admissible values can indeed occur even if $\Phi_{\ell,M}^n \in [0,1]$ since the transport-and-project algorithm is not a divergence-free process.

We thus need to *decompress* the fields $\Phi_{\ell,M}^{n+\frac{1}{2}}$ and $\Phi_M^{n+\frac{1}{2}}$ with a numerical technique that allows to conserve the mass in a global sense [19, 26]. This algorithm is applied after the solution of the transport equations, but before the solution of the diffusion equations. It proceeds in two steps: first, we compute the excess of each liquid phase in each cell after advection and projection onto $\mathscr{P}^{\mathcal{S}}$; second, we redistribute these amounts *proportionally to the amount already included in the cell* in a given arbitrary order, in a way that is similar to *global repair* algorithms [35]. This method is robust, but requires to order the liquid phases (arbitrarily) to know which phase is redistributed first into the other cells. Numerical experiments have shown in our case that, besides guaranteeing the mass conservation globally, the error due to this decompression algorithm is reduced as the time step decreases. This heuristic algorithm can be found in [19] and is not detailed more extensively here.

This multiphase decompression algorithm is illustrated in Figure 21.36, when $\mathscr{P}^S$ is a single layer of cells, for the case of two liquid phases. The rebalancing principle that allows to conserve the mass in each phase at each time step is detailed in this pseudo 1D configuration, but can be extended in three space dimensions in a straightforward manner.

After the decompression, the approximations $\Phi_{\ell,M}^{n+\frac{1}{2}}$, $\ell = 1,\ldots,P$ and $\mathbf{U}_M^{n+\frac{1}{2}}$ are projected into the finite element spaces

$$\Phi_{\ell,K}^{n+1} := \pi_{S\to FEM}\Phi_{\ell,M}^{n+\frac{1}{2}} \in \mathbb{V}^{FEM}, \qquad \ell = 1,\ldots,P,$$

and $\mathbf{U}_K^{n+\frac{1}{2}} \in \mathscr{V}(\tau_K^{n+1})$ is defined according to formula (21.27) and where τ_K^{n+1} is the collection of liquid tetrahedra, see Section 4.2.

Φ^{n+½}_{1,M}   Φ^{n+½}_{2,M}

| 0.5 | 0.9 | 0.9 | 0.0 | 0.0 | | 0.0 | 0.0 | 0.8 | 0.8 | 0.4 |

$\Phi_M^{n+\frac{1}{2}} := \Phi_{1,M}^{n+\frac{1}{2}} + \Phi_{2,M}^{n+\frac{1}{2}}$ before decompression

| 0.5 | 0.9 | 1.7 | 0.8 | 0.4 |

$\Phi_{1,M}^{n+\frac{1}{2}} - E_1$ Remove excess in overfilled cells $\Phi_{2,M}^{n+\frac{1}{2}} - E_2$
(excess $E_1 = 0.37$)   (excess $E_2 = 0.33$)

| 0.5 | 0.9 | 0.53 | 0.0 | 0.0 | | 0.0 | 0.0 | 0.47 | 0.8 | 0.4 |

Distribute excess into partially empty cells

Φ^{n+½}_{1,M}   Φ^{n+½}_{2,M}

| 0.77 | 1.0 | 0.53 | 0.0 | 0.0 | | 0.0 | 0.0 | 0.47 | 1.0 | 0.53 |

$\Phi_M^{n+\frac{1}{2}} := \Phi_{1,M}^{n+\frac{1}{2}} + \Phi_{2,M}^{n+\frac{1}{2}}$ after decompression
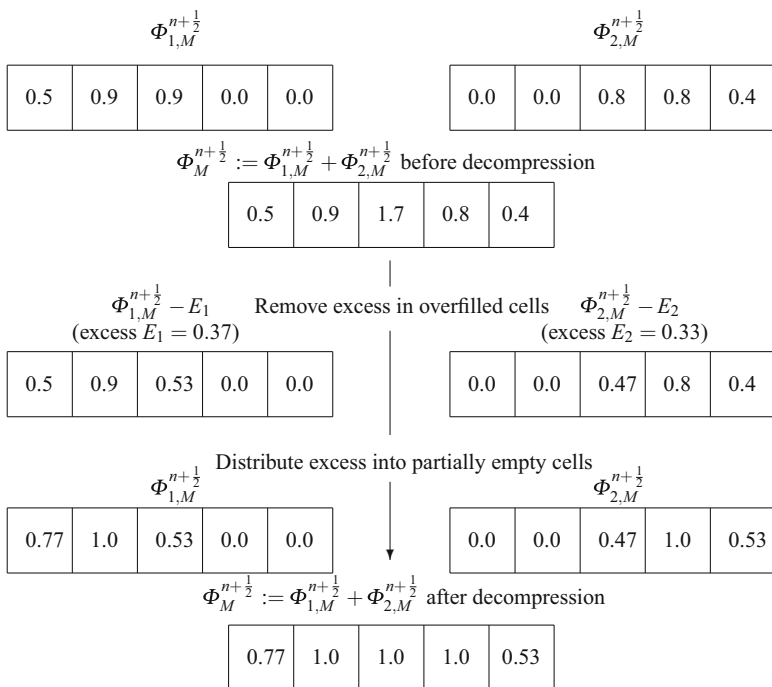
| 0.77 | 1.0 | 1.0 | 1.0 | 0.53 |

