Meshfree Numerical Scheme for Time Dependent Problems in Continuum Mechanics

Jörg Kuhnert[*] , Matthias Schäfer[*] , Robert Gerstenberger[#]

[*] *Fraunhofer Institut für Techno- und Wirtschaftsmathematik*

Fraunhofer Platz 1
67663 Kaiserslautern, Germany
e-mail:{joerg.kuhnert,matthias.schaefer}@itwm.fraunhofer.de, web page: http://www.itwm.fhg.de

[#] *Institut für Werkzeugmaschinen und Fabrikbetrieb*

Technische Universität Berlin
Produktionstechnisches Zentrum (PTZ)
Pascalstraße 8 - 9
10587 Berlin, Germany
e-mail: gerstenberger@iwf.tu-berlin.de, web page:http://www.iwf.tu-berlin.de

Abstract

The Finite Pointset Method (FPM) is a meshfree approach to numerically solve partial differential equations (PDE). The method uses a cloud of numerical points, each of which carrying necessary numerical data. In order to solve a PDE, FPM needs to establish approximations of derivatives, which are provided by a specialized least squares algorithm. The least squares idea turns the whole method into a generalized finite difference method. Even though the main field of application of FPM has been computational fluid dynamics (CFD), particularly applications in gasdynamics as well as incompressible flows, the purpose of the present article is to extend the application range of FPM to plastic and visco-plastic material behavior. Background to this field of application are cutting processes in metal industry. The article presents a new numerical solution procedure, that focuses on possibly general laws of visco-elasto-plastic material behavior, that also allows the employment of special stress tensor models as well as weak compressibility.

In this article, we will introduce the numerical scheme, which is used by the Finite Pointset Method (FPM) in order to solve for the classical conservation equations (mass, momentum, energy) in continuum mechanics on a meshfree basis. The material behavior is governed by general material models of visco-elasto-plastic type.

FPM is a meshfree numerical solver in continuum mechanics. It uses clouds of numerical points as a geometrical basis. The point cloud has to be sufficiently dense in the flow domain, such that it allows for a representative approximation of the continuum.

The points move with the velocity of the continuum, i.e. it is a Lagrangian idea.

The differential equations of motion are represented without the help of weak formulations. The local spatial derivatives are formed by a FPM-specialized least squares formulation, the time derivatives by simple finite differences.

In section 1, the point cloud management is briefly discussed.

In section 2, the differential equations to be considered are stated. This will give an overview about the spectrum of problems to be solved by FPM.

In section 3 provides the idea how to treat these differential equations in a compact numerical scheme. That includes a short derivation of the scheme, as it is non-classical.

In section 4, three practical implementations, that result out of chapter 3, are stated.

1. Point Cloud Management

A very important pre-condition to the method of FPM is the existence of a point cloud that compactly covers the whole computational domain Ω. The density of the points is given by a function

$$h = h(\mathbf{x},t) \tag{1.1}$$

i.e. a function in space and time, sufficiently smooth. Compact coverage of the computational domain is fulfilled if, in any ball with radius $r_{hole} \cdot h$ inside of Ω, at least one point is found. This can also be regarded as a quality constraint of the point cloud. $r_{hole} = 0.45$ leads to a point cloud, where 20...50 points are found within a ball of radius $h(\mathbf{x},t)$.

Another quality constraint is, that points are not allowed to assume a distance to each other smaller than $r_{small} \cdot h$. A practicable value is $r_{small} = 0.1...0.2$.

As the point cloud moves with the velocity of the continuum, the quality of the point cloud has to be maintained possibly after each numerical time step. New points are filled if holes occur (i.e. balls of radius $r_{hole} \cdot h$ with no point inside). Points are deleted, if a pair of points is found whose distance is less than $r_{small} \cdot h$.

2. Physical model / set of equations solved by FPM

The fluid flow, modeled in FPM, is governed by three conservation laws, given in differential form.

2.1. Conservation of mass

The mass conservation in Lagrangian form is given by

$$\frac{d}{dt}\rho + \rho \cdot \nabla^T \mathbf{v} = 0 \tag{2.1}$$

where the operator $\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{v}^T \cdot \nabla$ denotes the material derivative.

For fluids with constant density, (1.1) reduces to

$$\nabla \mathbf{v} = 0 \tag{2.2}$$

a typical constraint for incompressible flows.

2.2. Conservation of momentum

The momentum conservation, also in Lagrangian form, is given by

$$\frac{d}{dt}(\rho \mathbf{v}) + (\rho \mathbf{v}) \cdot \nabla^T \mathbf{v} = \left(\nabla^T \mathbf{S}\right)^T - \nabla p + \rho \cdot \mathbf{g} \tag{2.3}$$

2.3. Conservation of energy

The principle of conservation of thermal energy is expressed by the equation

$$\frac{d}{dt}(\rho E) + (\rho E) \cdot \nabla^T \mathbf{v} = \nabla^T (\mathbf{S} \cdot \mathbf{v}) - \nabla^T (p \cdot \mathbf{v}) + \left(\rho \cdot \mathbf{g}^T \cdot \mathbf{v}\right) + \nabla^T \cdot (k \cdot \nabla T) \tag{2.4}$$

The operator $\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{v}^T \cdot \nabla$ represents the material derivative, i.e. the change of the physical quantity along a path of a fluid particle, and the gradient operator is $\nabla = \left(\frac{\partial}{\partial x} \quad \frac{\partial}{\partial y} \quad \frac{\partial}{\partial z}\right)^T$, where especially $\nabla^T \cdot \mathbf{v}$ leads to the divergence of some vector field and ∇p to the gradient of some scalar function.

The physical quantities in detail:

Density	ρ	
Momentum	$\rho \mathbf{v}$	
Total energy	ρE	$= \rho e + \frac{1}{2}\rho\left(\mathbf{v}^T \cdot \mathbf{v}\right)$
Internal energy	e	and we have $de = c_V \cdot dT$
Temperature	T	
Velocity	$\mathbf{v}$	$= (u \quad v \quad w)^T$
Position	$\mathbf{x}$	$= (x \quad y \quad z)^T$
Pressure	p	
Stress tensor, deviator. part	$\mathbf{S}$	
Gravity / body forces	$\mathbf{g}$	

2.4. Formulation for Stress Tensor

In fact, the physical model above is very general. In order to derive a useful numerical scheme, we split the stress tensor into its viscous and solid parts

$$\mathbf{S} = \mathbf{S}_{solid} + \mathbf{S}_{visc} \tag{2.5}$$

These two types of stresses have different character and are defined as

$$\mathbf{S}_{visc} = \eta \cdot \frac{\mathrm{d}\boldsymbol{\varepsilon}}{\mathrm{dt}} \tag{2.6}$$

and

$$\frac{d\mathbf{S}_{solid}}{d\mathrm{t}} = \mu \cdot \frac{d\boldsymbol{\varepsilon}}{d\mathrm{t}} + \mathbf{K} \cdot \mathbf{S}_{solid} - \mathbf{S}_{solid} \cdot \mathbf{K} \tag{2.7}$$

The strain rate tensor $\frac{\mathrm{d}\boldsymbol{\varepsilon}}{\mathrm{dt}}$ is given by

$$\frac{d\boldsymbol{\varepsilon}}{d\mathrm{t}} = \frac{1}{2}\left[\nabla \mathbf{v}^T + \left(\nabla \mathbf{v}^T\right)^T\right] - \frac{1}{3}\left(\nabla^T \cdot \mathbf{v}\right) \cdot \mathbf{I} \tag{2.8}$$

accounting for the local deformation (form change) of the fluid. The tensor $\mathbf{K}$ represents the fact, that the stresses $\mathbf{S}_{solid}$ undergo a rigid rotation, which partially result from the rigid part of the continuous motion and partially result from the deformation itself. Some authors use here the so called Jaumann-rate by defining $\mathbf{K}_{Jaumann} = \frac{1}{2}\left[\nabla \mathbf{v}^T - \left(\nabla \mathbf{v}^T\right)^T\right]$.

