Journal of Computational Physics 229 (2010) 2740–2758

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Two-phase free-surface fluid dynamics on moving domains

Matthias Preisig [a,*], Thomas Zimmermann [b]

[a] EPFL-ENAC-LSMS, Station 18, CH-1015 Lausanne, Switzerland
[b] Zace Services Ltd, CH-1015 Lausanne, Switzerland

ARTICLE INFO

Article history:
Received 13 May 2009
Received in revised form 6 October 2009
Accepted 15 December 2009
Available online 29 December 2009

MSC:
65M60
76T99

Keywords:
Free-surface viscous flow
Two-phase
Lagrangian formulation
Debris flow

ABSTRACT

We develop in this paper a model for the numerical simulation of free surface debris- and mudflows, described as a mixture of two viscous phases. A Lagrangian approach with finite element approximation and regular remeshing is adopted in this paper. The proposed model solves the equations of motion in both phases, coupled with conservation of volume of the mixture, implicitly; volume fractions are computed explicitly at the end of each iteration. A one-step trapezoidal scheme is used for time integration. The resulting model combines simplicity, robustness and versatility and provides a novel and unified algorithmic framework for a large spectrum of applications. It is also easily extensible to three spatial dimensions, more complex constitutive models and coupling with geomechanical models. Application to sedimentation and to impact on an obstacle shows the effectiveness of the approach.

1. Introduction

In this paper we are interested in modeling the flow of a mass of mud as it propagates down along a slope. This problem requires adequate capturing of the free surface in order to satisfy mass conservation, and the ability to follow the flow mass over an extended path. Finite elements in space and finite differences in time appears to be the natural choice for a formulation which should ultimately be coupled with geomechanical capabilities. A Lagrangian formulation with rezoning of nodes, remeshing and remapping of nodal variables permits dealing with this type of problems. In the present paper, we extend the single-phase method to develop a method capable of representing a general two-phase material. The underlying single-phase model is documented in [22]. Parts of this work have been presented in [20,21].

Current state of the art in mud- or debris-flow modeling is almost exclusively based on depth-averaged models. The governing equations are integrated over depth and the depth of the flow, velocity and stress at the base are computed for each grid point. Some authors assume a hydrostatic pressure distribution [19], others assume a velocity profile and compute the pressure as a function of depth and the averaged velocity [23].

Single-phase models have been used extensively in the past for the modeling of mudflows. Newtonian models that are used in hydraulics can be extended to account for the transport of solid material up to a certain volume fraction. These models can be refined by allowing the viscosity of the fluid to vary as a function of the solid volume fraction or the velocity gradient. Such constitutive models are referred to as non-Newtonian fluid models. Bingham fluids have a yield shear stress below which no flow occurs. They can be used for instance for simulating the stopping of a flow as the slope flattens out. A lot of work in this field was done by Chen, see for instance [3,2]. Depth-averaged mudflow models frequently use a

* Corresponding author.
E-mail address: mpreisig@gmail.com (M. Preisig).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.12.020

combination of Newtonian or non-Newtonian fluids together with a Mohr-Coulomb friction law at the base of the flow [23,19]. In current engineering practice the flood-routing software FLO-2D,[1] which implements a depth-averaged formulation and offers a variety of constitutive models, is commonly used to predict flow-paths and establish hazard maps.

Single-phase models are, however, limited when it comes to predicting complex behavior of mudflows without the possibility to calibrate the model. As Hutter points out in [14] this is mainly due to neglecting the effect of the interstitial fluid. The effect of pore pressure on the fluidization of a solid–fluid mixture is often indispensable for accurately describing the mechanics of mudflows.

Two-phase models have been around in the literature for a long time describing processes in nuclear engineering, where the behavior of air–water mixtures is analyzed. The efforts of the petroleum industry have also led to considerable progress in the field of modeling of multi-phase fluids. The books by Soo [25] and Kolev [17] are two references worth citing. In the field of mud or debris flows Takahashi [27] was one of the first researchers to recognize the importance of taking into account the interaction between two phases and integrating it in a mathematical model. Svendsen et al. [26] laid out a thorough foundation for a general mixture of multiple phases by deriving all the balance equations from thermodynamical principles. Finally, Iverson and Denlinger [16,6] generalized the depth-averaged model by Savage et al. [23] for the flow of a two-phase mixture over three-dimensional terrain.

In the process of depth averaging information on the distribution of stresses and velocities is lost. In order to compute forces on protection structures a detailed vertical stress profile is required. Such detailed information can only be obtained from full three-dimensional continuum models. Furthermore, a continuum approach is the only sound way to obtain a model that can simulate all aspects of mudflows: initiation of the flow, propagation, entrainment of material on the flow path (erosion), extent of spreading of the deposition zone, velocity of propagation and forces. Only a few models that offer the possibility for such capabilities to be integrated are available in the literature. We note the work of Shao et al. [24] and Laigle et al. [18], who adopted a meshless method to simulate the impact of a mudflow on a structure. Both authors used a single-phase, non-Newtonian model. A two-phase model using an Eulerian approach in conjunction with level sets to define the free surface was developed by Frenette et al. [9].

In our approach, the two-phase material is a mixture of two continuous materials with separate velocity fields. Interaction between the two phases is modeled by a relation of momentum exchange. We limit ourselves in this work to a very simple constitutive model for both phases, in this case two Newtonian fluids. The algorithm is formulated to make it possible to accommodate more complex constitutive models without any major changes.

The mass of each phase is transported by nodes, whose positions are updated continuously according to the velocity field of the phase. At the beginning of each iteration all fields transported with the phases are mapped onto a new mesh that is common to both phases. This step allows the solution of the governing equation on a unique set of nodes for both phases. At the same time it also regularizes the mesh and avoids problems related to mesh distortion.

In Section 2 the main ideas and hypotheses of the model are laid out. Sections 3 and 4 give the governing equations and the corresponding weak form. After giving some indications on the spatial discretization technique used (Section 5) the time integration scheme is presented in detail in Section 6. Section 7 is devoted to the method for computing volume fractions. Some indications on the implementation are given in Section 8. In Section 9 numerical applications illustrate the performance of the method. Sedimentation of a heavier fluid in a mixture with a less dense fluid is compared with an analytical solution, and the impact of a mudflow on an obstacle is simulated and forces acting on the obstacle are extracted. Finally, conclusions are drawn in Section 10.

2. Two-phase mixture

The physical description of the two-phase fluid is inspired by the definitions given by Kolev [17] and Soo [25]. Both phases are modeled as viscous fluids. The phases are defined as follows:

- The fluid phase is a homogeneous mixture of water and microscopic solid particles. These solid particles are surrounded by water only.
- The pseudo solid phase, called solid phase in the sequel, consists of soil particles that are sufficiently large to start to sediment during the time span considered in the analysis. The solid phase includes water contained in pores and a fine layer surrounding the soil particles.

We assume both phases to be present in the entire domain. This means that no phase interfaces need to be tracked, the two phases are smeared over the volume. Their corresponding presence is given by a volume fraction that takes values between 0 and 1. As a consequence, stresses are essentially independent in the two phases; interaction between phases does not occur at an interface, but in a diffuse way through momentum exchange.

In order to remain most general in the description of the two-phase fluid, a mathematical structure is chosen that uses the momentum balance equations of both the solid and the fluid phase. The mass balance equations of both phases are combined into one equation that enforces conservation of the mass of the mixture. With the pressure being the same in both phases we

[1] www.flo-2d.com.

solve five coupled equations for five nodal unknowns: two solid and two fluid velocity components and one pressure. Volume fractions, as additional nodal unknowns, are treated explicitly.

In the flow of a two-phase mixture interaction between the phases becomes very important. The starting point for our model is a single-phase incompressible viscous fluid. By choosing both phases to be viscous fluids we are able to match single-phase behavior as a limit case. A momentum exchange term models interaction between the two phases. We provide an algorithmic framework that tracks the evolution of the two phases by their volume fractions. This conceptually simple model makes it possible to analyze the behavior of the mixture on simple test problems, and can accommodate more complex material behavior without major changes in the formulation.

Both phases are present in the entire domain. The phases occupy the computational domain completely. A unique pressure applies in both phases, which is justified by the absence of grain-to-grain contact making it possible to support a solid pressure.

3. Governing equations

3.1. Volume fractions

The volume fraction of phase p is given by C_p, where $p = s$ identifies the solid and $p = f$ the fluid phase:

$$C_p = \frac{V_p}{V} \tag{1}$$

Here, V represents the control volume and V_p the volume within V that is occupied by phase p. The relation

$$C_s + C_f = 1 \tag{2}$$

follows from the assumption that the phases fill the domain completely. Considering a control volume that is fixed in space another important observation can be made:

$$\frac{\partial C_s}{\partial t} + \frac{\partial C_f}{\partial t} = 0 \tag{3}$$

The equations of mass and momentum conservation of the phases can be derived from volume averaging, as shown in Appendix A, and finally written in terms of volume fractions.

3.2. Conservation of mass

We write the equations of conservation of mass for both phases in the Eulerian form:

$$\frac{\partial C_s \rho_s}{\partial t} + \nabla \cdot (C_s \rho_s \mathbf{u}_s) = 0 \tag{4}$$

$$\frac{\partial C_f \rho_f}{\partial t} + \nabla \cdot (C_f \rho_f \mathbf{u}_f) = 0 \tag{5}$$

ρ_s and ρ_f are the (intrinsic) phase densities and $\mathbf{u}_s$ and $\mathbf{u}_f$ stand for the phase velocities. With Eq. (3) and making use of incompressibility of both phases we obtain an equation of conservation of volume of the mixture:

$$\nabla \cdot (C_s \mathbf{u}_s) + \nabla \cdot (C_f \mathbf{u}_f) = 0 \tag{6}$$

3.3. Conservation of momentum

Conservation of momentum is written for both phases:

$$C_s \rho_s \frac{D\mathbf{u}_s}{Dt} = \nabla \cdot \langle \sigma_s \rangle + C_s \rho_s \mathbf{f} + \mathbf{m}_{sf} \tag{7}$$

$$C_f \rho_f \frac{D\mathbf{u}_f}{Dt} = \nabla \cdot \langle \sigma_f \rangle + C_f \rho_f \mathbf{f} + \mathbf{m}_{fs} \tag{8}$$

where $D\mathbf{u}/Dt$ is the material derivative of the velocity. $\mathbf{m}_{sf} = -\mathbf{m}_{fs}$ represents the momentum exchange term and $\mathbf{f}$ is an acceleration (e.g. gravitational acceleration).