Fig. 21.36 Decompression algorithm in the case of two liquid phases. The volume fractions in excess in some cells are redistributed into the under filled cells, proportionally to the contribution of each phase. The total liquid volume fraction is given by $\Phi_M^{n+\frac{1}{2}} = \Phi_{1,M}^{n+\frac{1}{2}} + \Phi_{2,M}^{n+\frac{1}{2}}$. The excesses are first removed in overfilled cells proportionally to the contribution of each phase $(0.37 = 0.7 \cdot (0.9/1.7)$ and $0.33 = 0.7 \cdot (0.8/1.7))$. The excesses are then redistributed into each phase independently before recalculating the total liquid volume fraction $\Phi_M^{n+\frac{1}{2}} = \Phi_{1,M}^{n+\frac{1}{2}} + \Phi_{2,M}^{n+\frac{1}{2}}$.

6.2.3 The Correction Step

After the prediction step, the approximations of the liquid domains Ω_K^{n+1} and the set of liquid elements τ_K^{n+1} are defined as in the single phase case. Note that the approximation of the liquid domains $\Omega_{\ell,K}^{n+1}$ can be defined similarly, but they are not used explicitly in the correction step. Indeed the global Stokes system is defined and solved on the global liquid domain, and the interfaces between phases are implicitly taken into account in a diffuse modeling via the density and viscosity fields. In fact, the velocity and pressure correction $\mathbf{U}_K^{n+1} \in (\mathbb{V}^{FEM}(\tau_K^{n+1}))^d$, $P_K^{n+1} \in \mathbb{V}^{FEM}(\tau_K^{n+1})$ are defined as the solution to (21.28) upon redefining on each tetrahedral element $T \in \tau_K^{n+1}$ the density and viscosity as

$$\rho|_T := \rho^{n+1}|_T := \frac{1}{d+1} \sum_{i=1}^{d+1} \frac{\sum_{\ell=1}^P \Phi_{\ell,K}^{n+1}(\mathbf{v}_i^T)\rho_\ell}{\sum_{\ell=1}^P \Phi_{\ell,K}^{n+1}(\mathbf{v}_i^T)},$$

$$\mu| := \mu^{n+1}|_T := \frac{1}{d+1} \sum_{i=1}^{d+1} \frac{\sum_{\ell=1}^P \Phi_{\ell,K}^{n+1}(\mathbf{v}_i^T)\mu_\ell}{\sum_{\ell=1}^P \Phi_{\ell,K}^{n+1}(\mathbf{v}_i^T)},$$

where $\{\mathbf{v}_i^T, i = 1,\dots,d+1\}$ denotes the vertices of T.

6.3 Numerical Results for Multiphase Flows

We consider again the extrusion with initial contraction described in Section 4.3.1. The computational domain is still the one reported in Figure 21.11.

6.3.1 Successive Phases

We consider three incompressible and immiscible liquid phases, each of them a Newtonian fluid, with equal densities $\rho_1 = \rho_2 = \rho_3 = 1300$ kg m^{-3}, and corresponding (equal) viscosities $\mu_1 = \mu_2 = \mu_3 = 10$ kg(ms)$^{-1}$. As the goal of this example is to study the accuracy of the splitting algorithm, the choice of the three phases is artificial. The liquids are initially located in a successive sequence such that the liquid 1 is pushing the liquid 2 and then the liquid 3. The boundary conditions at the inflow boundary are $\mathbf{u} = 0.00023$ ms^{-1}, such that the order of magnitude of the velocity in the die is approximately 0.05 ms^{-1}. There is only a liquid from phase 1 that flows inside the computational domain. Slip boundary conditions are imposed on $\partial\Lambda$, except at the bottom of the computational domain where no-slip boundary conditions are enforced. Gravity forces, with amplitude $|\mathbf{g}| = 9.81$ ms^{-2} are oriented along the die. The time step is constant and equal to $\delta t = 0.005$ s.