It is important to understand, that the Jaumann approach is risky especially for big deformations and therefore not used by our approach. We use a special time integration of $\mathbf{S}_{solid}$, described in [16].

Finally, the whole modeling above requires material properties, which are the heat conductivity k, viscosity η and shear modulus μ.

In metal forming, the shear modulus as well as viscosity in metal forming is, usually, implicitly given by models like Johnson-Cook or Zerilli-Armstrong. In the present paper, we will concentrate on the Jonson-Cook model. Treatment of other models will be similar. The Johnson-Cook model provides a correlation between equivalent stress and equivalent strain.

$$\mathrm{S}_{JC} = \left(\mathrm{S}_0 + \mathrm{B} \cdot \varepsilon_p^{\mathrm{n}}\right) \cdot \left(1 + \mathrm{C} \cdot \log\left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0}\right)\right) \cdot \left(1 - \mathrm{T}_{\mathrm{hom}}^{\mathrm{m}}\right) \tag{2.9}$$

with material parameters $\mathrm{S}_0, \mathrm{B}, \mathrm{n}, \mathrm{C}, \dot{\varepsilon}_0, \mathrm{m}$. The plastic strain ε_p measures the deformation of the material after reaching the yield stress.

It is obvious, that S_{JC} represents both solid and viscous parts of the deformation process.

$$\mathrm{S}_{solid,JC} = \left(\mathrm{S}_0 + \mathrm{B} \cdot \varepsilon_p^{\mathrm{n}}\right) \cdot \left(1 - \mathrm{T}_{\mathrm{hom}}^{\mathrm{m}}\right) \tag{2.10}$$

$$\mathrm{S}_{visc,JC} = \left(\mathrm{S}_0 + \mathrm{B} \cdot \varepsilon_p^{\mathrm{n}}\right) \cdot \left(\mathrm{C} \cdot \log\left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0}\right)\right) \cdot \left(1 - \mathrm{T}_{\mathrm{hom}}^{\mathrm{m}}\right) = \mathrm{S}_{solid,JC} \cdot \left(\mathrm{C} \cdot \log\left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0}\right)\right) \tag{2.11}$$

It is simple to reconstruct the shear modulus with the help of (2.10) and the viscosity using equation (2.11).

$$\mu_{JC} = \frac{dS_{solid,JC}}{d\varepsilon_p} = \begin{cases} \left(B \cdot n \cdot \varepsilon_p^{n-1}\right) \cdot \left(1 - T_{hom}^{m}\right) & \text{if } S_{solid,JC} > S_0 \\ \mu_{elastic} & \text{otherwise} \end{cases} \tag{2.12}$$

$$\eta_{JC} = \begin{cases} S_{solid,JC} \cdot \left(C \cdot \log\left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right) \cdot \frac{1}{\dot{\varepsilon}} & \text{if } \dot{\varepsilon} > \dot{\varepsilon}_0 \\ 0 & \text{otherwise} \end{cases} \tag{2.13}$$

3. Numerical Model – Derivation of Time integration of the conservation laws

For numerical modeling, we rewrite equations (2.1), (2.3), and (2.4) in primitive form, i.e. in a density, velocity and temperature formulation.

Conservation of mass

$$\frac{d}{dt}\rho + \rho \cdot \nabla^T \mathbf{v} = 0 \tag{3.1}$$

Conservation of momentum

$$\frac{d}{dt}(\mathbf{v}) = \frac{1}{\rho}\left(\nabla^T \mathbf{S}\right)^T - \frac{1}{\rho}\nabla p + \mathbf{g} \tag{3.2}$$

Conservation of energy

$$\left(\rho c_V\right)\left(\frac{d}{dt}T\right) = \nabla^T\left(\mathbf{S}\cdot\mathbf{v}\right) - \left(\nabla^T \mathbf{S}\right)\cdot\mathbf{v} - p\cdot\left(\nabla^T\cdot\mathbf{v}\right) + \nabla^T\cdot(k\cdot\nabla T) + q \tag{3.3}$$

Equations (3.1), (3.2), and (3.3) are equivalent to the model (2.1)-(2.3).

Solid Stresses

$$\frac{d\mathbf{S}_{solid}}{dt} = \mu \cdot \frac{d\boldsymbol{\varepsilon}}{dt} + \mathbf{K}\cdot\mathbf{S}_{solid} - \mathbf{S}_{solid}\cdot\mathbf{K} \tag{3.4}$$

Equation of state $$\rho = \rho(p, T) \tag{3.5}$$

the density is a function of pressure and temperature, in the special case of incompressible fluid flow, the density is constant along the path of a fluid particle.

For the following numerical model, we assume incompressible flow, i.e. we consider the velocity to be divergence free. The algorithm is implicit in most of its parts. For compressible fluid flow, gasdynamics, a very much different numerical scheme, explicit in most of its parts, was developed and presented in a dedicated paper.

With the equations (3.1)...(3.5), we are able to establish a numerical scheme for the meshfree basis of FPM. The scheme is a direct scheme, i.e. it tries to approximate the differential equations without the help of weak formulations. The spatial derivatives in the physical model are simply replaced by the least squares

approximations based on the local point cloud [10]. We simply take notice of this by replacing the mathematical differential operators ∇ and Δ (and others as well) by their numerical representations $\tilde{\nabla}$ and $\tilde{\Delta}$.

3.1. Numerical Model: Integration of velocity

First, we represent the numerical formulation of the time derivatives. For simplicity, we demonstrate it based on a first order time approximation. Please note, that this would in fact be possible also for higher order time integration.

$$\frac{\mathbf{v}^{n+1}-\mathbf{v}^{n}}{\Delta t} = \frac{1}{\rho}\left(\tilde{\nabla}^{T}\mathbf{S}^{n+1}\right)^{T} - \frac{1}{\rho}\tilde{\nabla}p + \mathbf{g} \tag{3.6}$$

$$\frac{\mathbf{S}_{solid}^{n+1}-\mathbf{S}_{solid}^{n}}{\Delta t} = \mu\cdot\left.\frac{d\boldsymbol{\varepsilon}}{dt}\right|^{n+1} + \mathbf{K}^{n}\cdot\mathbf{S}_{solid}^{n}-\mathbf{S}_{solid}^{n}\cdot\mathbf{K}^{n} \tag{3.7}$$

With the help of (2.6) we can rewrite (3.6) as

$$\frac{\mathbf{v}^{n+1}-\mathbf{v}^{n}}{\Delta t} = \frac{1}{\rho}\left(\tilde{\nabla}^{T}\left(\mathbf{S}_{solid}^{n+1} + \eta\cdot\left.\frac{d\boldsymbol{\varepsilon}}{dt}\right|^{n+1}\right)\right)^{T} - \frac{1}{\rho}\tilde{\nabla}p + \mathbf{g} \tag{3.8}$$

The time level (n+1) on the right hand side indicates, that we employ here a time-implicit scheme. Furthermore, the term $\mathbf{S}_{solid}^{n+1}$ in the equation above can be expressed with the help of equation (3.7).

$$\mathbf{S}_{solid}^{n+1} = \mathbf{S}_{solid}^{n} + \Delta t\cdot\left(\mu\cdot\left.\frac{d\boldsymbol{\varepsilon}}{dt}\right|^{n+1}\right) + \Delta t\cdot\left(\mathbf{K}^{n}\cdot\mathbf{S}_{solid}^{n} - \mathbf{S}_{solid}^{n}\cdot\mathbf{K}^{n}\right)$$

