Computers and Structures 87 (2009) 962–969

Contents lists available at ScienceDirect

Computers and Structures

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

Modeling of fresh concrete flow

B. Patzák *, Z. Bittnar

Faculty of Civil Engineering, Czech Technical University, Thákurova 7, 166 29 Prague, Czech Republic

ARTICLE INFO

Article history:
Received 26 October 2006
Accepted 29 April 2008
Available online 16 June 2008

Keywords:
Fresh concrete flow
Non-newtonian flow
Interface-capturing

ABSTRACT

This paper focuses on the modeling of fresh concrete flow. Concrete that is not properly casted or consolidated may have defects, such as air voids, honeycombs, and aggregate segregation. The modeling of fresh concrete flow can significantly contribute to the durability and strength of a structure and it is necessary for design optimization of casting procedure. The fresh concrete is considered as a non-Newtonian fluid. The Bingham model is used as constitutive model, with the yield stress and plastic viscosity as parameters. An interface-capturing approach is used to track the position of a free surface.

1. Introduction

The modeling of flow of freshly mixed concrete is very important for the construction industry because concrete is usually put into place in its plastic form. In the construction field, subjective terms like workability, flow-ability, and cohesion are used, sometimes interchangeably, to describe the behavior and flow properties of fresh concrete. These factors depend on flow (rheological) properties of concrete, which have direct influence on the strength and durability of concrete. Concrete that is not properly casted or consolidated may have defects, such as air voids, honeycombs, and aggregate segregation. The modeling of fresh concrete flow can significantly contribute to the durability and strength of a structure and it is necessary for design optimization of casting procedure. The aim of this contribution is to show that the reliable numerical modeling of fresh concrete flow is feasible, provided that suitable algorithms and constitutive models are combined together. The focus is on algorithms that can be combined with existing engineering codes used in structural or fluid mechanics. Particularly, the volume of fluid (VOF) based interface tracking algorithm is relatively simple and easy to implement into existing codes. The computational model developed is compared to experimental results in axisymmetric and two-dimensional settings to illustrate its performance and capabilities.

The fresh concrete is considered as a fluid. This assumption is valid when a certain degree of flow can be achieved and when concrete is homogeneous. This is usually satisfied, because concrete is put in place in its plastic form in majority of industrial applications. It is widely recognized, that concentrated suspensions, such as concrete, typically behave as non-Newtonian fluids. The constitutive equations that have a physical basis should include at least two parameters, one being the yield stress. The Bingham model is considered with the yield stress and plastic viscosity as parameters. All constitutive parameters are assumed to be a constant in this study. More refined model should be based on parameters dependent on time, or better on the degree of hydration. An extensive review of fresh concrete rheology is given in [1,2].

As the characteristic flow velocity will be very small compared to the speed of sound in the fresh concrete, the fluid will be treated as incompressible. In a case of incompressible flow, the mass and momentum conservation equations, together with the incompressibility condition and constitutive equation form a complete system. The energy conservation equation could be additionally used to obtain temperature field.

The present contribution is based on Eulerian formulation, which is typically used in CFD applications. This formulation is characterized by a coordinate system that is either stationary or moving in some prescribed manner in order to accommodate the continually changing solution domain. The mass travels between computational cells even if the grid moves, because grid movements are not related to the motion of the mass. The main advantage is the ability to represent large distortions without the lost of accuracy (in contrast to other methods) and handling of multiple interfaces is relatively straightforward. The main issue is the selection of the optimal strategy for the interface tracking. On the other hand, Lagrangian description can naturally handle material interfaces, but these methods are not suitable for very large deformation processes. An example of a Lagrangian approach employed in the context of fresh concrete flow modeling is the work of Dufour and Pijaudier-Cabot [3], who have developed a finite element formulation with Lagrangian integration points.

The numerical solution is based on the finite element method (FEM) and the interface-capturing method to track the position

* Corresponding author. Fax: +420 2 2431 0775.
E-mail address: borek.patzak@fsv.cvut.cz (B. Patzák).

0045-7949/$ - see front matter © 2008 Civil-Comp Ltd and Elsevier Ltd. All rights reserved.
doi:10.1016/j.compstruc.2008.04.015

of a free surface. Solution algorithm is based on a stabilized FEM formulation to prevent potential numerical instabilities. The stabilization techniques include streamline-upwind/Petrov–Galerkin (SUPG) and pressure-stabilizing/Petrov–Galerkin (PSPG) formulations. These stabilization techniques were introduced by Tezduyar and Hughes, see [4,5] for further reference.