Figure 21.37 illustrates, in a medium plane inside the tube, snapshots of the volume fractions of liquid Φ_ℓ and of the magnitude of the corresponding speed $|\mathbf{U}|$. We observe that the operator splitting algorithm does not introduce any additional error

as long as the flow is laminar (i.e., does not hit the boundary of the computational domain and starts to buckle). The liquid 1 is perfectly pushing the liquids 2 and 3. The velocity is perfectly aligned with the direction of the die, even when using different values of the time step if needed, without the drawback of a CFL condition. The mass in each phase is conserved.

The previous results have been obtained when the three phases have the same densities and viscosities. We now provide a short sensitivity analysis with respect to the value of the viscosities, all the other physical quantities remaining the same. More precisely, we consider again three successive liquid phases. The initial configuration, denoted (a), with $\mu_1 = \mu_2 = \mu_3 = 10$, is compared with two cases, namely (b) $\mu_1 = 10$, $\mu_2 = 1$, $\mu_3 = 0.1$, and (c) $\mu_1 = 10$, $\mu_2 = \mu_3 = 0.1$. Figure 21.38 illustrates in a medium plane inside the tube, snapshots of the volume fractions of liquid
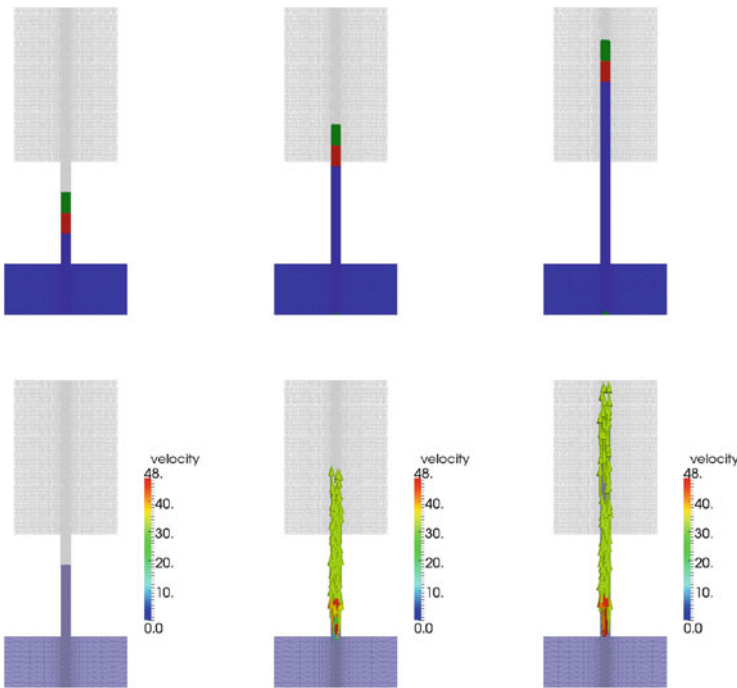

Fig. 21.37 Die swell with extrusion of a Newtonian fluid with three phases (liquid 1 in blue, liquid 2 in red, liquid 3 in green). Snapshots of the solution at times $t = 0, 0.5$ and 1.0 s on a plane located in the middle of the tubes. Top: representation of volume fractions of liquid Φ_ℓ; bottom: velocity field **U**.