With this, equation (3.8) becomes

$$\begin{aligned}&\frac{\mathbf{v}^{n+1}-\mathbf{v}^{n}}{\Delta t}\\ &= \frac{1}{\rho}\left(\tilde{\nabla}^{T}\left(\mathbf{S}_{solid}^{n} + \Delta t\cdot\left(\mathbf{K}^{n}\cdot\mathbf{S}_{solid}^{n}-\mathbf{S}_{solid}^{n}\cdot\mathbf{K}^{n}\right)+\left(\eta+\Delta t\cdot\mu\right)\cdot\left.\frac{d\boldsymbol{\varepsilon}}{dt}\right|^{n+1}\right)\right)^{T}\\ &- \frac{1}{\rho}\tilde{\nabla}p + \mathbf{g}\end{aligned} \tag{3.9}$$

After some reordering, it furthermore becomes

$$\begin{aligned}\frac{\mathbf{v}^{n+1}-\mathbf{v}^{n}}{\Delta t} &= \frac{1}{\rho}\left(\tilde{\nabla}^{T}\left(\left(\eta+\Delta t\cdot\mu\right)\cdot\left.\frac{d\boldsymbol{\varepsilon}}{dt}\right|^{n+1}\right)\right)^{T} - \frac{1}{\rho}\tilde{\nabla}p +\\ &\mathbf{g}+\frac{1}{\rho}\tilde{\nabla}^{T}\left(\mathbf{S}_{solid}^{n} + \Delta t\cdot\left(\mathbf{K}^{n}\cdot\mathbf{S}_{solid}^{n}-\mathbf{S}_{solid}^{n}\cdot\mathbf{K}^{n}\right)\right)^{T}\end{aligned} \tag{3.10}$$

The definitions of effective (numerical) body forces

$$\hat{\mathbf{g}} \equiv \mathbf{g}+\frac{1}{\rho}\tilde{\nabla}^{T}\left(\mathbf{S}_{solid}^{n} + \Delta t\cdot\left(\mathbf{K}^{n}\cdot\mathbf{S}_{solid}^{n}-\mathbf{S}_{solid}^{n}\cdot\mathbf{K}^{n}\right)\right)^{T} \tag{3.11}$$

and effective (numerical) viscosity

$$\hat{\eta} \equiv \eta + \Delta t \cdot \mu \tag{3.12}$$

help to rewrite equation (3.10) in a simple way

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\Delta t} = \frac{1}{\rho}\left(\tilde{\nabla}^T \left(\hat{\eta} \cdot \frac{d\boldsymbol{\varepsilon}}{dt}\bigg|^{n+1} \right) \right)^T - \frac{1}{\rho}\tilde{\nabla}\hat{p} + \hat{\mathbf{g}} \tag{3.13}$$

Here, the effective body forces $\hat{\mathbf{g}}$ combine the original body forces (gravity, ...) with the local inner stresses. The effective viscosity $\hat{\eta}$ provides enough damping such that the numerical scheme produces smooth solutions. It is a particular nice property that the term ($\Delta t \cdot \mu$) becomes small as the time step size is forced to be small. That means, elastic waves (shear waves) will be resolved very well with appropriately small time step size. Bigger time step sizes will wash out sound waves and emphasize on the bulk behavior.

Details about the numerical treatment of the resulting pressure $\hat{p}$ will be given in section 3.2.

We now turn the attention to the term $\frac{1}{\rho}\left(\tilde{\nabla}^T \left(\hat{\eta} \cdot \frac{d\boldsymbol{\varepsilon}}{dt}\bigg|^{n+1} \right) \right)^T$, found in equation (3.13).

which is not ready yet to be incorporated into a numerical scheme. However, recalling the definition of the strain rate tensor (equation (2.8)), we are able to define

$$\Psi_{\hat{\eta}}(\mathbf{v}) \equiv \frac{1}{\rho}\left(\nabla^T \left(\hat{\eta} \cdot \frac{d\boldsymbol{\varepsilon}}{dt}\bigg| \right) \right)^T \tag{3.14}$$

knowing, that this term is just a function of the local velocity and its local, spatial derivatives. As for example, for pure incompressible flows together with a locally constant viscosity, we have

$$\tilde{\Psi}_{\hat{\eta}}(\mathbf{v}) = \hat{\eta} \cdot \tilde{\Delta}\mathbf{v} = \hat{\eta} \cdot \left(\frac{\tilde{\partial}^2 \mathbf{v}}{\tilde{\partial} x^2} + \frac{\tilde{\partial}^2 \mathbf{v}}{\tilde{\partial} y^2} + \frac{\tilde{\partial}^2 \mathbf{v}}{\tilde{\partial} z^2} \right)$$

For locally varying viscosity term, we have

$$\tilde{\Psi}^T_{\hat{\eta}}(\mathbf{v}) = \tilde{\nabla}^T \left(\hat{\eta} \cdot \tilde{\nabla} \right) \mathbf{v}^T + \left(\tilde{\nabla}\hat{\eta} \right)^T \cdot \left(\tilde{\nabla}\mathbf{v}^T \right)^T + \frac{1}{3}\hat{\eta}\left(\tilde{\nabla} D \right)^T - \frac{2}{3}\left(\tilde{\nabla}\hat{\eta} \right)^T \cdot D \tag{3.15}$$

where $D = \tilde{\nabla}^T \mathbf{v}^T$ is simply the divergence of velocity. If a standard computation is performed with $\tilde{\nabla}\hat{\eta} = 0$ and $D=0$, it suffices to evaluate the numerical operator of $\tilde{\nabla}^T \left(\hat{\eta} \cdot \tilde{\nabla} \right)$. For $\tilde{\nabla}\hat{\eta} \neq 0$, additional evaluations of first order numerical derivatives are necessary. For $D \neq 0$, the term $\tilde{\nabla} D$ contains mixed derivatives of second order, which cause even more computational effort. Measurements of computation times show, that cases with $\tilde{\nabla}\hat{\eta} \neq 0$ and $D \neq 0$ run by a factor 2 ... 2.5 slower.

Using definition (3.14), equation (3.13) is expressed as

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\Delta t} = \frac{1}{\rho}\tilde{\Psi}_{\hat{\eta}}\left(\mathbf{v}^{n+1} \right) - \frac{1}{\rho}\tilde{\nabla}\hat{p} + \hat{\mathbf{g}} \tag{3.16}$$

which, after shifting the terms, appears as

$$\mathbf{v}^{n+1} - \frac{\Delta t}{\rho}\tilde{\Psi}_{\hat{\eta}}\left(\mathbf{v}^{n+1}\right) \quad = \quad \mathbf{v}^{n} - \quad \frac{\Delta t}{\rho}\tilde{\nabla}\hat{p} \quad + \quad \Delta t \cdot \hat{\mathbf{g}} \tag{3.17}$$

or, even more compactly as

$$\left(\mathbf{I} - \frac{\Delta t}{\rho}\tilde{\Psi}_{\hat{\eta}}\right)\left(\hat{\mathbf{v}}^{n+1}\right) \quad = \quad \mathbf{v}^{n} - \quad \frac{\Delta t}{\rho}\tilde{\nabla}\hat{p} \quad + \quad \Delta t \cdot \hat{\mathbf{g}} \tag{3.18}$$

In order to find the solution for the velocity at time level n+1, FPM solves exactly this equation. On the left had side, we find terms depending on time level n+1, on the right hand sight, there are terms of time level n. By its character, equation (3.18) is a time implicit scheme for the velocity. The left hand side operator $\left(\mathbf{I} - \frac{\Delta t}{\rho}\tilde{\Psi}_{\hat{\eta}}\right)$ gives rise to the construction of a big sparse matrix, each line of which containing the local, discrete representation of this operator. The right hand side appears as a load vector. Hence, equation (3.18) represents a big sparse linear system. The solution to this are the velocity components associated with the appropriate particle location.