3.4. Constitutive relation

The constitutive relations of the two-phase fluid are derived from the constitutive relation of a single-phase viscous fluid. In order to make sure that the limit case of two phases with exactly the same properties is identical with the situation of a single-phase the following relation must hold:

$$\langle\sigma\rangle = \langle\sigma_s\rangle + \langle\sigma_f\rangle \tag{9}$$

This relation is satisfied for the following constitutive equations:

$$\langle\sigma_s\rangle = C_s(\tau(\mathbf{u}_s) + p\mathbf{I}) \tag{10}$$

$$\langle\sigma_f\rangle = C_f(\tau(\mathbf{u}_f) + p\mathbf{I}) \tag{11}$$

I is an identity matrix. The deviatoric stress tensors are defined as

$$\tau(\mathbf{u}_s) = 2\mu_s\left(\epsilon(\mathbf{u}_s) - \frac{1}{3}(\nabla\cdot\mathbf{u}_s)\mathbf{I}\right) \tag{12}$$

$$\tau(\mathbf{u}_f) = 2\mu_f\left(\epsilon(\mathbf{u}_f) - \frac{1}{3}(\nabla\cdot\mathbf{u}_f)\mathbf{I}\right) \tag{13}$$

where μ_s and μ_f are the dynamic viscosities of the solid and the fluid phase. The symmetric parts of the velocity gradients are given by $\epsilon_s = \frac{1}{2}(\nabla\mathbf{u}_s + (\nabla\mathbf{u}_s)^T)$ and $\epsilon_f = \frac{1}{2}(\nabla\mathbf{u}_f + (\nabla\mathbf{u}_f)^T)$.

Remark 1. We can easily verify that by adding Eqs. (12) and (13) and substituting μ for μ_s and μ_f and $\mathbf{u}$ for $\mathbf{u}_s$ and $\mathbf{u}_f$ we obtain the constitutive relation of a single-phase fluid:

$$\langle\sigma\rangle = \langle\sigma_s\rangle + \langle\sigma_f\rangle = 2(C_s + C_f)\mu\left(\epsilon(\mathbf{u}) - \frac{1}{3}(\nabla\cdot\mathbf{u})\mathbf{I}\right) + p\mathbf{I} = 2\mu\left(\epsilon(\mathbf{u}) - \frac{1}{3}(\nabla\cdot\mathbf{u})\mathbf{I}\right) + p\mathbf{I} \tag{14}$$

3.5. Momentum exchange

As shown in Appendix A a momentum exchange term arises naturally if we apply volume averaging to the momentum conservation equation of one of the phases. This momentum exchange term is expressed in the form of a surface integral over the interface between the two phases. Lets denote the momentum exchange term, transferring momentum from the fluid phase to the solid phase, by $\mathbf{m}_{sf}$:

$$\mathbf{m}_{sf} = \frac{1}{V}\int_\Gamma \sigma_s\cdot\mathbf{n}_s\,dS \tag{15}$$

From the point of view of the fluid phase the momentum exchange term has to be equal but with opposite sign: $\mathbf{m}_{fs} = -\mathbf{m}_{sf}$.

$$\mathbf{m}_{fs} = -\mathbf{m}_{sf} = \frac{1}{V}\int_\Gamma \sigma_f\cdot\mathbf{n}_f\,dS \tag{16}$$

The reciprocity is satisfied if the following holds:

$$\sigma_s\cdot\mathbf{n}_s = -\sigma_f\cdot\mathbf{n}_f \quad \text{on } \Gamma \tag{17}$$

Continuity of the stress at the interface requires $\sigma_s = \sigma_f$. The unit normal vector pointing outward of the fluid phase at the solid–fluid interface is equal to the negative of the unit normal vector pointing outward of the solid phase: $\mathbf{n}_s = -\mathbf{n}_f$. Reciprocity between the two momentum exchange terms is therefore satisfied.

In our model we have no explicit modeling of the interface. Therefore, we need to find an expression of the momentum exchange where this is not required.

To gain insight into the structure of the momentum exchange term let us consider the case of a sphere pulled through a viscous fluid. Stokes' law says that the drag force F applied to a very small spherical particle of radius r follows the linear relationship

$$F = 6\pi\mu r v \tag{18}$$

where μ is the viscosity and v the velocity of the sphere. A linear relationship is commonly adopted to express the momentum exchange term, as, e.g. in De la Cruz et al. [5], who derived the term by expanding the surface integral (Eq. (16)) in powers and keeping only the lowest order terms. The resulting relationship can be written as:

$$\mathbf{m}_{sf} = -\mathbf{m}_{fs} = K'_{drag}(\mathbf{u}_s - \mathbf{u}_f) \tag{19}$$

An expression for K'_{drag} can be derived by considering a fluid moving past a cloud of particles, which was done in Soo [25]. This leads to

$$K'_{drag} = \frac{75}{2}\frac{C_s}{(1-C_s)^2}\frac{\mu_f}{a^2} = K_{drag}\frac{C_s}{(1-C_s)^2}\mu_f \tag{20}$$

where a is the radius of a particle. $K_{drag} = \frac{75}{2a^2}$ has the unit of $[\mathrm{m}^{-2}]$. Its value has to be determined based on an estimation of the particle diameter a. Even if this model is valid only approximately and for an idealized granular phase it can serve as an indication for selecting appropriate values of K_{drag}. For mudflows it gives values of K_{drag} in the range between 1 for very

coarse-grained mixtures and 10^5 for slurry flows with a large fines content. In problems where the volume fractions vary only slightly and the distribution is smooth K'_{drag} can be assumed to be constant.

3.6. Summary of the initial/boundary value problem

The boundary value problem consists in solving the following equations for velocities and pressure, given the boundary conditions $\mathbf{g}_s, \mathbf{g}_f, \mathbf{h}_s, \mathbf{h}_f$ and the initial conditions $\mathbf{u}_{s,0}$ and $\mathbf{u}_{f,0}$:

$$C_s\rho_s \frac{D\mathbf{u}_s}{Dt} = \nabla\cdot[C_s(\tau(\mathbf{u}_s)+p\mathbf{I})] + C_s\rho_s\mathbf{f} + \mathbf{m}_{sf} \quad \text{on } \Omega\times[0,T] \tag{21}$$

$$C_f\rho_f \frac{D\mathbf{u}_f}{Dt} = \nabla\cdot[C_f(\tau(\mathbf{u}_f)+p\mathbf{I})] + C_f\rho_f\mathbf{f} + \mathbf{m}_{fs} \quad \text{on } \Omega\times[0,T] \tag{22}$$

$$0 = \nabla\cdot(C_s\mathbf{u}_s) + \nabla\cdot(C_f\mathbf{u}_f) \quad \text{on } \Omega\times[0,T] \tag{23}$$

$$\mathbf{u}_s = \mathbf{g}_s \quad \text{on } \Gamma_{g_s}\times[0,T] \tag{24}$$

$$\mathbf{u}_f = \mathbf{g}_f \quad \text{on } \Gamma_{g_f}\times[0,T] \tag{25}$$

$$\langle\sigma_s\rangle\cdot\mathbf{n} = \mathbf{h}_s \quad \text{on } \Gamma_{h_s}\times[0,T] \tag{26}$$

$$\langle\sigma_f\rangle\cdot\mathbf{n} = \mathbf{h}_f \quad \text{on } \Gamma_{h_f}\times[0,T] \tag{27}$$

$$\mathbf{u}_s(t=0) = \mathbf{u}_{s,0} \quad \text{on } \Omega \tag{28}$$

$$\mathbf{u}_f(t=0) = \mathbf{u}_{f,0} \quad \text{on } \Omega \tag{29}$$

Γ_{g_p} denotes the part of the boundary on which we impose the displacement $\mathbf{g}_p$, while Γ_{h_p} denotes the Neumann part, where we impose surface tractions $\mathbf{h}_p$.

4. Weak form

The global weak form is established as follows: Let $\mathscr{S}_i^s = \{u_i^s \in H^1(\Omega) \mid u_i^s = g_i^s \text{ on } \Gamma_{g_i^s}\}$ and $\mathscr{S}_i^f = \{u_i^f \in H^1(\Omega) | u_i^f = g_i^f \text{ on } \Gamma_{g_i^f}\}$ be spaces of solid and fluid trial functions, $\mathscr{V}_i^s = \{w_i^s \in H^1(\Omega) | w_i^s = 0 \text{ on } \Gamma_{g_i^s}\}$ and $\mathscr{V}_i^f = \{w_i^f \in H^1(\Omega) | w_i^f = 0 \text{ on } \Gamma_{g_i^f}\}$ spaces of solid and fluid test functions and $\mathscr{P} = \{p \in L^2(\Omega)\}$ a space of both trial and test functions.[2,3] Then the weak form associated with Eqs. (21)–(29) consists in finding $u_i^s \in \mathscr{S}_i^s, u_i^f \in \mathscr{S}_i^f$ and $p \in \mathscr{P}$, such that for all $w_i^s \in \mathscr{V}_i^s, w_i^f \in \mathscr{V}_i^f$ and $q \in \mathscr{P}$. Combining all components of u^s and u^f into a vector $\mathbf{u}$ the following equation holds, see Appendix B for details:

$$\begin{aligned}
&\int_\Omega \mathbf{w}^T[\mathbf{C}\rho]\dot{\mathbf{u}}\,d\Omega - \int_\Omega \mathbf{w}^T[\nabla\mathbf{C}]_1\overline{\mathbf{D}}\epsilon(\mathbf{u})\,d\Omega - \int_\Omega \mathbf{w}^T[\nabla\mathbf{C}]_2\,p\,d\Omega + \int_\Omega [\nabla\cdot\mathbf{w}C]p\,d\Omega + \int_\Omega \epsilon(\mathbf{w})^T[\mathbf{C}]_1\overline{\mathbf{D}}\epsilon(\mathbf{u})\,d\Omega \\
&\quad + \int_\Omega \mathbf{w}^T\mathbf{K}_{drag}\mathbf{u}\,d\Omega + \int_\Omega q[C\nabla\cdot\mathbf{u}]\,d\Omega + \int_\Omega q[\nabla C]_2^T\mathbf{u}\,d\Omega + \sum_{e=1}^{n_{ele}}\tau_e\int_{\Omega^e}\nabla q^T\nabla p d\Omega \\
&\quad = \int_{\Gamma_h}\mathbf{w}^T\mathbf{t}\,d\Gamma + \int_\Omega \mathbf{w}^T[\mathbf{C}\rho]\mathbf{f}\,d\Omega - \sum_{e=1}^{n_{ele}}\tau_e\int_{\Omega^e}\nabla q^T\mathbf{f}\,d\Omega
\end{aligned} \tag{30}$$

Remark 2. The present two-phase formulation of a mixture of two incompressible viscous fluids requires stabilization in order to prevent spurious oscillations in the pressure field. This is due to the choice of equal order interpolation functions, analogously to the single-phase formulation presented in [22]. The stabilization terms in Eq. (30) can be identified by the summation over all elements. The stabilization parameter τ_e of element e is given by

$$\tau_e = \frac{1}{\sqrt{\left(\frac{2\bar{\rho}}{\Delta t}\right)^2 + \left(\frac{4\bar{\mu}}{h_e^2}\right)^2}} \tag{31}$$

$\bar{\mu} = C_s\mu_s + C_f\mu_f$ is the averaged viscosity, $\bar{\rho} = C_s\rho_s + C_f\rho_f$ the averaged density, Δt the time step length and h_e a characteristic element length.