Φ_ℓ for the three configurations. One can observe that, when the two liquid phases at the front of the jet have a smaller viscosity, they are crashed by the more viscous phases when the jet hits the boundary of the domain; in that particular case, the less viscous liquid phases do not contribute to the buckling effect. The difference

between configurations (b) and (c) is not remarkable. Before touching the boundary of the domain, the laminar behavior of the three liquid phases is identical to that illustrated in Figure 21.37.
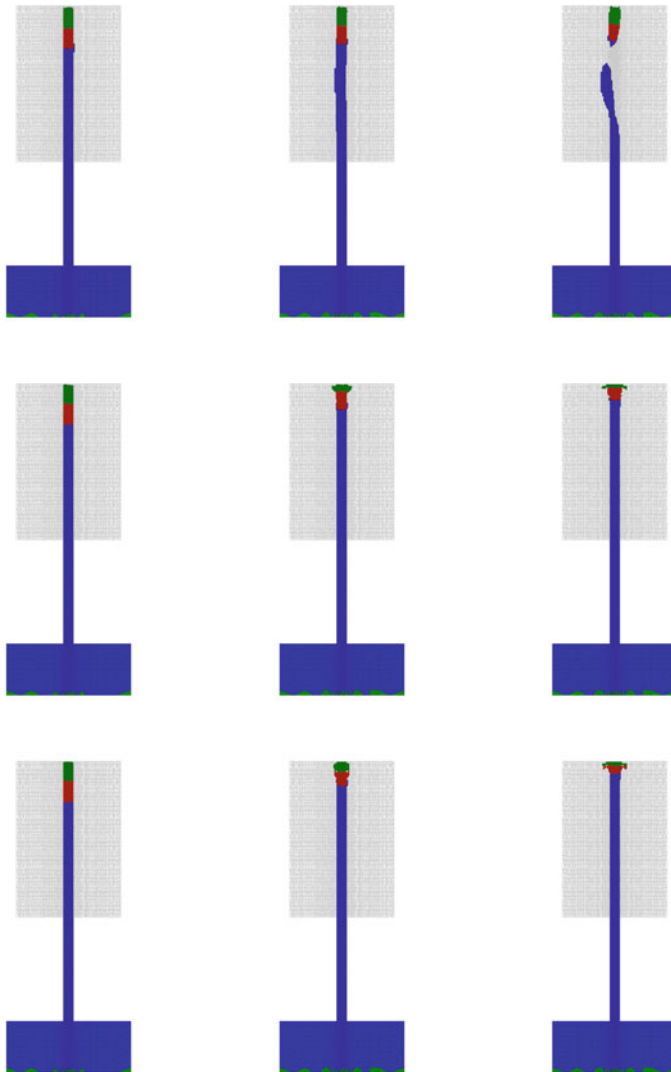

Fig. 21.38 Die swell with extrusion of a Newtonian fluid with three phases (liquid 1 in blue, liquid 2 in red, liquid 3 in green). Snapshots of the solution at times $t = 1.2, 1.3$, and 1.4 s on a plane located in the middle of the tubes. First row: (a) $\mu_1 = \mu_2 = \mu_3 = 10$; Second row: (b) $\mu_1 = 10$, $\mu_2 = 1$, $\mu_3 = 0.1$; Third row: (c) $\mu_1 = 10$, $\mu_2 = \mu_3 = 0.1$.

6.3.2 Parallel Phases

Finally, let us consider the configuration where the three liquid phases (with equal viscosities) are next to each other. The initial configuration is illustrated in Figure 21.39 and shows that each phase is contained in one-third of the total angle along the die direction. Figure 21.40 shows that the three phases remain parallel when advected through the operator splitting algorithm (the red phase is 'hidden' behind the two other phases!). The velocities are parallel and the reconstruction of the interface does not jeopardize the approximation of the location of each liquid phase. Figure 21.41 shows the buckling effect when the jet hits the boundary of the domain; as all phases have the same viscosity, this effect is quite similar as in the case of one single Newtonian liquid.
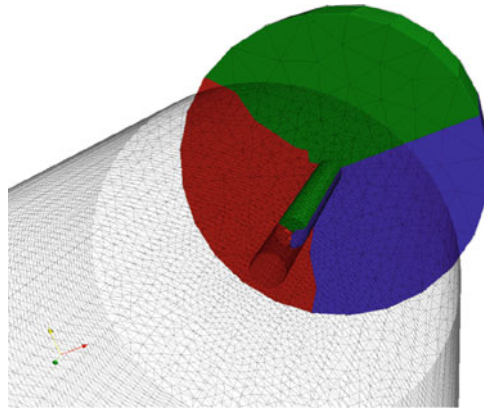

Fig. 21.39 Die swell with extrusion of a Newtonian fluid with three phases (liquid 1 in blue, liquid 2 in red, liquid 3 in green). Initial configuration of the three phases, each having one-third of the total volume.