With $\hat{\mathbf{v}}^{n+1}$, we do not compute the true solution of the velocity but a first guess. $\hat{\mathbf{v}}^{n+1}$ might be in need of further correction, as it might not meet all constraints. For example, $\tilde{\nabla}^T\hat{\mathbf{v}}^{n+1}$ (divergence of velocity) might not be of correct value. The correction of $\hat{\mathbf{v}}^{n+1}$ towards the correct velocity $\hat{\mathbf{v}}^{n+1}$ is directly connected to the computation of the correction pressure, see section 3.2.

3.2. Numerical Model: Solution to the Pressure

3.2.1. Classical use of Chorins correction pressure

In order to produce a correct approximation of the velocity by the numerical model (3.18), we need a good model for the pressure $\hat{p}$. If $\hat{p}$ is wrong, especially the divergence of the velocity $\tilde{\nabla}^T\hat{\mathbf{v}}^{n+1}$ produced by (3.18) might most probably be wrong. On the other hand, the pressure at time level n+1 is not known in advance. In order to provide at least a good guess of $\hat{p}$, it is worth while recalling how pressure and velocity are connected. Equation (3.2), solved by FPM, can be studied in more detail:

$$\frac{d}{dt}(\mathbf{v}) \quad = \quad \frac{1}{\rho}\left(\nabla^T\mathbf{S}\right)^T \quad - \quad \frac{1}{\rho}\nabla p \quad + \quad \mathbf{g}$$

Application of the divergence operator to this equation brings

$$\nabla^T\left(\frac{d}{dt}(\mathbf{v})\right) \quad = \quad \nabla^T\left(\frac{1}{\rho}\left(\nabla^T\mathbf{S}_{visc}\right)^T\right) \quad - \nabla^T\left(\frac{1}{\rho}\nabla p\right) \quad + \quad \nabla^T\left(\mathbf{g} + \frac{1}{\rho}\left(\nabla^T\mathbf{S}_{solid}\right)^T\right)$$

which equivalently gives

$$\frac{d}{dt}\left(\nabla^T\mathbf{v}\right) + \Phi(\mathbf{v}) \quad = \quad \nabla^T\left(\frac{1}{\rho}\left(\nabla^T\mathbf{S}_{visc}\right)^T\right) \quad - \quad \nabla^T\left(\frac{1}{\rho}\nabla p\right) \quad + \quad \nabla^T\left(\mathbf{g} + \frac{1}{\rho}\left(\nabla^T\mathbf{S}_{solid}\right)^T\right)$$

Here, $\Phi(\mathbf{v})$ is given by

$$\Phi(\mathbf{v}) \quad \equiv \quad \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial z}\right)^2 \quad + \quad 2\left(\frac{\partial u}{\partial y}\right)\cdot\left(\frac{\partial v}{\partial x}\right) + 2\left(\frac{\partial u}{\partial z}\right)\cdot\left(\frac{\partial w}{\partial x}\right) + 2\left(\frac{\partial v}{\partial z}\right)\cdot\left(\frac{\partial w}{\partial y}\right)$$

So, by knowing the velocity field throughout the computational domain, we can write down a Poisson-type equation in order to solve for the pressure field:

$$\nabla^T\left(\frac{1}{\rho}\nabla \mathrm{p}\right) \quad = \quad -\frac{d}{dt}\left(\nabla^T \mathbf{v}\right) - \Phi\left(\mathbf{v}\right) \; + \; \nabla^T\left(\frac{1}{\rho}\left(\nabla^T \mathbf{S}_{visc}\right)^T\right) \quad + \; \nabla^T\left(\mathbf{g} + \frac{1}{\rho}\left(\nabla^T \mathbf{S}_{solid}\right)^T\right) \tag{3.19}$$

We split the pressure into a hydrostatic part and into a dynamic part. The hydrostatic part is formed from body forces and inner stresses

$$\nabla^T\left(\frac{1}{\rho}\nabla \mathrm{p}_{hyd}\right) \quad = \quad \nabla^T\left(\mathbf{g} + \frac{1}{\rho}\left(\nabla^T \mathbf{S}_{solid}\right)^T\right) \tag{3.20}$$

And the dynamic part arises due to the movement of the fluid (i.e. the dynamic part vanishes as the fluid does not move)

$$\nabla^T\left(\frac{1}{\rho}\nabla \mathrm{p}_{dyn}\right) \quad = \quad -\frac{d}{dt}\left(\nabla^T \mathbf{v}\right) - \Phi\left(\mathbf{v}\right) \; + \; \nabla^T\left(\frac{1}{\rho}\left(\nabla^T \mathbf{S}_{visc}\right)^T\right) \tag{3.21}$$

Equations (3.20) and (3.21) provide a natural splitting of the pressure, which will be used for the numerical scheme.

We define the pressure $\hat{\mathrm{p}}$ of the solution model for the velocity (3.18) as

$$\hat{\mathrm{p}} = \; \mathrm{p}_{hyd}^{n+1} \; + \; \mathrm{p}_{dyn}^{n} \tag{3.22}$$

Logically, following equations (3.20) and (3.11), the numerical scheme for the hydrostatic pressure follows by

$$\tilde{\nabla}^T\left(\frac{1}{\rho}\tilde{\nabla} \mathrm{p}_{hyd}^{n+1}\right) \quad = \quad \tilde{\nabla}^T \hat{\mathbf{g}} \tag{3.23}$$

It is not straight forward to determine the dynamic pressure at time level n+1, as we would need the velocity $\mathbf{v}^{n+1}$, yet unknown. Therefore, p_{dyn}^{n} of the previous time level is taken into account. We keep in mind, that, by this, we introduce an error concerning the pressure. In order to correct this error, we employ a correction pressure, which we apply as a posteriori formulation.

Suppose we have computed the solution to the numerical model (3.18), i.e. we solved

$$\hat{\mathbf{v}}^{n+1} - \frac{\Delta t}{\rho}\tilde{\Psi}_{\hat{\eta}}\left(\hat{\mathbf{v}}^{n+1}\right) \quad = \quad \mathbf{v}^{n} - \quad \frac{\Delta t}{\rho}\tilde{\nabla}\hat{\mathrm{p}} \; + \; \Delta t \cdot \hat{\mathbf{g}} \tag{3.24}$$

for $\hat{\mathbf{v}}^{n+1}$. As mentioned, there is an error on the pressure, which we attempt to correct by a correction term ε^{n+1} similar to Chorin's projection idea [3]. Our model (3.18) would produce a slightly different solution, namely

$$\mathbf{v}^{n+1} - \frac{\Delta t}{\rho}\tilde{\Psi}_{\hat{\eta}}\left(\mathbf{v}^{n+1}\right) \quad = \quad \mathbf{v}^{n} - \quad \frac{\Delta t}{\rho}\tilde{\nabla}\hat{\mathrm{p}} - \quad \frac{\Delta t}{\rho}\tilde{\nabla}\varepsilon^{n+1} \; + \; \Delta t \cdot \hat{\mathbf{g}} \tag{3.25}$$

Subtracting equation (3.24) from (3.25) yields

$$\mathbf{v}^{n+1} - \hat{\mathbf{v}}^{n+1} - \frac{\Delta t}{\rho}\tilde{\Psi}_{\hat{\eta}}\left(\mathbf{v}^{n+1} - \hat{\mathbf{v}}^{n+1}\right) \quad = \quad - \quad \frac{\Delta t}{\rho}\tilde{\nabla}\varepsilon^{n+1} \tag{3.26}$$