Remark 3. In order for the stabilization to be consistent the entire momentum conservation equation has to be included and multiplied by the weighting function ∇q [11]. In practice, however, non-satisfaction of consistency has very little effect on convergence of the solution. We therefore include only the pressure gradient term in our formulation.

[2] The letters s and f identify the phases. In general they are used as subscripts, except when there is already an index. In that case they are written as superscripts (Example: The velocity of the solid phase $\mathbf{u}_s = [u_x^s\ u_y^s]^T$.

[3] $H^1(\Omega)$ is the space of functions u for which the inner product $(u,u)_1 = \int_\Omega (u^2 + (u_{,i})^2)\,d\Omega$ is finite and the norm $\|u\|_1 = \sqrt{(u,u)}$ exists (i denotes the spatial dimensions. $i = 1,2$ for 2D). $L_2(\Omega)$ is the space of functions u for which the inner product $(u,u) = \int_\Omega u^2\,d\Omega$ is finite and the norm $\|u\|_{L_2} = \sqrt{(u,u)}$ exists.

5. Discretization

Discretization uses finite elements in space and finite differences in time. We choose finite elements over other methods, e.g. finite volume methods, for their convenience of computing stresses. This will become important if more complex constitutive models, that take into account grain-to-grain contact in the solid phase, are used.

We use an updated Lagrangian technique for following particles by updating their coordinates at each time step. This technique emerges naturally from the arbitrary Lagrangian–Eulerian approach in [12]. An updated Lagrangian formulation coupled with a meshless interpolation based on a modified Delaunay tesselation and the α-shape algorithm for free surface tracking is called the particle finite element method in [15]. Direct use of Eqs. (21) and (22), in which inertia terms are particle related (Lagrangian) and stresses space related (Eulerian), gives rise to the following algorithm in which particle motions (accelerations and velocities) are computed first and stresses from the gradient of the spatial velocity field are computed consecutively. After each iteration a new mesh is constructed, according to the procedure outlined in the following section, and the nodal variables (velocities, pressure and volume fractions) are mapped onto this new mesh.

6. Time integration scheme

The position $\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta\mathbf{d}_{n+1}$ is the (unknown) position at the end of step $n+1$ at $t = t_{n+1}$ and $\Delta\mathbf{d}_{n+1}$ the displacement increment within the time step.

In the updated Lagrangian formulation $D\mathbf{v}/Dt$ corresponds to the particle acceleration and $\mathbf{v}$ to the particle velocity. Replacing $\mathbf{v}$ and $D\mathbf{v}/Dt$ by their approximations in terms of element shape functions and nodal unknowns leads to:

$$\mathbf{v}_h = \sum N_I \mathbf{v}_I \tag{32}$$

$$\mathbf{a}_h = \sum N_I \frac{D\mathbf{v}_I}{Dt} = \sum N_I \mathbf{a}_I \tag{33}$$

Remark 4. $\mathbf{v}_I$ and $\mathbf{a}_I$ are vectors of nodal velocities and accelerations. The nodal vectors include two vectors, one for each phase, and the nodal pressures:

$$\mathbf{v}_I = [\, v^s_{x,I} \quad v^s_{y,I} \quad v^f_{x,I} \quad v^f_{y,I} \quad p_I \,]^T \tag{34}$$

$$\mathbf{a}_I = \left[\frac{Dv^s_{x,I}}{Dt} \quad \frac{Dv^s_{y,I}}{Dt} \quad \frac{Dv^f_{x,I}}{Dt} \quad \frac{Dv^f_{y,I}}{Dt} \quad 0 \right]^T \tag{35}$$

Invoking arbitrariness of the test functions $\mathbf{w}$ and q allows us to write a global matrix equation, at time t_{n+1}, of the form:

$$\mathbf{M}(\mathbf{x}_{n+1}, C_{n+1})\mathbf{a}_{n+1} + \mathbf{K}(\mathbf{x}_{n+1}, C_{n+1})\mathbf{v}_{n+1} = \mathbf{F}^{ext}_{n+1} \tag{36}$$

Remark 5. The volume fractions C_s and C_f are computed after each iteration. During the iteration they are assumed to be constant.

The left-hand side can be grouped into a non-linear term $\mathbf{N}$. The time stepping iterative scheme is then developed along the lines of Hughes et al. [13]. The generalized trapezoidal algorithm can be stated as:

$$\mathbf{N}(\mathbf{a}_{n+1}, \mathbf{v}_{n+1}, \mathbf{x}_{n+1}, C_{n+1}) = \mathbf{F}^{ext}_{n+1} \tag{37}$$

with

$$\mathbf{a}_{n+1} = \frac{1}{\Delta t\gamma}(\mathbf{v}_{n+1} - \tilde{\mathbf{v}}_{n+1}) \tag{38}$$

$$\Delta\mathbf{d}_{n+1} = \Delta\tilde{\mathbf{d}}_{n+1} + \Delta t^2 \beta \mathbf{a}_{n+1} \tag{39}$$

and the predictors given by

$$\tilde{\mathbf{v}}_{n+1} = \mathbf{v}_n + \Delta t(1-\gamma)\mathbf{a}_n \tag{40}$$

$$\Delta\tilde{\mathbf{d}}_{n+1} = \Delta t\mathbf{v}_n + \frac{\Delta t^2}{2}(1-2\beta)\mathbf{a}_n \tag{41}$$

Now we write the system of equations in terms of increments of velocities. Linearization of $\mathbf{N}$ at iteration i yields

$$\frac{\partial \mathbf{N}(\mathbf{a}^i_{n+1}, \mathbf{v}^i_{n+1}, \mathbf{x}^i_{n+1}, C^i_{n+1})}{\partial \mathbf{v}^i_{n+1}} \Delta\mathbf{v} = \mathbf{F}^{ext}_{n+1} - \mathbf{N}(\mathbf{a}^i_{n+1}, \mathbf{v}^i_{n+1}, \mathbf{x}^i_{n+1}, C^i_{n+1}) \tag{42}$$

with

$$\frac{\partial \mathbf{N}(\mathbf{a}^i_{n+1}, \mathbf{v}^i_{n+1}, \mathbf{x}^i_{n+1}, C^i_{n+1})}{\partial \mathbf{v}^i_{n+1}} = \frac{\partial (\mathbf{M}(\mathbf{x}^i_{n+1}, C^i_{n+1})\mathbf{a}^i_{n+1})}{\partial \mathbf{v}^i_{n+1}} + \frac{\partial (\mathbf{K}(\mathbf{x}^i_{n+1}, C^i_{n+1})\mathbf{v}^i_{n+1})}{\partial \mathbf{v}^i_{n+1}} \tag{43}$$

In this work we assume that the two matrices **M** and **K** do not change within an iteration. With this assumption the tangent matrix in Eq. (43) is approximated by

$$\frac{\partial \mathbf{N}(\mathbf{a}^i_{n+1}, \mathbf{v}^i_{n+1}, \mathbf{x}^i_{n+1}, C^i_{n+1})}{\partial \mathbf{v}^i_{n+1}} \approx \mathbf{M}(\mathbf{x}^i_{n+1}, C^i_{n+1}) \frac{1}{\Delta t \gamma} + \mathbf{K}(\mathbf{x}^i_{n+1}, C^i_{n+1}) \tag{44}$$

The algorithm is summarized in Table 1.

6.1. Mesh update

Table 2 summarizes the algorithm of mesh update. The particularity in comparison with a single-phase formulation resides in the fact that the Lagrangian update produces two new, distinct nodes. This is addressed in Section 6.2. Re-meshing is required not only to preserve a good quality mesh, but also in order to maintain the number of nodes approximately constant. Phase velocities are mapped from the deformed mesh of the corresponding phase onto the new mesh. The pressure is mapped from the union of all updated nodes of the two phases onto the new mesh.

6.2. Lagrangian update

The Lagrangian update is performed for the predictor step, applying the displacement increment $\Delta\tilde{\mathbf{d}}_{n+1}$, and for the corrector step, using $\Delta\mathbf{d}^{i+1}_{n+1}$. The update procedure for a general update using an increment $\Delta\mathbf{d}$, updating a coordinate $\mathbf{x}^0$ to $\mathbf{x}^1$, is given next:

Table 1
Generalized trapezoidal algorithm for two-phase formulation.

1. At t_{n+1} do:

   Initialize the iteration counter : $i = 0$
   Predictor phase : $\tilde{\mathbf{v}}_{n+1} = \mathbf{v}_n + \Delta t(1-\gamma)\mathbf{a}_n$

   $$\Delta\tilde{\mathbf{d}}_{n+1} = \Delta t \mathbf{v}_n + \frac{\Delta t^2}{2}(1-2\beta)\mathbf{a}_n$$

   Interior nodes : $\tilde{\mathbf{x}}_{n+1} = \mathbf{x}_n + \Delta\tilde{\mathbf{d}}_{n+1}$
   Boundary nodes : $\tilde{\mathbf{x}}^{mix}_{n+1} = \mathbf{x}^{mix}_n + (C_s\Delta\tilde{\mathbf{d}}^{n+1}_s + C_f\Delta\tilde{\mathbf{d}}^{n+1}_f)$

   $\mathbf{x}^{i=0}_{n+1} = \tilde{\mathbf{x}}_{n+1}$
   $\mathbf{v}^{i=0}_{n+1} = \tilde{\mathbf{v}}_{n+1}$
   $\mathbf{a}^{i=0}_{n+1} = 0$

   - Mesh update
   - Compute volume fractions $C_s(\mathbf{x}^{i+1}_{n+1})$ and $C_f(\mathbf{x}^{i+1}_{n+1})$ (Section 7)

2. Compute the residual force, the tangent stiffness matrix and solve the linear system of equations:

   $$\Delta\mathbf{F} = \mathbf{F}^{ext}_{n+1} - \mathbf{N}(\mathbf{a}^i_{n+1}, \mathbf{v}^i_{n+1}, \mathbf{x}^i_{n+1}, C^i_{n+1})$$
   $$\mathbf{K}^* = \frac{1}{\Delta t\gamma}\mathbf{M}(\mathbf{x}^i_{n+1}, C^i_{n+1}) + \mathbf{K}(\mathbf{x}^i_{n+1}, C^i_{n+1})$$
   $$\mathbf{K}^*\Delta\mathbf{v} = \Delta\mathbf{F}$$

3. Corrector phase:

   $$\mathbf{v}^{i+1}_{n+1} = \mathbf{v}^i_{n+1} + \Delta\mathbf{v}$$
   $$\mathbf{a}^{i+1}_{n+1} = \frac{1}{\Delta t\gamma}(\mathbf{v}^{i+1}_{n+1} - \tilde{\mathbf{v}}_{n+1})$$
   Interior nodes : $\mathbf{x}^{i+1}_{n+1} = \mathbf{x}^i_{n+1} + \Delta\mathbf{d}^{i+1}_{n+1}$
   Boundary nodes : $\mathbf{x}^{mix,i+1}_{n+1} = \mathbf{x}^{mix,i}_{n+1} + (C_s\Delta\mathbf{d}^{s,i+1}_{n+1} + C_f\Delta\mathbf{d}^{f,i+1}_{n+1})$
   $$= \mathbf{x}^{mix,i}_{n+1} + \frac{\Delta t\beta}{\gamma}(C_s\Delta\mathbf{u}_s + C_f\Delta\mathbf{u}_f)$$

   - Mesh update
   - Compute volume fractions $C_s(\mathbf{x}^{i+1}_{n+1})$ and $C_f(\mathbf{x}^{i+1}_{n+1})$ (Section 7)

4. Test if computation has converged: If $|\Delta\mathbf{F}| < C \in \mathbb{R}$, go to 1. (step $n = n+1$). Else go to 2. (iteration $i = i+1$)

Table 2
Mesh update algorithm for two-phase formulation.