7 Perspectives: Application to Emulsion in Food Engineering

The simulation of emulsion in microfluidic devices is a stringent application, as the physical process involves instabilities and strong influence from surface tension effects. Thus the numerical method requires an accurate approximation of the interfaces and of those surface tension effects. Further details about microfluidic emulsions can be found in [3, 7, 27] and references therein. Applications of interest exist in food engineering when producing types of mayonnaise for instance [17, 18]. Furthermore, from the numerical viewpoint, adaptive mesh refinement techniques help tremendously to increase the accuracy of the method and sharpen the approximation of the interfaces. Details about an adaptive method making a first attempt into this direction can be found in [9] when discussing the mesh refinement between one liquid phase and a vacuum. The same type of techniques have been extended in
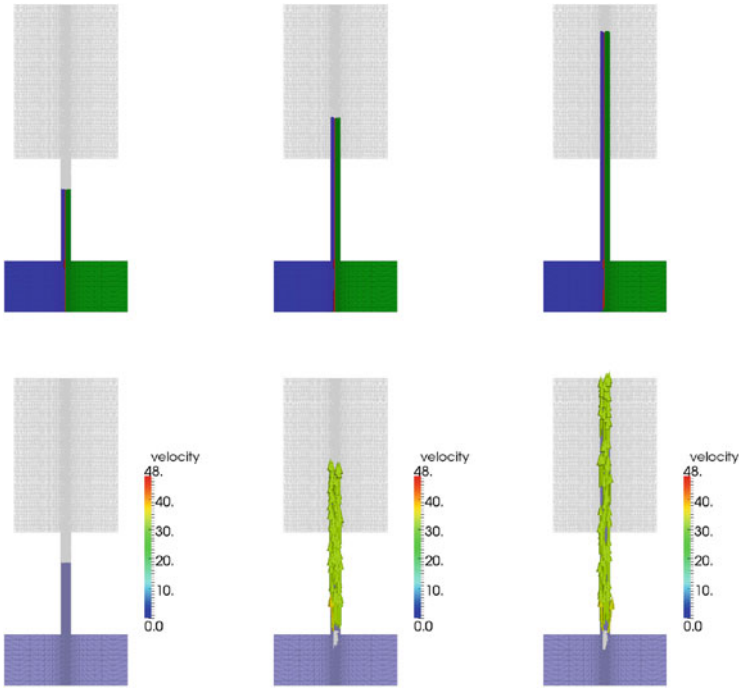
Fig. 21.40 Die swell with extrusion of a Newtonian fluid with three phases (liquid 1 in blue, liquid 2 in red, liquid 3 in green). Snapshots of the solution at times $t = 0, 0.5$, and 1.0 s on a plane located in the middle of the tubes. Top: representation of volume fraction of liquid Φ_ℓ; bottom: velocity field **U**.
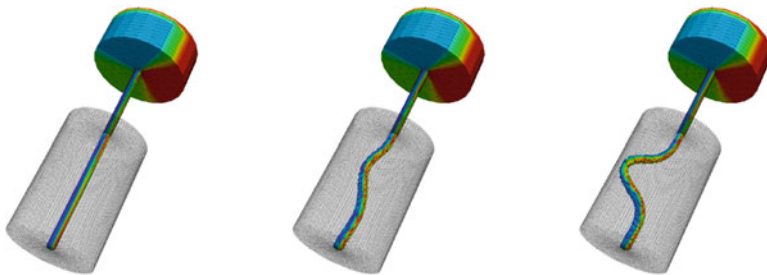

Fig. 21.41 Die swell with extrusion of a Newtonian fluid with three phases (liquid 1 in blue, liquid 2 in red, liquid 3 in green). Snapshots of the approximation of the liquid domain at times $t = 1.2, 1.4$, and 1.6 s.

subsequent work to mesh refinement around interfaces between two liquid phases. The results presented in this section have been obtained by P. Clausen while staying at EPFL on a postdoctoral position. Details of the method will be presented in a forthcoming paper.