We apply the divergence operator to (3.26) and obtain

$$\left(\tilde{\nabla}^T \mathbf{v}^{n+1}\right) - \left(\tilde{\nabla}^T \hat{\mathbf{v}}^{n+1}\right) - \tilde{\nabla}^T \left(\frac{\Delta t}{\rho} \tilde{\Psi}_{\hat{\eta}} \left(\mathbf{v}^{n+1} - \hat{\mathbf{v}}^{n+1} \right) \right) \quad = \quad - \tilde{\nabla}^T \left(\frac{\Delta t}{\rho} \tilde{\nabla} \varepsilon^{n+1} \right) \tag{3.27}$$

Moreover we assume the term

$$\tilde{\nabla}^T \left(\frac{\Delta t}{\rho} \tilde{\Psi}_{\hat{\eta}} \left(\mathbf{v}^{n+1} - \hat{\mathbf{v}}^{n+1} \right) \right) \approx 0 \tag{3.28}$$

such that we can neglect it. One can always achieve this by adjusting the size of the time step. Thus, from (3.27) it follows

$$\tilde{\nabla}^T \left(\frac{\Delta t}{\rho} \tilde{\nabla} \varepsilon^{n+1} \right) = - \left(\tilde{\nabla}^T \mathbf{v}^{n+1}\right) + \left(\tilde{\nabla}^T \hat{\mathbf{v}}^{n+1}\right) \tag{3.29}$$

The equation above represents a Poisson-type equation to solve for the correction pressure ε^{n+1}. That, however requires the knowledge of the two right hand side terms $\tilde{\nabla}^T \hat{\mathbf{v}}^{n+1}$ and $\tilde{\nabla}^T \mathbf{v}^{n+1}$.

For the first one, it is a straight forward task to evaluate the divergence of the known velocity $\hat{\mathbf{v}}^{n+1}$. For incompressible flows, we also know immediately that the second term

$$\tilde{\nabla}^T \mathbf{v}^{n+1} \overset{!}{=} 0 \, .$$

In general however, the flow might be compressible. We can estimate the divergence of the velocity by the equation of mass conservation (3.1) and the given equation of state (3.5):

$$\nabla^T \mathbf{v} = - \frac{1}{\rho} \left(\tfrac{\mathrm{d}}{\mathrm{dt}} \rho \right) \tag{3.30}$$

With

$$\frac{d}{dt} \rho = \frac{\partial \rho}{\partial \mathrm{p}} \left(\tfrac{\mathrm{d}}{\mathrm{dt}} \mathrm{p} \right) + \frac{\partial \rho}{\partial \mathrm{T}} \left(\tfrac{\mathrm{d}}{\mathrm{dt}} \mathrm{T} \right) + \frac{\partial \rho}{\partial \mathrm{t}} \tag{3.31}$$

we have

$$\nabla^T \mathbf{v} = - \frac{1}{\rho} \left(\frac{\partial \rho}{\partial \mathrm{p}} \left(\tfrac{\mathrm{d}}{\mathrm{dt}} \mathrm{p} \right) + \frac{\partial \rho}{\partial \mathrm{T}} \left(\tfrac{\mathrm{d}}{\mathrm{dt}} \mathrm{T} \right) + \frac{\partial \rho}{\partial \mathrm{t}} \right) \tag{3.32}$$

Numerical estimations of pressure and temperature change rates for the time level n+1 lead to the approximative formulation of the divergence of velocity to be numerically prescribed to the scheme (3.29).

$$\tilde{\nabla}^T \mathbf{v}^{n+1} = - \frac{1}{\rho} \left(\frac{\partial \rho}{\partial \mathrm{p}} \frac{\left(\mathrm{p}_{\mathrm{hyd}}^{n+1} - \mathrm{p}_{\mathrm{hyd}}^{n} + \varepsilon^{n+1} \right)}{\Delta t} + \frac{\partial \rho}{\partial \mathrm{T}} \frac{\left(\mathrm{T}^{n+1} - \mathrm{T}^{n} \right)}{\Delta t} + \frac{\partial \rho}{\partial \mathrm{t}} \right) \tag{3.33}$$

Let us write down equation (3.33) in a more simple way.

$$\tilde{\nabla}^T \mathbf{v}^{n+1} \quad = \quad - \frac{1}{\rho} \frac{\partial \rho}{\partial \mathrm{p}} \frac{\varepsilon^{n+1}}{\Delta t} \quad + \quad \overline{\nabla^T \mathbf{v}^{n+1}} \tag{3.34}$$

together with the definition

$$\overline{\nabla^T \mathbf{v}^{n+1}} = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial p}\frac{\left(p_{hyd}^{n+1} - p_{hyd}^{n}\right)}{\Delta t} + \frac{\partial \rho}{\partial T}\frac{\left(T^{n+1} - T^{n}\right)}{\Delta t} + \frac{\partial \rho}{\partial t} \right) \tag{3.35}$$

With the conclusion (3.34), we can finally rewrite (3.29) as

$$\tilde{\nabla}^T \left(\frac{\Delta t}{\rho}\tilde{\nabla}\varepsilon^{n+1} \right) = -\left(-\frac{1}{\rho}\frac{\partial \rho}{\partial p}\frac{\varepsilon^{n+1}}{\Delta t} \; + \; \overline{\nabla^T \mathbf{v}^{n+1}} \right) + \left(\tilde{\nabla}^T \hat{\mathbf{v}}^{n+1} \right) \tag{3.36}$$

which, after some shifting, appears as

$$-\frac{1}{\rho}\frac{\partial \rho}{\partial p}\frac{1}{\Delta t}\varepsilon^{n+1} + \tilde{\nabla}^T \left(\frac{\Delta t}{\rho}\tilde{\nabla}\varepsilon^{n+1} \right) \; = \; \nabla^T \hat{\mathbf{v}}^{n+1} \; - \; \overline{\nabla^T \mathbf{v}^{n+1}} \tag{3.37}$$

This equation is exactly the one, which is solved by FPM for the correction pressure. The term $-\frac{1}{\rho}\frac{\partial \rho}{\partial p}\frac{1}{\Delta t}\varepsilon^{n+1}$ accounts for the compressibility of the fluid. Equation (3.37) is a differential equation of parabolic type. The left hand side contains the terms of the unknown pressure ε^{n+1}.

By definition of the differential operators

$$D_\varepsilon \equiv \nabla^T \left(\frac{\Delta t}{\rho}\nabla \right) \tag{3.38}$$

and

$$\mathbf{I}_\varepsilon \equiv -\frac{1}{\rho}\frac{\partial \rho}{\partial p}\frac{1}{\Delta t} \tag{3.39}$$

We have

$$\left(\mathbf{I}_\varepsilon + \tilde{D}_\varepsilon \right)\varepsilon^{n+1} \; = \; \tilde{\nabla}^T \hat{\mathbf{v}}^{n+1} \; - \; \overline{\nabla^T \mathbf{v}^{n+1}} \tag{3.40}$$

The left hand side forms a matrix, each line of which representing the discrete representation of the operator, the right hand side forms a load vector of terms which are all known. Hence, (3.40) forms a linear system with the unknown ε^{n+1}.

(3.40) is very similar to Chorin's projection idea, a commonly used idea for re-projection of incompressible flows. In fact, for incompressible flows, i.e. for $\frac{\partial \rho}{\partial p} = 0$ and $\frac{\partial \rho}{\partial T} = 0$,

(3.40) reduces exactly to the idea of Chorin. Equation (3.26) directly suggest the correction of the velocity towards the right divergence:

$$\mathbf{v}^{n+1} = \hat{\mathbf{v}}^{n+1} \; - \; \frac{\Delta t}{\rho}\tilde{\nabla}\varepsilon^{n+1} \tag{3.41}$$

Thus, the new velocity field is established.