1. Find the boundary of the fluid, using the α-shape method (see Edelsbrunner et al. [7] for a description of the method and CGAL [1] for an implementation into a computational geometry library)
2. Re-mesh inside the boundary
3. Re-map the nodal variables on the new mesh

$$\mathbf{x}_s^1 = \mathbf{x}^0 + \Delta\mathbf{d}_s \tag{45}$$
$$\mathbf{x}_f^1 = \mathbf{x}^0 + \Delta\mathbf{d}_f \tag{46}$$

$\mathbf{x}^0$ is the previous spatial coordinate of the node, which is identical for both phases, and $\mathbf{x}_s^1$ and $\mathbf{x}_f^1$ are the new spatial coordinates of the solid and the fluid material points.

For nodes that are part of the boundary before the Lagrangian update a slightly different update strategy is adopted. A unique displacement increment for both phases, given by the displacement increment of the mixture $\Delta\mathbf{d}_{mix}$, is applied to the material points of the solid and the fluid phase in order to update the boundary (Eqs. (47) and (48), Fig. 1(a)).

$$\Delta\mathbf{d}_{mix} = C_s\Delta\mathbf{d}_s + C_f\Delta\mathbf{d}_f \tag{47}$$
$$\mathbf{x}_s^1 = \mathbf{x}_f^1 = \mathbf{x}^0 + \Delta\mathbf{d}_{mix} \tag{48}$$

Remark 6. Different approaches have been examined that relax the hypothesis of presence of both phases for the benefit of applying the real displacements of the phases, according to Fig. 1(b). This brings up several problems. The updated configuration is not consistent with the assumption that the volume fractions remain approximately constant during an iteration. Since the velocities at the boundary are obtained by assuming volume fractions between 0 and 1, the single-phase domain after the update should only contain that same volume fraction of the single-phase material. The complement would have to be void in order to conserve mass.

Remark 7. Using the displacement increment of the mixture for the update of the boundary satisfies Eq. (6) and therefore conserves the volume of the mixture

$$\nabla\cdot\mathbf{u}_{mix} = \nabla\cdot(C_s\mathbf{u}_s + C_f\mathbf{u}_f) = \nabla\cdot(C_s\mathbf{u}_s) + \nabla\cdot(C_f\mathbf{u}_f) = 0 \tag{49}$$

Remark 8. Dirichlet boundary conditions on velocities are imposed on nodes that are in contact with a solid boundary. Nodes that are newly brought into contact with a solid boundary are identified as their trajectories penetrate the solid boundary. Such nodes are projected back perpendicularly onto the solid boundary line.

7. Computation of volume fractions

Computation of volume fractions is done at the beginning of each time step and after each iteration, according to the computed velocity fields. The problem can be stated as follows: *Given the set of nodes before the update and the nodal sets of the fluid and the solid phases after the update, volume fractions are to be computed on the new, re-zoned nodal set*. While Eq. (6) ensures conservation of the global volume the algorithm for computing volume fractions is responsible for conserving the mass of each phase (or volume in the case of incompressible media). Furthermore the method should yield smooth volume fraction fields. Different approaches were evaluated in Preisig [22].

Among the methods that we have tried out the one that best respects the above requirements is presented hereafter.

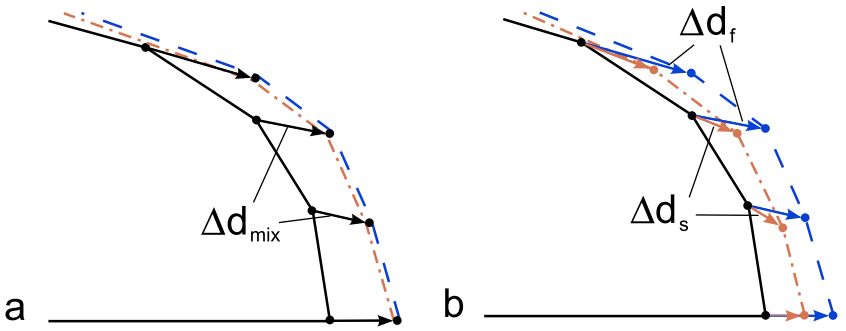

Fig. 1. Lagrangian update of nodes on the boundary: (a) Modified update using displacement increment of mixture $\Delta\mathbf{d}_{mix}$, (b) Separate update of each phase s and f using displacement increments of phases $\Delta\mathbf{d}_s$ and $\Delta\mathbf{d}_f$.

De-coupling of the computation of volume fractions from the computation of velocities and pressure is justified by the assumption that variations of volume fractions during an iteration are small. This is true if the time step size is reasonably small. The volume fractions can therefore be updated a posteriori according to the positions of the nodes while being kept constant during the computation of the primary unknowns. The idea is that the Lagrangian update of the nodes creates a new distribution of phases, where all the information necessary to compute volume fractions is available at the updated nodes. Computation of volume fractions basically consists in evaluating the local density (of volume) of each phase.

$A_I^{0,p}$ denotes the approximation of the local volume fraction of phase p, weighted by the associated volume, at node I before (initial configuration) and $A_I^{1,p}$ after the Lagrangian update (deformed configuration). The local densities $A_I^{0,p}$ and $A_I^{1,p}$ are evaluated using a linear hat-function of radius R, centered at node I:

$$A_I^{0,p} = \sum_J^{N^0} \left(1 - \frac{d_{IJ}^0}{R}\right) C_J^{0,p} V_J^0 \tag{50}$$

$$A_I^{1,p} = \sum_J^{N_p^1} \left(1 - \frac{d_{IJ}^{1,p}}{R}\right) C_J^{0,p} V_J^0 \tag{51}$$

where $C_J^{0,p}$ is the volume fraction of phase p at node J before the update and V_J^0 is the volume attributed to node J, also before the update. The product $C_J^{0,p} V_J^0$ represents the volume of phase p, carried by node J, that is projected from the initial to the updated position. d_{IJ}^0 and $d_{IJ}^{1,p}$ are the distances between nodes I and J in the undeformed and in the deformed configuration of phase p. The summation is performed over all nodes within a radius R around node I, including node I itself. The method is illustrated in Fig. 2.

The approximated local densities are either transported to or computed on the nodes of the deformed old mesh of the corresponding phase. $C_I^{0,s}, A_I^{0,s}$ and $A_I^{1,s}$ have to be interpolated from the deformed solid mesh onto the re-zoned nodes of the new mesh. The same has to be done with $C_I^{0,f}, A_I^{0,f}$ and $A_I^{1,f}$ on the deformed fluid mesh. In each node I of the new mesh the volume fractions are then computed according to

$$C_I^{1,s} = C_I^{0,s} + \frac{A_I^{1,s}}{A_I^{1,s} + A_I^{1,f}} - \frac{A_I^{0,s}}{A_I^{0,s} + A_I^{0,f}} \tag{52}$$

$$C_I^{1,f} = C_I^{0,f} + \frac{A_I^{1,f}}{A_I^{1,s} + A_I^{1,f}} - \frac{A_I^{0,f}}{A_I^{0,s} + A_I^{0,f}} \tag{53}$$

In the last step, the volume fractions are corrected in order to conserve the volumes occupied by each phase, and to enforce $C_s + C_f = 1$. The volumes are corrected by multiplying the volume fractions by the following correction factor, which is defined for the entire domain:

$$\Lambda^p = \frac{\text{mass of phase } p \text{ after deformation}}{\text{initial(exact)mass}} \tag{54}$$

The corrected volume fractions are obtained from:

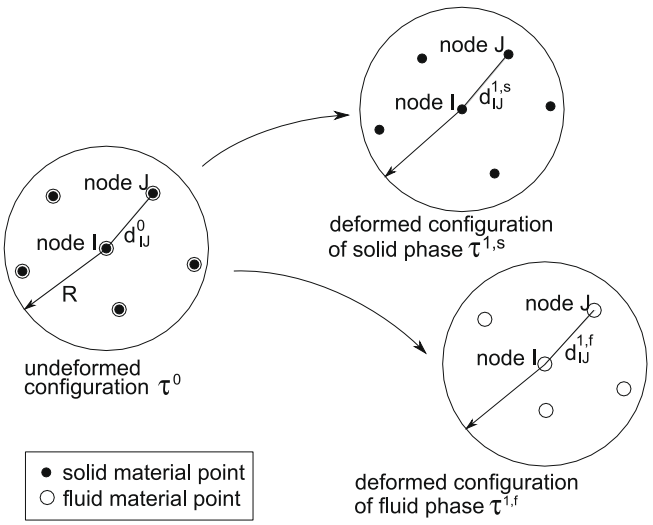

Fig. 2. Illustration of the method for computing volume fractions on the deformed configurations of both phases.

$$C_I^s = \frac{C_I^{1,s}\Lambda^s}{C_I^{1,s}\Lambda^s + C_I^{1,f}\Lambda^f} \tag{55}$$

$$C_I^f = 1 - C_I^s \tag{56}$$

The method is summarized on Table 3.

Remark 9. The radius R in Eqs. (50) and (51) is a parameter that has to be chosen by the user. It has to be small in order to limit numerical diffusion of sharp gradients, but it cannot be too small in order to still produce smooth results. Appropriate values are in the range between $0.6h$ and $0.8h$, where h is the average spacing between nodes. Tests showed that in this range varying the radius R has very little influence on the resulting volume fraction fields.

7.1. Analysis of the performance of the method for computing volume fractions

The method presented above computes volume fractions based on a neighborhood of nodes. In order to examine the influence of the mesh and the length of the time step, thus the displacement, on the resulting volume fraction field, we impose a predefined motion to the nodes of the mesh. In this way we eliminate the physics of two-phase flows from the method and this allows us to focus on the algorithm of computing volume fractions. We perform two tests on the unit squares shown in Fig. 3. In both tests the motion imposed to the nodes of the mesh corresponds to a vortex. The nodes rotate around the origin with an angular velocity of $\omega(r)$, given by

$$\omega(r,t) = \begin{cases} \eta(0.5-r)^2 & \forall\ r < 0.5 \\ 0 & \forall\ r \geqslant 0.5 \end{cases} \tag{57}$$

$$\text{where} \quad \begin{array}{ll} \eta = 1 & \forall\ t \leqslant 20 \\ \eta = -1 & \forall\ t > 20 \end{array}$$

Table 3
Algorithm for computing volume fractions.