In order to illustrate such a situation, we consider a microfluidic device composed by a tube intersected by another tube. The geometrical domain, as well as the corresponding finite element mesh $\mathscr{F}^{FEM}$, are shown in Figure 21.42. A liquid from phase 1 is introduced at the longitudinal entrance, while a phase 2 liquid is injected transversally and "cuts" the flow of the liquid 1 to form droplets of one liquid phase trapped into the other. This phenomenon is called *droplet breakup*, and is repeated periodically by the process leading to the formation of a sequence of droplets. The velocity is initialized with a parabolic velocity profile at each of the three entrances (a zero tangential velocity is prescribed, and the normal velocity is given by a parabolic profile). Along the channel, no-slip boundary conditions are prescribed. At the outlet, zero tangential velocity and zero normal stress are enforced.
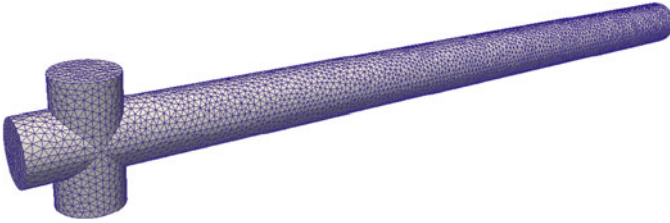

Fig. 21.42 Microfluidic emulsion simulation. Description of the geometry and representation of the finite element mesh $\mathscr{F}^{FEM}$

In an emulsion, two motions are interacting: first the movement initiated by the flow induced by the inlet velocities; second the displacements induced by the surface tension effects at the interfaces. These two effects are on different time scales, and the time step we choose has to take into account the smallest of these two scales. Despite the fact that our method does not suffer from a CFL condition, here we observe that the treatment of the surface tension effects impose a constraint on the time step to prevent instabilities.

For illustration, we consider the injection of oil in water. Oil has density 1000 kg m^{-3} and viscosity 0.5 kg(ms)$^{-1}$, while water has density 1000 kg m^{-3} and viscosity 0.001 kg(ms)$^{-1}$. The surface tension coefficient is given by $\gamma = 0.02$ Nm^{-1}. Oil is introduced with a maximum velocity of $\mathbf{u}_{max,oil} = 0.01$ ms^{-1}, while water is injected with a maximum velocity of $\mathbf{u}_{max,water} = 0.02$ ms^{-1}.

Figure 21.43 illustrates, in a medium plane inside the tube, snapshots of the two liquid phases and the corresponding velocity field.

The results of these numerical experiments show the difficulty of producing a regular succession of droplet breakings, which is controlled by the balance between surface energy and viscosity effects. Numerical difficulties include droplets generated with different sizes and volume losses when the stabilization terms are large.
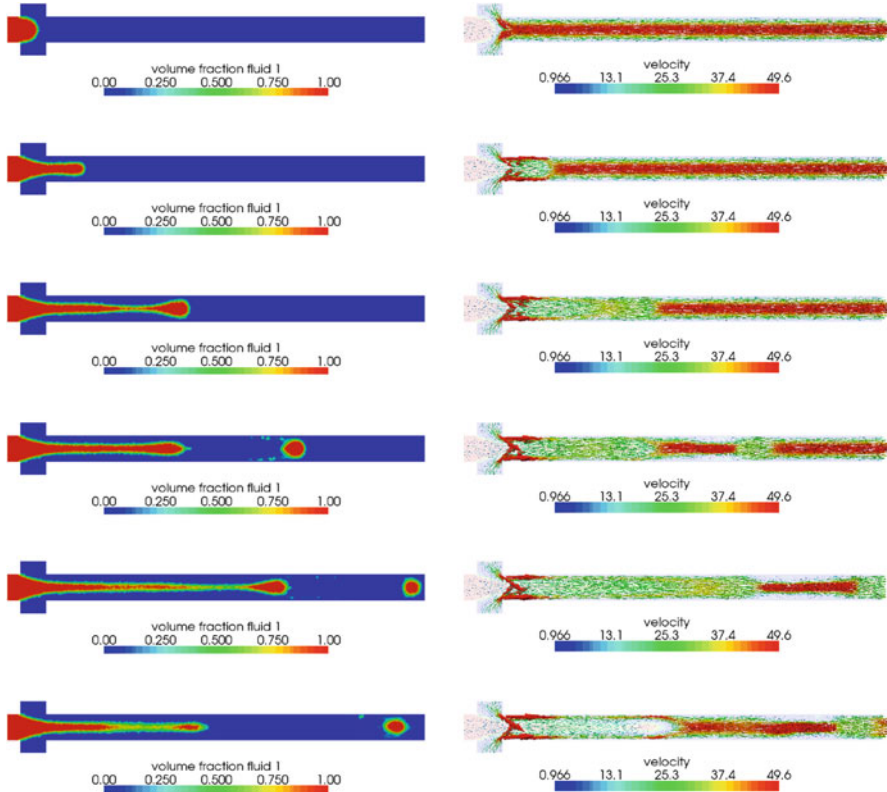
Fig. 21.43 Microfluidic emulsion simulation. Snapshots of the solution at times $t = 0.2$, 0.5,0.8,1.1,1.4, and 1.7 s on a plane located in the middle of the longitudinal tube. Left: representation of volume fraction of liquid ; right velocity field.