Finally, the dynamic pressure has to be updated in the sense suggested by the correction approach (3.25), i.e.

$$p_{dyn}^{n+1} = p_{dyn}^{n} \; + \; \varepsilon^{n+1} \tag{3.42}$$

3.2.2. Derivation of the dynamic pressure from the equation of momentum

Equation (3.21) another valid way of modelling the dynamic pressure, in fact it is the direct way. Applied for the time level n+1, it reads

$$\nabla^T\left(\frac{1}{\rho}\nabla \mathrm{p}_{dyn}^{n+1}\right) \quad = \quad -\frac{d}{d\mathrm{t}}\left(\nabla^T\mathbf{v}^{n+1}\right)-\Phi\left(\mathbf{v}^{n+1}\right) \; + \; \nabla^T\left(\frac{1}{\rho}\left(\nabla^T\mathbf{S}_{visc}^{n+1}\right)^T\right) \tag{3.43}$$

Equations (3.42) and (3.43) should deliver comparable results. It is up to the user whether to employ this or that. Using (3.43) explicitly means to numerically approximate the total time derivative of the divergence of the velocity, i.e.

$$\frac{d}{d\mathrm{t}}\left(\nabla^T\mathbf{v}^{n+1}\right)\approx\frac{\tilde{\nabla}^T\mathbf{v}^{n+1}-\tilde{\nabla}^T\mathbf{v}^{n}}{\Delta\mathrm{t}} \tag{3.44}$$

This term could be neglected if we work with incompressible flows, however in general, for weakly compressible flows, it has to be there. The values for the divergence at the two time levels n and $n+1$ have to be numerically approximated, there is unfortunately no direct way to provide an analytical solution. If used for incompressible flows, (3.44) might generate additional fluctuations, as the numerical computation of the divergence $\tilde{\nabla}^T\mathbf{v}$ is never really zero.

3.3. Numerical Model: Integration of the Internal Stresses

According to equation (3.4), the inner stresses can now be computed at time level n+1.

$$\mathbf{S}_{solid}^{n+1} \quad = \mathbf{S}_{solid}^{n} \quad + \quad \Delta\mathrm{t}\cdot\left(\mu\cdot\left.\frac{d\boldsymbol{\varepsilon}}{d\mathrm{t}}\right|^{\mathrm{n}+1}\right) \; + \; \Delta\mathrm{t}\cdot\left(\mathbf{K}^{\mathrm{n}}\cdot\mathbf{S}_{solid}^{\mathrm{n}}-\mathbf{S}_{solid}^{\mathrm{n}}\cdot\mathbf{K}^{\mathrm{n}}\right) \tag{3.45}$$

This now is a straight forward task as all quantities on the right hand side are known (remember that $\left.\frac{d\boldsymbol{\varepsilon}}{d\mathrm{t}}\right|^{\mathrm{n}+1}$ depends only on $\mathbf{v}^{n+1}$).

A direct integration of (2.7) by the model (3.45) is however not always the best choice. Particularly the employment of material models different from Johnson-Cook (2.9), where the shear modulus as well as viscosity is not directly reproducible from the model itself (see (2.12) and (2.13)), the employment of some black box solver function $\mathfrak{F}$ is necessary. Suppose, $\mathfrak{F}$ provides an optimized numerical integration of the stress tensor by

$$\mathbf{S}_{solid}^{n+1} \quad = \mathfrak{F}\left(\mathbf{S}_{solid}^{n},\mathbf{v}^{\mathrm{n}+1},\mathbf{v}^{\mathrm{n}},\Delta\mathrm{t}\right) \tag{3.46}$$

Using black box solvers means, that there is no explicit formulation for the effective shear modulus μ. The numerical model for the velocity (3.10) however requires the knowledge of μ. How to get an estimation of μ out of the black box solver? Let us assume that the change of the inner stresses always perform similar to the pattern (3.45). It is at least possible to provide values $\hat{\mu}$ and $\hat{\mathbf{K}}$ such that the black box solution (3.46) is possibly close to the pattern (3.45), i.e. find values $\hat{\mu}$ and $\hat{\mathbf{K}}$ such that

$$\left\|\mathfrak{F}\;\left(\mathbf{S}_{solid}^{n},\mathbf{v}^{\mathrm{n}+1},\mathbf{v}^{\mathrm{n}}\right)\;-\;\left(\mathbf{S}_{solid}^{n}+\Delta\mathrm{t}\cdot\left(\hat{\mu}\cdot\left.\frac{d\boldsymbol{\varepsilon}}{d\mathrm{t}}\right|^{\mathrm{n}+1}\right)\;+\;\Delta\mathrm{t}\cdot\left(\hat{\mathbf{K}}\cdot\mathbf{S}_{solid}^{\mathrm{n}}-\mathbf{S}_{solid}^{\mathrm{n}}\cdot\hat{\mathbf{K}}\right)\right)\right\|_2 \;\overset{!}{=}\; \min \tag{3.47}$$

FPM integrates the stress tensor using a solver described in [16]. The values $\hat{\mu}$ and $\hat{\mathbf{K}}$ can be employed in the next time cycle for the velocity solution model (3.10). The equation (3.12) for the effective numerical viscosity will use $\hat{\mu}$ instead of μ. It is obvious, that the value of $\hat{\mu}$ crucial to the whole scheme. Being too small, it will lead to numerical instabilities, as the idea of implicit incorporation of the stresses (3.8) is violated. Being too big, the numerical solution will smooth out, as now the effective viscosity becomes bigger than necessary.

3.4. Numerical Model: Integration of Energy Equation

It remains to integrate the temperature as solution to the energy balance equation (3.3).

$$\left(\rho c_V\right)\left(\frac{d}{dt}T\right) = \nabla^T\left(\mathbf{S}\cdot\mathbf{v}\right) - \left(\nabla^T\mathbf{S}\right)\cdot\mathbf{v} - p\cdot\left(\nabla^T\cdot\mathbf{v}\right) + \nabla^T\cdot(k\cdot\nabla T) + q \tag{3.48}$$

We again formulate it as an implicit scheme in the sense

$$\begin{aligned}\left(\rho c_V\right)\cdot T^{n+1} &- \Delta t\cdot\tilde{\nabla}^T\cdot(k\cdot\tilde{\nabla}T^{n+1}) = \left(\rho c_V\right)\cdot T^n + \\ &\Delta t\cdot\left(\tilde{\nabla}^T\left(\mathbf{S}^{n+1}\cdot\mathbf{v}^{n+1}\right) - \left(\tilde{\nabla}^T\mathbf{S}^{n+1}\right)\cdot\mathbf{v}^{n+1}\right) - \\ &\Delta t\cdot p^{n+1}\cdot\left(\tilde{\nabla}^T\cdot\mathbf{v}^{n+1}\right) + \Delta t\cdot q^{n+1}\end{aligned} \tag{3.49}$$

Similarly as for velocity and pressure above, we define differential operators

$$\mathbf{I}_T + D_T \equiv \left(\rho c_V\right) - \Delta t\cdot\nabla^T\cdot(k\cdot\nabla) \tag{3.50}$$

With this, (3.49) might be rewritten as

$$\begin{aligned}\left(\mathbf{I}_T + \tilde{D}_T\right)T^{n+1} &= \left(\rho c_V\right)\cdot T^n + \\ &\Delta t\cdot\left(\tilde{\nabla}^T\left(\mathbf{S}^{n+1}\cdot\mathbf{v}^{n+1}\right) - \left(\tilde{\nabla}^T\mathbf{S}^{n+1}\right)\cdot\mathbf{v}^{n+1}\right) - \\ &\Delta t\cdot p^{n+1}\cdot\left(\tilde{\nabla}^T\cdot\mathbf{v}^{n+1}\right) + \Delta t\cdot q^{n+1}\end{aligned} \tag{3.51}$$

The left hand side forms a stiffness matrix, whose lines represent the discrete approximations of the differential operator (3.50). The right hand side represents a load vector. In fact, (3.51) forms a sparse linear system of equations, whose unknowns are the discrete Temperature values T^{n+1} at each particle position.