1. For each node I of the **old mesh**, do: For each phase $p = s,f$, do:
   1.1 For each node J within a distance R, evaluate distance d_{IJ}^0 to node I in the undeformed configuration and compute $A_I^{0,p} = \sum_J \left(1 - \frac{d_{IJ}^0}{R}\right) C_J^{0,p} V_J^0$
   1.2 For each node J within a distance R, evaluate distance $d_{IJ}^{1,p}$ to node I in the deformed configuration and compute $A_I^{1,p} = \sum_J \left(1 - \frac{d_{IJ}^{1,p}}{R}\right) C_J^{0,p} V_J^0$
   1.3 Interpolate $C_I^{0,p}, A_I^{0,p}$ and $A_I^{1,p}$ onto the new re-zoned mesh
2. For each node I of the **new mesh**, compute new volume fractions (Eqs. (52) and (53))
3. Correct volume fractions (Eqs. (55) and (56))

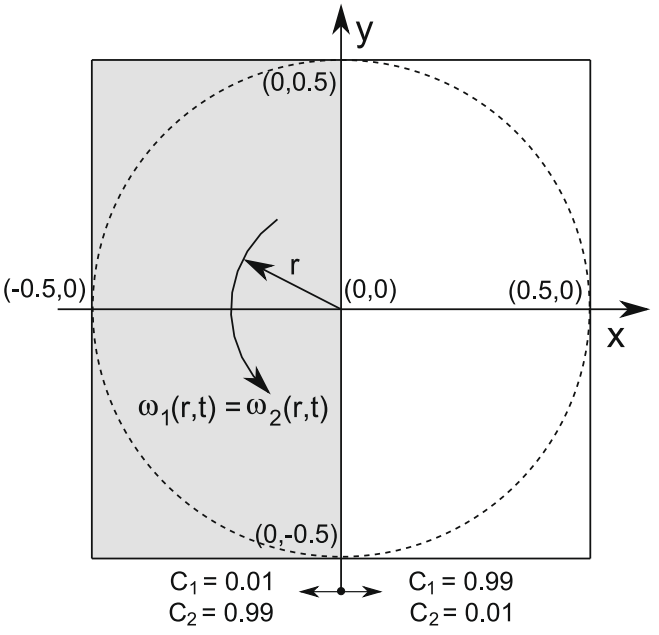

Fig. 3. Analysis of the method for computing volume fractions. Illustration of the performed tests.

where r is the distance from the origin. The new coordinates (x', y') of a node with initial coordinates (x, y) can be computed as

$$\begin{Bmatrix} x' \\ y' \end{Bmatrix} = \begin{bmatrix} \cos(\omega t) & -\sin(\omega t) \\ \sin(\omega t) & \cos(\omega t) \end{bmatrix} \begin{Bmatrix} x \\ y \end{Bmatrix} \tag{58}$$

In this test we use a fixed structured mesh on which the volume fractions are interpolated after each increment of motion. In other words, the re-zoned mesh is always identical with the initial undeformed mesh. The two phases occupy each one half of the domain and are separated by a vertical line.

The vortex motion is imposed in 20 steps of length $\Delta t = 1$, then the motion is reversed for another 20 steps. The final positions of all material points are identical with the initial positions. The goal is to see how well the straight vertical line separating the two phases is preserved during the motion and at the end of the computation. The initial volume fractions are $C_1 = 0.99$ and $C_2 = 0.01$ on the right half and $C_1 = 0.01$ and $C_2 = 0.99$ on the left half. Computations are performed on four meshes consisting of 221, 613, 1301 and 3281 nodes. Fig. 4 shows the volume fractions after 20 steps and after 40 steps. The results show that the accuracy with which the separation between the two regions is captured increases as the mesh is refined. The volume-fraction gradient becomes steeper near the separation while at the same time the contour line $C_1 = C_2 = 0.5$ gets closer to the exact separation line.

8. Implementation

The numerical model is implemented into FEM_object, an object-oriented finite element program in C++ [8]. A description of the structure of the basic code can be found in Commend et al. [4]. For the geometric tasks, most importantly the Delaunay triangulation, the α-shape method and point-in-triangle searches required for interpolating variables from one mesh to another, the Computational Geometry Algorithms Library (CGAL) [1] was used.

9. Numerical applications

9.1. Sedimentation

Sedimentation of a phase of solid particles in a viscous fluid is a problem of great interest for many industrial processes. In the context of debris flows it is an important test for the interaction behavior between the two phases. This test verifies that the drag force term acts as intended. We compare our results with the analytical solution given by Soo [25]. His formulation is slightly different from ours, therefore some adjustments need to be made. In his solution, Soo neglects diffusion. Furthermore, he assumes the density of the fluid phase to be negligible in comparison with the density of the solid phase. Re-writing Soo's formulation of conservation of momentum of the solid and the fluid phase in vertical direction, using our notation and pressure sign convention, and re-arranging terms, yields:

$$C_s \rho_s \frac{Du_s}{Dt} = C_s \frac{\partial p}{\partial z} - C_s \rho_s g + C_s \rho_s F(u_f - u_s) \tag{59}$$

$$0 = C_f \frac{\partial p}{\partial z} - C_s \rho_s F(u_f - u_s) \tag{60}$$

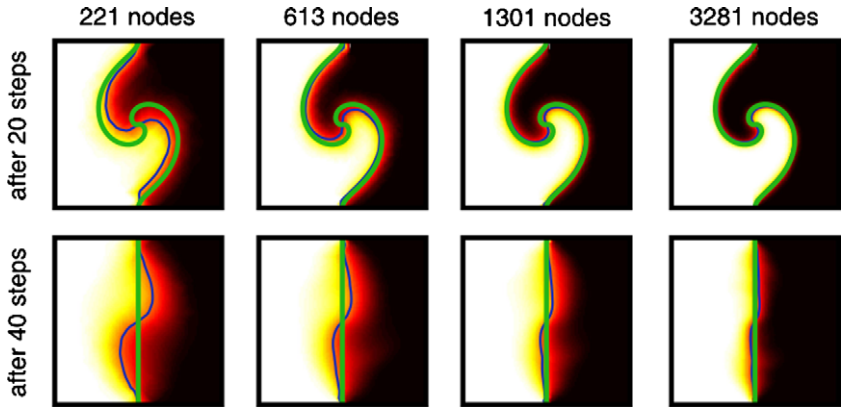

Fig. 4. Distribution of volume fractions for imposed vortex motion. The thick green line indicates the exact solution, the thin blue line indicates the contour line of $C_1 = C_2 = 0.5$ of the numerical result. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The coefficient $F/(1-C_s)^2 = F' = 10$ is assumed to be constant. Setting $\rho_f = 0, \mu_s = \mu_f = 0$, using the same one-dimensional form, Eqs. (21) and (22) become

$$C_s\rho_s \frac{Du_s}{Dt} = \frac{\partial C_s p}{\partial z} - C_s\rho_s g + m_z^{sf} \tag{61}$$

$$0 = \frac{\partial C_f p}{\partial z} - m_z^{sf} \tag{62}$$

Remark 10. For numerical reasons, the viscosity of the fluid phase is set to $\mu_f = 0.01$, which adds an additional viscous term to Eq. (62). Numerical experiments showed that the effect of this term is limited to reducing numerical oscillations.

If we define the drag force coefficient in Eq. (19) as $K'_{drag} = \rho_s C_s (1-C_s)^2 F'$, the equations used by Soo are almost identical with Eqs. (61) and (62). The only remaining difference is the volume-fraction-gradient term $p\nabla C_s$, which is absent in the

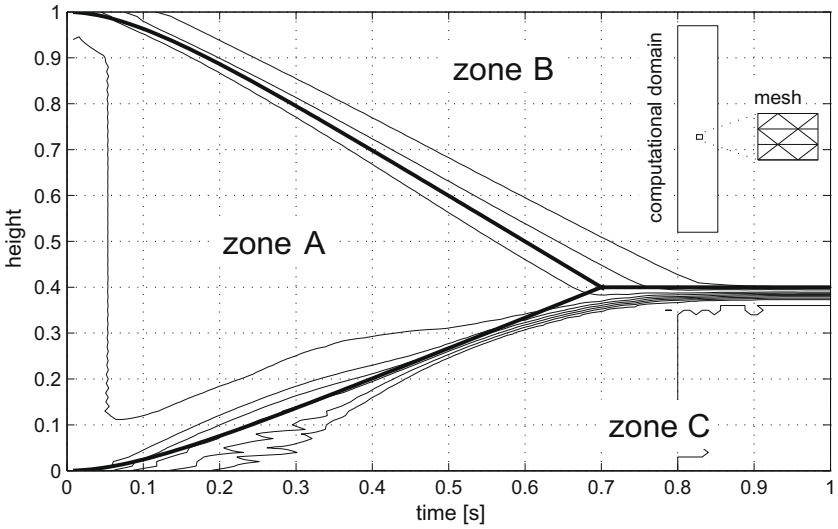

Fig. 5. Contours of solid volume fraction as a function of vertical coordinate and time for the sedimentation problem, using the formulation by Soo. The thick line indicates the delimitation of three separate zones given by the analytical solution: zone A: $C_s = 0.2$, zone B: $C_s = 0$, zone C: $C_s = 0.5$.

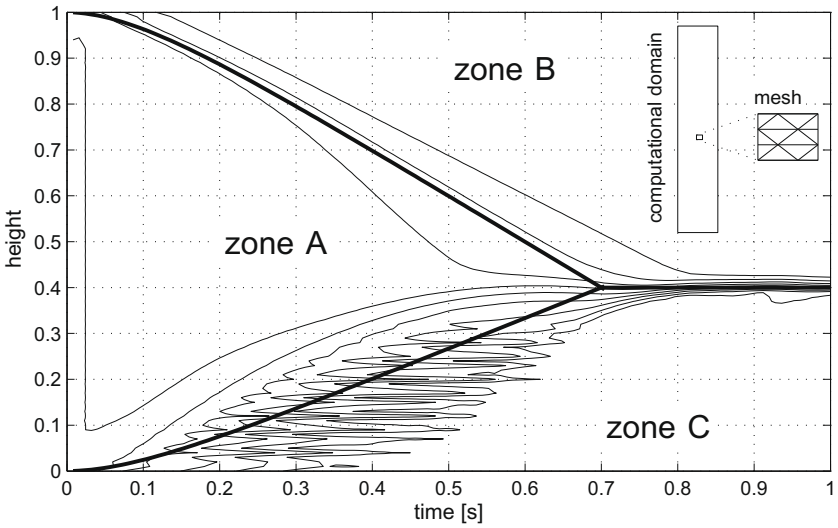

Fig. 6. Contours of solid volume fraction as a function of vertical coordinate and time for the sedimentation problem, using the original formulation with the $p\nabla C_s$-term (Eqs. (61) and (62)). The thick line indicates the delimitation of three separate zones given by the analytical solution: zone A: $C_s = 0.2$, zone B: $C_s = 0$, zone C: $C_s = 0.5$.

model by Soo. This term results from applying mixture theory and acts as a force against the separation of the phases. In order to verify the implementation we perform the computation with and without the volume-fraction gradient.