References

1. Antonietti, P.F., Fadel, N.A., Verani, M.: Modelling and numerical simulation of the polymeric extrusion process in textile products. Commun. Appl. Ind. Math. **1**(2), 1–13 (2010)
2. Bänsch, E.: Finite element discretization of the Navier-Stokes equations with a free capillary surface. Numer. Math. **88**(2), 203–235 (2001)
3. Baroud, C.N., Gallaire, F., Dangla, R.: Dynamics of microfluidic droplets. Lab on a Chip **10**(16), 2032 (2010)
4. Bird, R., Curtiss, C., Armstrong, R., Hassager, O.: Dynamics of Polymeric Liquids, vol. 1 and 2. John Wiley & Sons, New-York (1987)
5. Bonito, A., Clément, P., Picasso, M.: Mathematical analysis of a simplified Hookean dumb-bells model arising from viscoelastic flows. J. Evol. Equ. **6**(3), 381–398 (2006)
6. Bonito, A., Picasso, M., Laso, M.: Numerical simulation of 3D viscoelastic flows with free surfaces. J. Comput. Phys. **215**(2), 691–716 (2006)
7. Brun, P.T., Nagel, M., Gallaire, F.: Generic path for droplet relaxation in microfluidic channels. Physical Review E **88**(4) (2013)

8. Caboussat, A.: A numerical method for the simulation of free surface flows with surface tension. Computers and Fluids **35**(10), 1205–1216 (2006)
9. Caboussat, A., Clausen, P., Rappaz, J.: Numerical simulation of two-phase flow with interface tracking by adaptive Eulerian grid subdivision. Math. Comput. Modelling **55**, 490–504 (2012)
10. Caboussat, A., James, N., Boyaval, S., Picasso, M.: Numerical simulation of free surface flows, with multiple liquid phases. In: E. Onate, J. Oliver, A. Huerta (eds.) Proceedings of the 11th World Congress on Computational Mechanics (WCCM XI), pp. 5381–5391 (2014)
11. Caboussat, A., Maronnier, V., Picasso, M., Rappaz, J.: Numerical simulation of three dimensional free surface flows with bubbles. In: Challenges in Scientific Computing—CISC 2002, *Lect. Notes Comput. Sci. Eng.*, vol. 35, pp. 69–86. Springer, Berlin (2003)
12. Caboussat, A., Picasso, M., Rappaz, J.: Numerical simulation of free surface incompressible liquid flows surrounded by compressible gas. J. Comput. Phys. **203**(2), 626–649 (2005)
13. Choi, B.Y., Bussmann, M.: A piecewise linear approach to volume tracking a triple point. Int. J. Numer. Methods Fluids **53**(6), 1005–1018 (2007)
14. Douglas, J., Rachford, H.H.: On the numerical solution of heat conduction problems in two and three space variables. Trans. Amer. Math. Soc. **82**, 421–439 (1956)
15. Glowinski, R.: Finite element methods for incompressible viscous flow. In: P.G. Ciarlet, J.L. Lions (eds.) Handbook of Numerical Analysis, vol. IX, pp. 3–1176. North-Holland, Amsterdam (2003)
16. Glowinski, R., Dean, E.J., Guidoboni, G., Juárez, L.H., Pan, T.W.: Applications of operator-splitting methods to the direct numerical simulation of particulate and free-surface flows and to the numerical solution of the two-dimensional elliptic Monge-Ampère equation. Japan J. Indust. Appl. Math. **25**(1), 1–63 (2008)
17. Gunes, D.Z., Bercy, M., Watske, B., Breton, O., Burbidge, A.S.: A study of extensional flow induced coalescence in microfluidic geometries with lateral channels. Soft Matter **9**, 7526–7537 (2013)
18. Hughes, E., Maan, A.A., Acquistapace, S., Burbidge, A., Johns, M.L., Gunes, D.Z., Clausen, P., Syrbe, A., Hugo, J., Schroen, K., Miralles, V., Atkins, T., Gray, R., Homewood, P., Zick, K.: Microfluidic preparation and self diffusion PFG-NMR analysis of monodisperse water-in-oil-in-water double emulsions. J. Colloid and Interface Science **389**, 147–156 (2013)
19. James, N., Boyaval, S., Caboussat, A., Picasso, M.: Numerical simulation of 3D free surface flows, with multiple incompressible immiscible phases. Applications to impulse waves. Int. J. Numer. Meth. Fluids. **76**(12), 1004–1024 (2014)
20. Jouvet, G., Huss, M., Blatter, H., Picasso, M., Rappaz, J.: Numerical simulation of Rhonegletscher from 1874 to 2100. J. Comput. Phys. **228**(17), 6426–6439 (2009)
21. Jouvet, G., Picasso, M., Rappaz, J., Huss, M., Funk, M.: Modelling and numerical simulation of the dynamics of glaciers including local damage effects. Math. Model. Nat. Phenom. **6**(5), 263–280 (2011)
22. Kratzer, A., Handschin, S., Lehmann, V., Gross, D., Escher, F., Conde-Petit, B.: Hydration dynamics of durum wheat endosperm as studied by magnetic resonance imaging and soaking experiments. Cereal Chemistry **85**(5), 660–666 (2008)
23. Marchuk, G.I.: Methods of Numerical Mathematics. Springer-Verlag, New York (1975)
24. Marchuk, G.I.: Splitting and alternating direction methods. In: P.G. Ciarlet, J.L. Lions (eds.) Handbook of Numerical Analysis, vol. I, pp. 197–462. North-Holland, Amsterdam (1990)
25. Maronnier, V., Picasso, M., Rappaz, J.: Numerical simulation of free surface flows. J. Comput. Phys. **155**(2), 439–455 (1999)
26. Maronnier, V., Picasso, M., Rappaz, J.: Numerical simulation of three-dimensional free surface flows. Int. J. Numer. Meth. Fluids **42**(7), 697–716 (2003)
27. Nagel, M., Brun, P.T., Gallaire, F.: A numerical study of droplet trapping in microfluidic devices. Physics of Fluids **26**(3), 032,002 (2014)
28. Noh, W.F., Woodward, P.: SLIC (Simple Line Interface Calculation). Proceedings of the Fifth International Conference on Numerical Methods in Fluid Dynamics **59**, 330–340 (1976)
29. Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. J. Comput. Phys. **79**(1), 12–49 (1988)