In the computation of problems with moving boundaries, depending on the complexity of the interface and other aspects of the problem, interface tracking or interface-capturing techniques are usually used [6–9]. The interface tracking technique requires meshes that track or follow the interface. The mesh is updated as the interface evolves. In most cases, an automatic mesh generator needs to be used, sometimes at an overwhelming expense, to generate a new 3D mesh. Furthermore, every time the new mesh is generated, the solution needs to be projected from the old mesh to the new one. This process involves projection errors, and, in 3D, it requires significant computational time.

In the interface-capturing technique, the computation is done on fixed spatial domains. An interface function, marking the location of the interface, needs to be computed to track the interface. The interface is captured within the resolution of the finite element mesh, in the sense that the actual physical discontinuity is located near the middle value of characteristic function, which is defined to be equal to one for one reference phase and zero for the other.

Surface tracking methods represent interface as a series of interpolation curves using discrete set of points on the interface. The points move according to interface evolution. In the simplest case, only the sequence of heights above a reference line is maintained. More general algorithms rely on a parametric representation, allowing to represent more complex interface shapes. The advantage is the ability to represent details of the interface that are smaller than the cell spacing. On the other hand, it is extremely difficult to handle topological changes of the boundary (merging, folding, etc.). Volume tracking methods [7,10,11] solve a transport equation for the fraction of the cell occupied by the liquid phase. These methods rely on the ability to advect volume fraction through the grid without smearing. The reconstruction of the interface is done on a cell by cell basis using volume fraction value within the cell and its neighborhood. The best reconstruction algorithm is still to be developed. Typical reconstruction algorithms are based on SLIC or PLIC techniques. In general, the resolution of the volume of fluid method is limited by the grid size and it may be difficult to impose boundary conditions on the interface. The advantage is the capability to treat any number of mass constituents, interfaces with large distortion can be treated, and topology changes of the interface can be handled implicitly.

2. Stabilized finite element formulation

In this section, the variational formulation with SUPG and PSPG stabilization terms is briefly described. The stabilization provides stability and accuracy in the solution of advection-dominated problems and permits usage of equal-order interpolation functions for velocity and pressure. Furthermore, stabilized formulation significantly improves convergence rate in iterative solution of large non-linear systems of equations. The governing equations are based on the velocity–pressure formulation of the Navier–Stokes equations

$$\rho\left(\frac{\partial \boldsymbol{u}}{\partial t}+\boldsymbol{u}\cdot\nabla\boldsymbol{u}\right)-\nabla\cdot\boldsymbol{\sigma}=\boldsymbol{0}\quad\text{on }\Omega_t \tag{1}$$

$$\nabla\cdot\boldsymbol{u}=0\quad\text{on }\Omega_t \tag{2}$$

where ρ is the density and $\boldsymbol{u}$ is the velocity vector. The stress tensor $\boldsymbol{\sigma}$ is decomposed into its deviatoric part $\boldsymbol{\tau}$ and pressure p

$$\boldsymbol{\sigma}(p,\boldsymbol{u})=\boldsymbol{\tau}-p\boldsymbol{I} \tag{3}$$

The essential and natural boundary conditions are

$$\boldsymbol{u}=\boldsymbol{g}\quad\text{on }\Gamma_g \tag{4}$$

$$\boldsymbol{n}\cdot\boldsymbol{\sigma}=\boldsymbol{h}\quad\text{on }\Gamma_h \tag{5}$$

where Γ_g and Γ_h are complementary subsets of the boundary Γ, $\boldsymbol{n}$ is the unit outward normal vector, and $\boldsymbol{g}$ and $\boldsymbol{h}$ are given functions. A divergence free velocity field is specified as initial condition.

If the problem does not involve any moving boundary, the spatial domain does not need to change in time. This may be even the case for flows with moving boundaries, provided that the spatial domain is not defined to be the part of space occupied by the fluid. For example, one can have a fixed domain and by assuming that the domain is occupied by two immiscible fluids, the problem can be modeled by simultaneous tracking of mutual interface between the two fluids.

When modeling a free surface problem, where one fluid is irrelevant, one can assign a sufficiently low density to that fluid. This is also the approach adopted in this work.