The analysis is performed on a rectangular container of height $h = 1$ and width $d = 0.15$. The geometry as well as a zoom on the structure of the mesh are given in Figs. 5 and 6. At time $t = 0$ the volume is filled with a homogeneous mixture with a solid volume fraction of $C_s = 0.2$ and intrinsic solid phase density $\rho_s = 10$. The maximum volume fraction is limited to $C_s^{max} = 0.5$, which represents the void ratio of a granular material after sedimentation is completed. In order to stop the sedimentation process when C_s^{max} is reached we increase the viscosities linearly from 0 to 10^6 as the solid volume fraction increases from 0.48 to 0.5. The vertical gravitational acceleration is $g = 10$.

Remark 11. Increasing the viscosities when C_s^{max} is reached can be considered a simple form of a non-Newtonian constitutive model. In this case the constitutive relations (Eqs. (12) and (13)) are modified as

$$\tau(\mathbf{u}_s) = 2\mu_s(C_s)\left(\epsilon(\mathbf{u}_s) - \frac{1}{3}(\nabla\cdot\mathbf{u}_s)\mathbf{I}\right)$$
$$\tau(\mathbf{u}_f) = 2\mu_f(C_s)\left(\epsilon(\mathbf{u}_f) - \frac{1}{3}(\nabla\cdot\mathbf{u}_f)\mathbf{I}\right)$$

where $\mu_s(C_s)$ and $\mu_f(C_s)$ are volume fraction dependent viscosities.

The minimum solid volume fraction is limited to $C_s^{min} = 0.01$.

In Fig. 5 we show a comparison of the analytical results by Soo with our results. The contour plot shows the solid volume fraction along the vertical axis, versus time. The results match very well. The time for complete separation of phases is reproduced almost exactly. In the numerical results the separation between the zones A, B and C, where the analytical solution assumes constant volume fractions, is more diffuse due to physical diffusion, but also due to some numerical diffusion. Still three areas of almost constant volume fraction can clearly been identified.

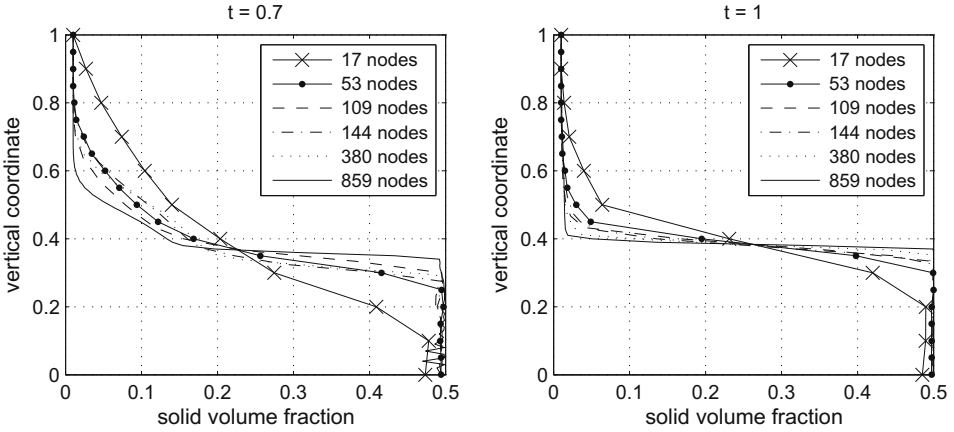

Fig. 7. Vertical profiles of solid volume fraction at $t = 0.7$ and $t = 1$. Formulation without volume-fraction-gradient term (Soo's formulation).

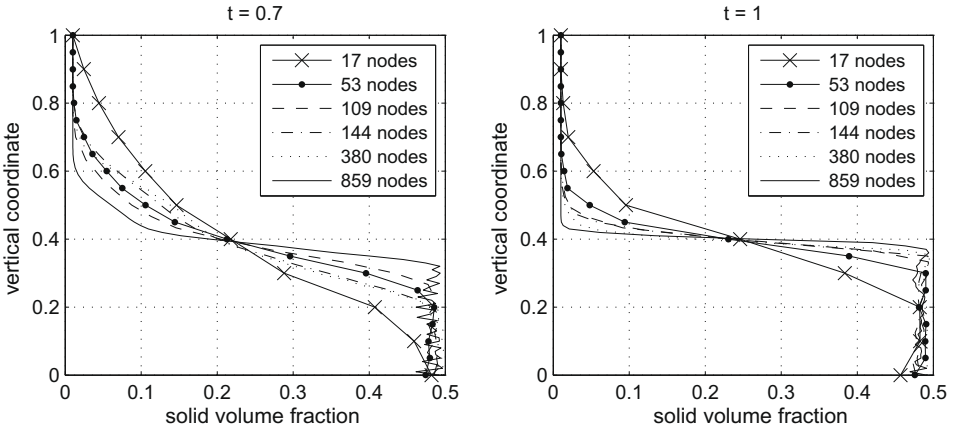

Fig. 8. Vertical profiles of solid volume fraction at $t = 0.7$ and $t = 1$. Formulation with $p\nabla C_s$-term (Eqs. (61) and (62)).

For comparison we also show the result using the original formulation (Eqs. (61) and (62)), including the $p\nabla C_s$-term (Fig. 6). The main difference between the two results is in the separation between zones A and C, where the formulation omitting the $p\nabla C_s$-term has a much sharper interface. We also note that including the $p\nabla C_s$-term can cause oscillations in the volume fraction field.

In order to show that the separation between the two phases becomes sharper as the mesh is refined we plot the vertical profile of solid volume fractions in Fig. 7. We use six different meshes with 17 nodes in the coarsest mesh and 859 in the finest. The spacing of nodes in the vertical direction was 0.2, 0.1, 0.0667, 0.05, 0.0294 and 0.02 for the finest mesh. The time steps were also adapted to the mesh size in such a way that the maximum differential displacement per time step, divided by the vertical mesh spacing, $|u_s - u_f|\Delta t/\Delta h$, does not exceed a value of 0.9. From the coarsest to the finest mesh, Δt was chosen to be 0.08, 0.04, 0.025, 0.02, 0.0125 and 0.008. The profile is given at $t = 0.7$ and $t = 2$. Fig. 7 shows that on sufficiently fine meshes very steep gradients can be reproduced. The corresponding result in Fig. 8, this time including the $p\nabla C_s$-term shows a very similar behavior. The presence of strong gradients of volume fraction in this formulation can, however, lead to instabilities, as can be seen in the plot for $t = 2$. This gradient term appears from the definition of the pressure in each phase (Eqs. (10) and (11)). It is sometimes neglected (see Hutter [14] for a discussion).

9.2. Mudflow impacting an obstacle

The simulation of the downhill propagation of a two-phase mixture and its impact on an obstacle is analyzed next. The obstacle, representing a protection dam, is modeled as a solid block that is placed at the bottom of a slope. The geometry, together with the mesh at time $t = 0$ is given in Fig. 9. The flow is initiated by the sudden release of a homogeneous two-phase mixture under the gravitational acceleration $\mathbf{f} = [0 \quad -10]^T$. Material parameters together with details of the discretization are given on Table 4. For comparison the same problem is simulated with a single-phase fluid, using the average material properties of the two-phase mixture.

The shape of the two-phase fluid during the event is shown in Fig. 10. Colors indicate solid volume fractions. The solid phase initially accumulates at the base, while the fluid phase stays on the surface and accumulates at the front of the flow. After the flow tip reaches the obstacle the solid phase quickly catches up with the faster flowing fluid phase, filling the space behind the barrier. The tip of the mixture shooting over the barrier is essentially fluid, due to lower viscosity and density. At the end of the simulation, that is after 1000 time steps, the total volume of the mixture has increased by 2.3%. About half of this error results from the relatively simple contact algorithm used in the model. We consider this error acceptable.

From the simulation the resultant force acting on an obstacle that obstructs the flow path is extracted. In Fig. 11 the resultant force of the single-phase and the two-phase model are compared in a global view (top) as well as in a zoom on the time

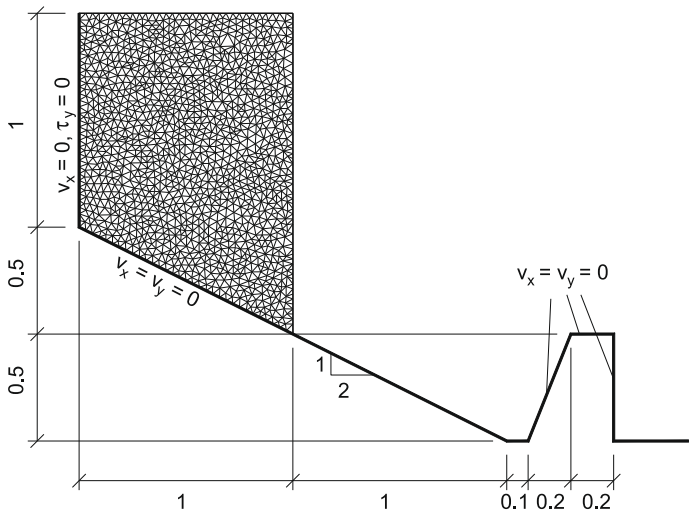

Fig. 9. Geometry and initial mesh used in the simulation of a mudflow impacting on an obstacle.

Table 4
Parameters of the simulation of a mudflow impacting on an obstacle.

Model	ρ_s	ρ_f	μ_s	μ_f	K_{drag}	C_s^{init}	Δt	# of nodes
Two-phase	1000	500	100	2	10′000	0.5	0.002	≈1550
Single-phase	$\rho = 750$		$\mu = 51$		–	–	0.002	≈1550

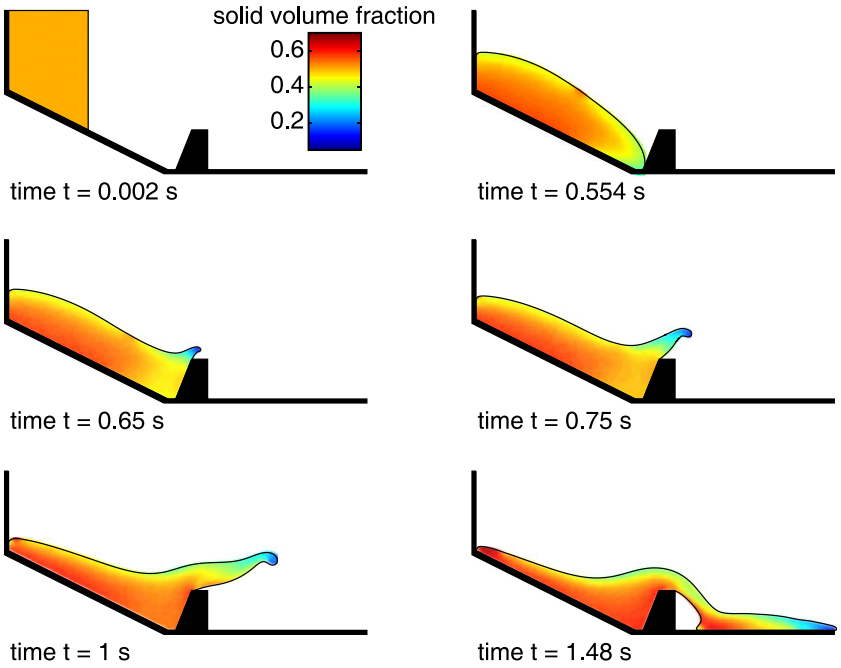

Fig. 10. Solid volume fractions on a mudflow impacting on obstacle.