4. Numerical Model -Compact Solution Algorithms for Time-Dependent Conservation Equations used by FPM

From the considerations above, three different numerical schemes can be deducted. The first idea is the simple application of Chorin's projection to the needs of FPM. The second is an enhancement of the first towards highly viscous flows. However the second scheme is not unconditionally stable, which gives rise to the third scheme, which is computationally more expensive than the second one.

4.1. Classical Chorin Reprojection Idea

For low viscous flows, especially in the case where the time step size is smaller than the viscous relaxation time $\Delta t_{relax} = \frac{\rho \cdot h^2}{\eta + \Delta t \cdot \mu}$, the classical Chorin idea [3] works perfectly even for FPM.

0.) Initialize

$$p_{dyn}^{n+1} := p_{dyn}^{n}$$

$$T^{n+1} := T^{n}$$

1.) Move the point cloud:

$$\mathbf{x}^{n+1} = \mathbf{x}^{n} + \Delta t \cdot \mathbf{v}^{n}$$

2.) Compute actual hydrostatic pressure: see equation (3.23)

$$\nabla^T \left(\frac{1}{\rho} \nabla p_{hyd}^{n+1} \right) = \nabla^T \hat{\mathbf{g}}$$

3.) Establish pressure guess:

$$\hat{p} = p_{hyd}^{n+1} + p_{dyn}^{n+1}$$

4.) Produce best guess for velocity: see equation (3.18)

$$\left(\mathbf{I} - \frac{\Delta t}{\rho} \Psi_{\hat{\eta}} \right) \left(\hat{\mathbf{v}}^{n+1} \right) = \mathbf{v}^{n} - \frac{\Delta t}{\rho} \nabla \hat{p} + \Delta t \cdot \hat{\mathbf{g}}$$

5.) Compute Correction Pressure: see equation (3.40)

$$\left(\mathbf{I}_{\varepsilon} + D_{\varepsilon} \right) \varepsilon^{n+1} = \nabla^T \hat{\mathbf{v}}^{n+1} - \overline{\nabla^T \mathbf{v}^{n+1}}$$

6.) Correct velocity and produce final velocity: see equation (3.41)

$$\mathbf{v}^{n+1} = \hat{\mathbf{v}}^{n+1} - \frac{\Delta t}{\rho} \nabla \varepsilon^{n+1}$$

7.) Update dynamic pressure: see equation (3.42)

$$p_{dyn}^{n+1} := p_{dyn}^{n+1} + \varepsilon^{n+1}$$

8.) Update internal stresses: see equation (3.45)

$$\mathbf{S}_{solid}^{n+1} = \mathbf{S}_{solid}^{n} + \Delta t \cdot \left(\mu \cdot \frac{d\boldsymbol{\varepsilon}}{dt} \Big|^{n+1} \right) + \Delta t \cdot \left(\mathbf{K}^{n} \cdot \mathbf{S}_{solid}^{n} - \mathbf{S}_{solid}^{n} \cdot \mathbf{K}^{n} \right)$$

9.) Update temperature: see equation (3.51)

$$\left(\mathbf{I}_T + D_T\right)T^{n+1} = \left(\rho c_V\right)\cdot T^n + \Delta t\cdot\left(\nabla^T\left(\mathbf{S}^{n+1}\cdot\mathbf{v}^{n+1}\right)-\left(\nabla^T\mathbf{S}^{n+1}\right)\cdot\mathbf{v}^{n+1}\right)- \Delta t\cdot p^{n+1}\cdot\left(\nabla^T\cdot\mathbf{v}^{n+1}\right) + \Delta t\cdot q^{n+1}$$

We have to keep in mind that the assumption (3.28) always be fulfilled. That works well for high Reynolds-Numbers down to Re=0.1 or even a bit less. However, with decreasing Re, the term (3.28) becomes more and more dominant, so it is not any more negligible. A very easy solution is to iterate the above scheme between the numbers 3.) up to 9.). However, for Re numbers tending to zero, even an iteration would be enormously costly, many iteration cycles would have to be performed. Thus, the performance of the FPM scheme would break down. Therefore, we have extended the present scheme for low Reynolds numbers.

4.2. Chorin's Reprojection for Low Reynolds Number Flows

This scheme is an extension of the scheme presented in 4.1. designed to perform better for Reynolds numbers tending to zero.

0.) Initialize

$$p_{dyn}^{n+1} := p_{dyn}^{n}$$

$$T^{n+1} := T^{n}$$

Chose A

1.) Move the point cloud:

$$\mathbf{x}^{n+1} = \mathbf{x}^n + \Delta t\cdot\mathbf{v}^n$$

2.) Compute actual hydrostatic pressure: see equation (3.23)

$$\nabla^T\left(\frac{1}{\rho}\nabla p_{hyd}^{n+1}\right) = \nabla^T\hat{\mathbf{g}}$$

3.) Establish preliminary pressure:

$$\hat{p} = p_{hyd}^{n+1} + p_{dyn}^{n+1}$$

4.) Produce best guess for velocity: see equation (3.18)

$$\left(\mathbf{I}-\frac{\Delta t}{\rho}\Psi_{\hat{\eta}}\right)\left(\hat{\mathbf{v}}^{n+1}\right) = \mathbf{v}^n - \frac{\Delta t}{\rho}\nabla\hat{p} + \Delta t\cdot\hat{\mathbf{g}}$$

5.) Compute virtual time step:

$$\Delta t_{virt} = \min\left(A\cdot\frac{\rho\cdot h^2}{\eta+\Delta t\cdot\mu}, \Delta t\right)$$

Remark: h = local mean distance between points

6.) Compute Correction Pressure: see equation (3.40)

$$\left(\mathbf{I}_{\varepsilon} + \mathrm{D}_{\varepsilon,virt}\right)\varepsilon^{n+1} \;=\; \left(\nabla^{T}\hat{\mathbf{v}}^{n+1}\right) \;-\; \overline{\nabla^{T}\mathbf{v}^{n+1}}$$

Remark: $$\mathrm{D}_{\varepsilon,virt} \equiv \nabla^{T}\left(\frac{\Delta t_{virt}}{\rho}\nabla\right)$$

7.) Correct velocity and produce final velocity: see equation (3.41)

$$\mathbf{v}^{n+1} = \hat{\mathbf{v}}^{n+1} \;-\; \frac{\Delta t_{virt}}{\rho}\nabla\varepsilon^{n+1}$$

8.) Update dynamic pressure: see equation (3.42)

$$p_{dyn}^{n+1} = p_{dyn}^{n} \;+\; \varepsilon^{n+1}$$

9.) Update internal stresses: see equation (3.45)

$$\mathbf{S}_{solid}^{n+1} \;=\; \mathbf{S}_{solid}^{n} \;+\; \Delta t\cdot\left(\mu\cdot\frac{d\boldsymbol{\varepsilon}}{dt}\bigg|^{n+1}\right) \;+\; \Delta t\cdot\left(\mathbf{R}^{n}\cdot\mathbf{S}_{solid}^{n} - \mathbf{S}_{solid}^{n}\cdot\mathbf{R}^{n}\right)$$

10.) Update temperature: see equation (3.51)

$$\begin{aligned}\left(\mathbf{I}_{T} + \mathrm{D}_{T}\right)T^{n+1} \;=& \left(\rho c_{V}\right)\cdot T^{n} + \\ & \Delta t\cdot\left(\nabla^{T}\left(\mathbf{S}^{n+1}\cdot\mathbf{v}^{n+1}\right) - \left(\nabla^{T}\mathbf{S}^{n+1}\right)\cdot\mathbf{v}^{n+1}\right) - \\ & \Delta t\cdot p^{n+1}\cdot\left(\nabla^{T}\cdot\mathbf{v}^{n+1}\right) \;+\; \Delta t\cdot q^{n+1}\end{aligned}$$

This scheme takes into account the effect that many highly viscous flows reach a quasi-stationary state after a very short viscous relaxation time $\Delta t_{relax} \;=\; A\cdot\frac{\rho\cdot h^{2}}{\eta + \Delta t\cdot\mu}$. The relaxation time becomes important when the dynamic pressure is computed.