Provided that trial solution and test function spaces for velocity and pressure (S_u^h, V_u^h, S_p^h, and $V_p^h=S_p^h$) are defined, then the stabilized finite element formulation of Eqs. (1) and (2) can be written as follows: find $\boldsymbol{u}^h\in S_u^h$ and $p^h\in S_p^h$ such that $\forall\boldsymbol{w}^h\in V_u^h$ and $\forall q^h\in V_p^h$

$$\begin{aligned}&\int_\Omega \boldsymbol{w}^h\cdot\rho\left(\frac{\partial\boldsymbol{u}^h}{\partial t}+\boldsymbol{u}^h\cdot\nabla\boldsymbol{u}^h\right)\mathrm{d}\Omega+\int_\Omega\boldsymbol{\varepsilon}(\boldsymbol{w}^h):\boldsymbol{\sigma}(p^h,\boldsymbol{u}^h)\mathrm{d}\Omega\\&+\int_{\Gamma_h}\boldsymbol{w}^h\cdot\boldsymbol{h}\,\mathrm{d}\Gamma+\int_\Omega q^h\nabla\cdot\boldsymbol{u}^h\,\mathrm{d}\Omega\\&+\sum_{e=1}^{n_{el}}\int_{\Omega_e}(\delta^h+\varepsilon^h)\cdot\left[\rho\left(\frac{\partial\boldsymbol{u}^h}{\partial t}+\boldsymbol{u}^h\cdot\nabla\boldsymbol{u}^h\right)-\nabla\cdot\boldsymbol{\sigma}(p^h,\boldsymbol{u}^h)\right]\mathrm{d}\Omega=0\end{aligned} \tag{6}$$

The first four terms correspond to the standard Galerkin formulation of (1)–(5). The last term contains two additional terms; the one with δ^h is the SUPG term, and the one with ε^h is the PSPG term. The functions δ^h and ε^h are defined as

$$\delta^h=\tau_{\mathrm{SUPG}}\boldsymbol{u}^h\cdot\nabla\boldsymbol{w}^h,\quad \varepsilon^h=\tau_{\mathrm{PSPG}}\frac{1}{\rho}\nabla q^h \tag{7}$$

The parameters τ_{SUPG} and τ_{PSPG} depend on element Reynolds numbers, which are based on the local and global scaling velocities and element length measures. Their definitions can be found, for example, in [5,12,13]. The spatial discretization of Eq. (6) leads to the following set of non-linear differential equations:

$$(\boldsymbol{M}+\boldsymbol{M}_\delta)\boldsymbol{a}+\boldsymbol{N}(\boldsymbol{v})+\boldsymbol{N}_\delta(\boldsymbol{v})+(\boldsymbol{K}+\boldsymbol{K}_\delta)\boldsymbol{v}-(\boldsymbol{G}+\boldsymbol{G}_\delta)\boldsymbol{p}=\boldsymbol{F}+\boldsymbol{F}_\delta \tag{8}$$

$$\boldsymbol{G}^{\mathrm{T}}\boldsymbol{v}+\boldsymbol{M}_\varepsilon\boldsymbol{a}+\boldsymbol{N}_\varepsilon(\boldsymbol{v})+\boldsymbol{K}_\varepsilon\boldsymbol{v}+G_\varepsilon\boldsymbol{p}=\boldsymbol{E}+\boldsymbol{E}_\varepsilon \tag{9}$$

where $\boldsymbol{v}$ is the vector of unknown nodal velocities, $\boldsymbol{a}$ is the vector of nodal accelerations, and $\boldsymbol{p}$ is the vector of nodal pressure values. The matrices $\boldsymbol{M}$, $\boldsymbol{N}$, $\boldsymbol{K}$, and $\boldsymbol{G}$ are derived from the time-dependent, advective, viscous, and pressure terms. The right-hand side vectors $\boldsymbol{F}$ and $\boldsymbol{E}$ are due to the boundary conditions. The subscripts δ and ε identify the SUPG and PSPG contributions.

By considering the following time discretization

$$\boldsymbol{a}^{t+\Delta t}=\boldsymbol{a}^t+\Delta\boldsymbol{a} \tag{10}$$

$$\frac{\boldsymbol{v}^{t+\Delta t}-\boldsymbol{v}^t}{\Delta t}=\alpha\boldsymbol{a}^{t+\Delta t}+(1-\alpha)\boldsymbol{a}^t \tag{11}$$

$$\boldsymbol{p}^{t+\Delta t}=\boldsymbol{p}^t+\Delta\boldsymbol{p} \tag{12}$$

we can formulate the following solution algorithm:

1. Evaluate velocity vector predictor at time $t+\Delta t$ as

$$\boldsymbol{v}=\boldsymbol{v}^t+\Delta t\boldsymbol{a}^t$$
$$\boldsymbol{a}=\boldsymbol{a}^t$$
$$\boldsymbol{p}=\boldsymbol{p}^t$$