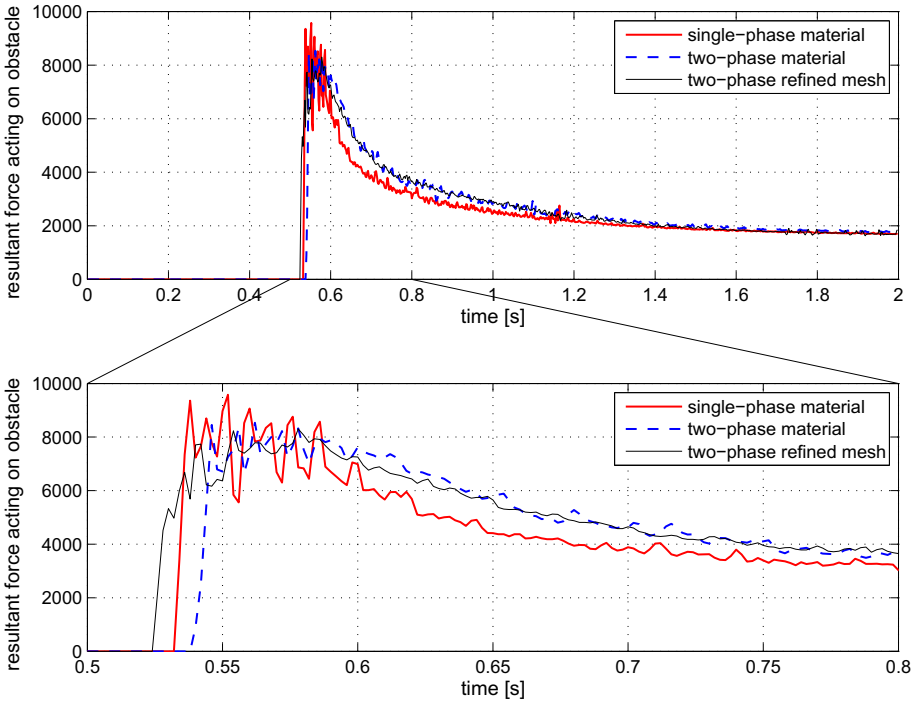

Fig. 11. Smoothed resultant force acting on obstacle. Above: global view; below: zoom on the impact.

when the impact occurs (bottom). The result of a computation using a finer mesh is also presented. The force is computed by integrating the pressure along the front side of the obstacle. Right after the impact the force attains its peak, before it slowly decays to the hydrostatic level. The peak force immediately after the impact is about the same in both cases. Then, the force decays faster in the single-phase case. This is due to the solid phase accumulating behind the obstacle in the two-phase sim-

ulation. The difference between the coarse and the fine mesh is very small, the predicted maximum force acting on the obstacle is almost identical.

10. Conclusions

In this paper we develop a method for modeling free-surface flow of a two-phase material. The two-phase material is modeled as a continuous mixture, where the presence of the individual phases is tracked by volume fractions. The method uses an updated Lagrangian approach to convect the masses of the two phases. A remeshing strategy ensures the regularity of the mesh. The approach consisting in the computation of two sets of phase velocities on the same mesh, a Lagrangian update resulting in two convected meshes, a remesh and remap step as well as the method for computing volume fractions are the core for a general algorithmic framework for the modeling of two-phase flows.

The very simple constitutive model consisting of a mixture of two viscous fluids is validated on a test simulating sedimentation of a denser phase within a less dense phase. The results very closely match an analytical solution. As an application to mudflow simulation the impact of a two-phase mixture on an obstacle is shown and forces acting on the obstacle are extracted. The numerical results clearly show the potential of the approach for a wide range of geophysical and industrial processes.

In order to extend the range of applications of the method, further work on a more sophisticated constitutive model is required. The inclusion of a solid, intergranular pressure will allow the simulation of initiation of the flow, erosion along the propagation path as well as deposition at the base of the slope. The algorithmic framework permits the inclusion of any constitutive model without major changes.

We see a lot of potential in including mesh adaptivity to the method. Increasing the density of nodes in areas of high volume fraction gradients will make it possible to limit numerical diffusion and greatly increase the accuracy of the method.

Finally, extending the method to three spatial dimensions is a crucial element towards making the method available for specific engineering applications. All components of the presented method are extensible to three dimensions. The library used for all meshing tasks, CGAL [1], supports triangulation as well as boundary detection using α-shapes in three dimensions.

Acknowledgments

M. Preisig was supported for this work by the Swiss National Science Foundation under Grants Nos. 2100-067954 and 200020-113295. We also gratefully acknowledge D. Eyheramendy, T.J.R. Hughes, M. Picasso, B. Jeremic and T. Tezduyar for their helpful remarks and suggestions.

Appendix A. Volume averaging of governing equations of two-phase flow

The following derivation shows exactly what assumptions are made if we go from a model with discrete interfaces between the phases to a smeared model where in each point both phases are present. In the process we assume two distinct phases with well defined interfaces to be blended in a mixture in such a way that the phase interfaces are scattered evenly through the domain. We follow the procedure of volume averaging as it is laid out in the book by Soo [25].

The operation of volume averaging is indicated by triangular brackets $\langle\cdot\rangle$. Volume averaging can be applied to any scalar, vector or tensor quantity ϕ_p belonging to phase p:

$$\langle\phi_p\rangle = \frac{1}{V}\int_{V_p}\phi_p\,dV = C_p\langle\phi_p\rangle^i \tag{63}$$

V_p is the volume in V that is occupied by phase p, C_p the volume fraction of phase p and V denotes the control volume. The intrinsic average $\langle\phi_p\rangle^i$ is:

$$\langle\phi_p\rangle^i = \frac{1}{V_p}\int_{V_p}\phi_p\,dV \tag{64}$$

Volume averages of derivatives of a quantity ϕ_p are given by the general transport theorem (Gray [10]). The following derivatives of ϕ_p are used:

$$\langle\frac{\partial\phi_p}{\partial t}\rangle = \frac{\partial\langle\phi_p\rangle}{\partial t} - \frac{1}{V}\int_\Gamma \phi_p\mathbf{u}^\Gamma\cdot\mathbf{n}_p\,dS \tag{65}$$

$$\langle\nabla\phi_p\rangle = \nabla\langle\phi_p\rangle + \frac{1}{V}\int_\Gamma \phi_p\mathbf{n}_p\,dS \tag{66}$$

$$\langle\nabla\cdot\boldsymbol{\phi}_p\rangle = \nabla\cdot\langle\boldsymbol{\phi}_p\rangle + \frac{1}{V}\int_\Gamma \boldsymbol{\phi}_p\cdot\mathbf{n}_p\,dS \tag{67}$$

Γ is the interface between phases. $\mathbf{u}^\Gamma$ is the velocity of the interface and $\mathbf{n}_p$ defines the unit normal vector pointing outward from phase p.

Averages of products are expressed as the product of an averaged value with the intrinsic average of the other value:

$$\langle \phi_p \psi_p \rangle = \langle \phi_p \rangle \langle \psi_p \rangle^i \tag{68}$$

The volume fraction of phase p can be written as a limit, as the control volume V tends to zero:

$$C_p = \lim_{V \to 0} \frac{1}{V} \int_{V_p \subset V} 1 \, dV \tag{69}$$

With these relations we can establish the governing equations of two-phase flow.

A.1. Conservation of mass

Volume averaging of the equation of conservation of the mass of phase p and application of the rule for averages of products leads to:

$$\begin{aligned} &\left\langle \frac{\partial \rho_p}{\partial t} \right\rangle + \langle \nabla \cdot (\rho_p \mathbf{u}_p) \rangle = 0 \\ &\frac{\partial \langle \rho_p \rangle}{\partial t} + \nabla \cdot (\langle \rho_p \rangle^i \langle \mathbf{u}_p \rangle) = -\frac{1}{V} \int_\Gamma \rho_p (\mathbf{u}_p - \mathbf{u}^\Gamma) \cdot \mathbf{n}_p dS \end{aligned} \tag{70}$$

The integral on the right-hand side represents the rate of mass generation per unit volume of phase p. Since in our model no mass exchange occurs between the two phases, this integral is equal to zero. If we let the control volume V tend to zero as in Eq. (69) we obtain

$$\frac{\partial C_p \rho_p}{\partial t} + \nabla \cdot (C_p \rho_p \mathbf{u}_p) = 0 \tag{71}$$

or, in a reference frame attached to the material

$$\frac{D C_p \rho_p}{Dt} + C_p \rho_p \nabla \cdot \mathbf{u}_p = 0 \tag{72}$$

A.2. Conservation of momentum

Volume averaging of the equation of single-phase conservation of momentum over phase p yields, after using mass conservation:

$$\left\langle \rho_p \frac{D \mathbf{u}_p}{Dt} \right\rangle = \langle \nabla \cdot \sigma_p \rangle + \langle \rho_p \mathbf{f} \rangle \tag{73}$$

Expanding the inertial term as a product of averages

$$\left\langle \rho_p \frac{D \mathbf{u}_p}{Dt} \right\rangle = \langle \rho_p \rangle^i \left\langle \frac{D \mathbf{u}_p}{Dt} \right\rangle \tag{74}$$

In order to use Eq. (65) we need to expand the total derivative

$$\begin{aligned} \left\langle \frac{D \mathbf{u}_p}{Dt} \right\rangle &= \left\langle \frac{\partial \mathbf{u}_p}{\partial t} \right\rangle + \langle \mathbf{u}_p (\nabla \cdot \mathbf{u}_p) \rangle \\ &= \frac{\partial \langle \mathbf{u}_p \rangle}{\partial t} - \frac{1}{V} \int_\Gamma \mathbf{u}_p (\mathbf{u}^\Gamma \cdot \mathbf{n}_p) dS \qquad (75) \\ &\quad + \langle \mathbf{u}_p \rangle^i \langle \nabla \cdot \mathbf{u}_p \rangle \qquad (76) \\ &= \frac{\partial \langle \mathbf{u}_p \rangle^i}{\partial t} - \frac{1}{V} \int_\Gamma \mathbf{u}_p (\mathbf{u}^\Gamma \cdot \mathbf{n}_p) dS \\ &\quad + \langle \mathbf{u}_p \rangle^i \left(\nabla \cdot \langle \mathbf{u}_p \rangle + \frac{1}{V} \int_\Gamma \mathbf{u}_p \cdot \mathbf{n}_p dS \right) \qquad (77) \end{aligned}$$

Since $\langle \mathbf{u}_p \rangle^i$ is the convective velocity of phase p, the total derivative of $\langle \mathbf{u}_p \rangle^i$ can be identified from Eq. (77) as:

$$\left\langle \frac{D \mathbf{u}_p}{Dt} \right\rangle = \frac{D \langle \mathbf{u}_p \rangle}{Dt} - \frac{1}{V} \int_\Gamma \mathbf{u}_p (\mathbf{u}^\Gamma \cdot \mathbf{n}_p) dS + \langle \mathbf{u}_p \rangle^i \frac{1}{V} \int_\Gamma \mathbf{u}_p \cdot \mathbf{n}_p \, dS \tag{78}$$

As we let the control volume V, and with it the volume $V_p \subset V$, tend to zero the intrinsic average phase velocity $\langle \mathbf{u}_p \rangle^i$ is equal to $\mathbf{u}_p$ and the average phase velocity $\langle \mathbf{u}_p \rangle$ can be written in terms of volume fractions. The surface integrals can then be combined:

$$\left\langle \frac{D\mathbf{u}_p}{Dt} \right\rangle = \frac{DC_p\mathbf{u}_p}{Dt} - \frac{1}{V}\int_\Gamma \mathbf{u}_p((\mathbf{u}^\Gamma - \mathbf{u}_p)\cdot\mathbf{n}_p)dS \tag{79}$$

As for the conservation of mass, the integral on the right-hand side reduces to zero because no mass is exchanged between phases.