30. Öttinger, H.C.: Stochastic Processes in Polymeric Fluids. Springer-Verlag, Berlin (1996)
31. Parolini, N., Burman, E.: A finite element level set method for viscous free-surface flows. In: Applied and industrial mathematics in Italy, *Ser. Adv. Math. Appl. Sci.*, vol. 69, pp. 416–427. World Sci. Publ., Hackensack, NJ (2005)
32. Peaceman, D.W., Rachford, H.H.: The numerical solution of parabolic and elliptic differential equations. J. Soc. Indust. Appl. Math. **3**, 28–41 (1955)
33. Picasso, M., Rappaz, J., Reist, A.: Numerical simulation of the motion of a three-dimensional glacier. Ann. Math. Blaise Pascal **15**(1), 1–28 (2008)
34. Pironneau, O.: On the transport-diffusion algorithm and its applications to the Navier-Stokes equations. Numer. Math. **38**(3), 309–332 (1981/82)
35. Shashkov, M., Wendroff, B.: The repair paradigm and application to conservation laws. J. Comput. Phys. **198**(1), 265–277 (2004)
36. Tome, M., McKee, S.: Numerical simulation of viscous flow: Buckling of planar jets. Int. J. Numer. Meth. Fluids **29**(6), 705–718 (1999)
37. Tryggvason, G., Scardovelli, R., Zaleski, S.: Direct Numerical Simulations of Gas-Liquid Multiphase Flows. Cambridge University Press, Cambridge (2011)
38. Turek, S.: Efficient Solvers for Incompressible Flow Problems, *Lecture Notes in Computational Science and Engineering*, vol. 6. Springer-Verlag, Berlin (1999)
39. Yanenko, N.N.: The Method of Fractional Steps. The Solution of Problems of Mathematical Physics in Several Variables. Springer-Verlag, New York (1971)
40. Ycoor Systems S.A.: cfsflow. http://www.ycoorsystems.com/. Online; accessed September 2014