The volume average of the stress divergence is given by the following expression (see Eq. (67)):

$$\langle \nabla\cdot\sigma_p\rangle = \nabla\cdot\langle\sigma_p\rangle + \frac{1}{V}\int_\Gamma \sigma_p\cdot\mathbf{n}_p\,dS \tag{80}$$

Finally, if we use Eq. (72) on the inertial term, the conservation of momentum of phase p can be written as:

$$C_p\rho_p\frac{D\mathbf{u}_p}{Dt} = \nabla\cdot\langle\sigma_p\rangle + C_p\rho_p\mathbf{f} + \frac{1}{V}\int_\Gamma \sigma_p\cdot\mathbf{n}_p\,dS \tag{81}$$

The surface integral represents the exchange of momentum between the two phases.

Appendix B. Element matrices and arrays

$$\mathbf{v}_h = \mathbf{N}_I\mathbf{v}_I$$
$$\epsilon(\mathbf{v}_h) = \mathbf{B}_I\mathbf{v}_I$$
$$[\nabla\cdot\mathbf{w}_hC] = (\mathbf{B}_I\mathbf{w}_I)^T[\mathbf{C}]_2$$
$$[C\nabla\cdot\mathbf{v}_h] = [\mathbf{C}]_2(\mathbf{B}_I\mathbf{v}_I)$$
$$q_h = N_Iq_I$$
$$\nabla q_h = \nabla N_Iq_I$$
$$\mathbf{v}_I = \begin{bmatrix} v_x^{s,I} & v_y^{s,I} & v_x^{f,I} & v_y^{f,I}\end{bmatrix}^T$$
$$\mathbf{N}_I = \begin{bmatrix} N_I & 0 & 0 & 0\\ 0 & N_I & 0 & 0\\ 0 & 0 & N_I & 0\\ 0 & 0 & 0 & N_I\end{bmatrix}$$
$$\mathbf{B}_I = \begin{bmatrix} N_{I,x} & 0 & N_{I,y} & 0 & 0 & 0\\ 0 & N_{I,y} & N_{I,x} & 0 & 0 & 0\\ 0 & 0 & 0 & N_{I,x} & 0 & N_{I,y}\\ 0 & 0 & 0 & 0 & N_{I,y} & N_{I,x}\end{bmatrix}^T$$
$$\nabla N_I = [N_{I,x}\quad N_{I,y}\quad N_{I,x}\quad N_{I,y}]^T$$
$$\overline{\mathbf{D}} = \begin{bmatrix} 4/3\,\mu_s & -2/3\,\mu_s & 0 & 0 & 0 & 0\\ -2/3\,\mu_s & 4/3\,\mu_s & 0 & 0 & 0 & 0\\ 0 & 0 & \mu_s & 0 & 0 & 0\\ 0 & 0 & 0 & 4/3\,\mu_f & -2/3\,\mu_f & 0\\ 0 & 0 & 0 & -2/3\,\mu_f & 4/3\,\mu_f & 0\\ 0 & 0 & 0 & 0 & 0 & \mu_f\end{bmatrix}$$
$$\mathbf{K}_{drag} = K_{drag}\begin{bmatrix} 1 & 0 & -1 & 0\\ 0 & 1 & 0 & -1\\ -1 & 0 & 1 & 0\\ 0 & -1 & 0 & 1\end{bmatrix}$$
$$[\mathbf{C}\boldsymbol{\rho}] = \begin{bmatrix} C_s\rho_s & 0 & 0 & 0\\ 0 & C_s\rho_s & 0 & 0\\ 0 & 0 & C_f\rho_f & 0\\ 0 & 0 & 0 & C_f\rho_f\end{bmatrix}$$
$$[\nabla\mathbf{C}]_1 = \begin{bmatrix} C_{s,x} & 0 & C_{s,y} & 0 & 0 & 0\\ 0 & C_{s,y} & C_{s,x} & 0 & 0 & 0\\ 0 & 0 & 0 & C_{f,x} & 0 & C_{f,y}\\ 0 & 0 & 0 & 0 & C_{f,y} & C_{f,x}\end{bmatrix}$$
$$[\nabla\mathbf{C}]_2 = [C_{s,y}\quad C_{s,x}\quad C_{f,y}\quad C_{f,x}]^T$$

$$[\mathbf{C}]_1 = \begin{bmatrix} C_s & 0 & 0 & 0 & 0 & 0 \\ 0 & C_s & 0 & 0 & 0 & 0 \\ 0 & 0 & C_s & 0 & 0 & 0 \\ 0 & 0 & 0 & C_f & 0 & 0 \\ 0 & 0 & 0 & 0 & C_f & 0 \\ 0 & 0 & 0 & 0 & 0 & C_f \end{bmatrix}$$

$$[\mathbf{C}]_2 = [\, C_s \quad C_s \quad 0 \quad C_f \quad C_f \quad 0 \,]^T$$

References

[1] Cgal, Computational Geometry Algorithms Library. <http://www.cgal.org>.
[2] C. Chen, General solutions for viscoplastic debris flow, Journal of Hydraulic Engineering 114 (3) (1988) 259–282.
[3] C. Chen, Generalized viscoplastic modeling of debris flow, Journal of Hydraulic Engineering 114 (3) (1988) 237–258.
[4] S. Commend, Th. Zimmermann, Object-oriented nonlinear finite element programming: a primer, Advances in Engineering Software 32 (8) (2001) 611–628.
[5] V. de la Cruz, T.J.T. Spanos, Mobilization of oil ganglia, AIChE Journal 29 (5) (1983) 854–858.
[6] R.P. Denlinger, R.M. Iverson, Flow of variably fluidized granular masses across 3-d terrain: 2. Numerical predictions and experimental tests, Journal of Geophysical Research 106 (B1) (2001) 553–566.
[7] H. Edelsbrunner, D.G. Kirkpatrick, R. Seidel, On the shape of a set of points in the plane, IEEE Transactions on Information Theory IT-29 (4) (1983) 551–559.
[8] FEM_Object, an object-oriented finite element package. <http://www.zace.com/femobj.htm>.
[9] R. Frenette, Th. Zimmermann, D. Eyheramendy, Debris-flows project report. Technical Report LSC-DGC Rep. # 98/9, Swiss Federal Institute of Technology, Lausanne, Switzerland, December 1998.
[10] W.G. Gray, A derivation of the equations for multi-phase transport, Chemical Engineering Science 30 (2) (1975) 229–233.
[11] T.J.R. Hughes, L.P. Franca, M. Balestra, A new finite-element formulation for computational fluid-dynamics: V. Circumventing the Babuska-Brezzi condition – A stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations, Computer Methods in Applied Mechanics and Engineering 59 (1) (1986) 85–99.
[12] T.J.R. Hughes, W.K. Liu, Th. Zimmermann, Lagrangian–Eulerian finite element formulation for incompressible viscous flows, Computer Methods in Applied Mechanics and Engineering 29 (1981) 329–349.
[13] T.J.R. Hughes, K.S. Pister, R.L. Taylor, Implicit–explicit finite elements in nonlinear transient analysis, Computer Methods in Applied Mechanics and Engineering 17/18 (1979) 159–182.
[14] K. Hutter, Debris and mudflows: are we asking the correct questions? What are the deficits?, Mitteilungen der Versuchsanstalt für Wasserbau Hydrologie und Glaziologie an der Eidgenössischen Technischen Hochschule Zürich 190 (2005) 91–105.
[15] S.R. Idelsohn, E. Oñate, F. Del Pin, The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves, International Journal for Numerical Methods in Engineering 61 (2004) 964–989.
[16] R.M. Iverson, R.P. Denlinger, Flow of variably fluidized granular masses across 3-d terrain: 1. Coulomb mixture theory, Journal of Geophysical Research 106 (B1) (2001) 537–552.
[17] N.I. Kolev, Multiphase Flow Dynamics, second ed., Springer, Berlin, 2005.
[18] D. Laigle, P. Lachamp, M. Naaim, SPH-based numerical investigation of mudflow and other complex fluid flow interactions with structures, Computational Geosciences 11 (4) (2007) 297–306.
[19] J.S. O'Brien, P.Y. Julien, W.T. Fullerton, Two-dimensional water flood and mudflow simulation, Journal of Hydraulic Engineering – ASCE 119 (2) (1993) 244–261.
[20] M. Preisig, Th. Zimmermann, Landslide simulation using natural neighbor based meshless method, in: Seventh World Congress on Computational Mechanics, Los Angeles, July 2006.
[21] M. Preisig, Th. Zimmermann, A lagrangian method for two-phase flow simulation, in: Ninth US National Congress on Computational Mechanics, San Francisco, July 2007.
[22] M. Preisig, Modeling two-phase flows on moving domains. Phd thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, 2008. Thesis no. 4250.
[23] S.B. Savage, K. Hutter, The motion of a finite mass of granular material down a rough incline, Journal of Fluid Mechanics 199 (1989) 177–215.
[24] S. Shao, E.Y.M. Lo, Incompressible SPH method for simulating newtonian and non-newtonian flows with a free surface, Advances in Water Resources 26 (7) (2003) 787–800.
[25] S.L. Soo, Multiphase Fluid Dynamics, Science Press, Beijing, China, 1990.
[26] B. Svendsen, K. Hutter, On the thermodynamics of a mixture of isotropic materials with constraints, International Journal of Engineering Science 33 (14) (1995) 2021–2054.
[27] T. Takahashi, Debris flow, A.A. Balkema, IAHR Monograph Series, Rotterdam, Brookfield, 1991.