This scheme works very well even for Reynolds tending to zero. However, the scheme depends on the choice of the parameter A. For many cases, A=1 is sufficient. However, there are even more applications, for which A has to be chosen bigger. Generally: if A is too small, the computations become unstable, especially non-regular p_{dyn} can be observed. Up to now, There is no way yet to automatically determine A If a computation breaks down, one has to change A by hand.

The parameter A, itself, strongly depends only on the mean particle distance (h) and the shape of the geometry concerned, i.e. it is a pure geometrical constant.

The choice of A is crucial in this scheme. It prevents, that FPM can run absolutely automatically. If instabilities occur, the user has to turn up A. As this procedure is still not yet acceptable for many industrial cases, we have developed a more sophisticated, penalty-oriented scheme based on the present one.

4.3. Penalty Formulation Developed for Low Reynolds Flow

0.) Initialize

$$p_{dyn}^{n+1} := p_{dyn}^{n}$$

$$T^{n+1} := T^{n}$$

Chose A

1.) Move the point cloud:

$$\mathbf{x}^{n+1} = \mathbf{x}^{n} + \Delta t \cdot \mathbf{v}^{n}$$

2.) Compute actual hydrostatic pressure: see equation (3.23)

$$\nabla^T \left(\frac{1}{\rho} \nabla p_{hyd}^{n+1} \right) = \nabla^T \hat{\mathbf{g}}$$

3.) Establish preliminary pressure:

$$\hat{p} = p_{hyd}^{n+1} + p_{dyn}^{n+1}$$

4.) Compute virtual time step:

$$\Delta t_{virt} = \min\left(A \cdot \frac{\rho \cdot h^2}{\eta + \Delta t \cdot \mu}, \Delta t \right)$$

Remark: h = local mean distance between points

5.) Solve simultaneously for velocity and correction pressure

$$\left(\mathbf{I} - \frac{\Delta t}{\rho} \Psi_{\hat{\eta}} \right) \left(\hat{\mathbf{v}}^{n+1} \right) = \mathbf{v}^{n} - \frac{\Delta t}{\rho} \nabla \hat{p} - \frac{\Delta t}{\rho} \nabla \varepsilon^{n+1} + \Delta t \cdot \hat{\mathbf{g}}$$

$$\left(\mathbf{I}_{\varepsilon} + D_{\varepsilon, virt} \right) \varepsilon^{n+1} = \left(\nabla^T \hat{\mathbf{v}}^{n+1} \right) - \overline{\nabla^T \mathbf{v}^{n+1}}$$

Remark: $D_{\varepsilon, virt} \equiv \nabla^T \left(\frac{\Delta t_{virt}}{\rho} \nabla \right)$

6.) Correct velocity and produce final velocity: see equation (3.41)

$$\mathbf{v}^{n+1} = \hat{\mathbf{v}}^{n+1} - \frac{\Delta t_{virt}}{\rho} \nabla \varepsilon^{n+1}$$

7.) Update dynamic pressure: see equation (3.42)

$$p_{dyn}^{n+1} = p_{dyn}^{n} + \varepsilon^{n+1}$$

8.) Update internal stresses: see equation (3.45)

$$\mathbf{S}_{solid}^{n+1} = \mathbf{S}_{solid}^{n} + \Delta t \cdot \left(\mu \cdot \frac{d\boldsymbol{\varepsilon}}{dt} \Big|^{n+1} \right) + \Delta t \cdot \left(\mathbf{R}^{n} \cdot \mathbf{S}_{solid}^{n} - \mathbf{S}_{solid}^{n} \cdot \mathbf{R}^{n} \right)$$

9.) Update temperature: see equation (3.51)

$$\begin{aligned} \left(\mathbf{I}_T + D_T \right) T^{n+1} = & \left(\rho c_V \right) \cdot T^{n} + \\ & \Delta t \cdot \left(\nabla^T \left(\mathbf{S}^{n+1} \cdot \mathbf{v}^{n+1} \right) - \left(\nabla^T \mathbf{S}^{n+1} \right) \cdot \mathbf{v}^{n+1} \right) - \\ & \Delta t \cdot p^{n+1} \cdot \left(\nabla^T \cdot \mathbf{v}^{n+1} \right) + \Delta t \cdot q^{n+1} \end{aligned}$$

In step 5.), we write one single linear system for the velocity components together with the correction pressure. This results in the fact, that A can be chosen arbitrarily small. If chosen smaller, the parameter A results in more precise results. However, A chosen in the range of 1 is usually sufficient, if A is chosen smaller, the effects of improvement are small, however the computation time rises. There is one drawback to this solution, which is not yet solved. The algorithm to solve sparse linear systems (BiCGstab) does not always converge for the velocity-pressure system in point 5.). The convergence itself also depends on the size of A. Some examples work very well, some examples break down. Currently, other algorithms are tested (AMG).

4.4. Conclusions

We have derived three different schemes. The classical Chorin-reprojection-idea works well for high and moderate Reynolds numbers (down to the order of 0.1). For smaller Re, we have extended the classical reprojection by Chorin to the second scheme, designed to converge fast for Re -> 0. There is uncertainty about some parameter A, depending on the geometry of the problem. There is not yet a secure method to choose the optimal A, moreover, A chosen too small will lead to unstable solutions. The third scheme (penalty formulation) overcomes the instability problems of the second scheme. Also here, there is a drawback consisting in occasional break down of the solver for sparse linear systems, attempting to solve the coupled velocity-pressure-equation.

For cutting processes with material laws for metals, the scheme 4.3 is the only one that produces good results [5].

5. References

[1] Belytschko, T.; Krongauz, Y.; Organ, D.; Flemming, M.; Krysl, P. (1996). Meshless methods. An overview and recent developments. Comput. Methods Appl. Mech. Engng., 139, 3.

[2] Chorin, A. J. (1968). Numerical solution of the Navier-Stokes equations, Math. Comp., 22, 745

[3] Chorin, A..J. (1967). A numerical method for solving incompressible flow problems; J. Comp. Physics 2, 12-26

[4] Duarte, C. A.; Oden, J. T.: (1996). H-p clouds and h-p meshless methods. Num. Meth. for PDEs.

[5] Gerstenberger, R., Schäfer M., Kuhnert J. (2010). Applications of the Finite Pointset Method (FPM) for the simulation of cutting processes, NN.

[6] Griebel, M., Schweitzer, A. (2002). Lecture Notes in Computational Science and Engineering, Vol. 26, Meshfree Methods for Partial Differential Equations I, Springer.

[7] Griebel, M., Schweitzer, A. (2005). Lecture Notes in Computational Science and Engineering, Vol. 43, Meshfree Methods for Partial Differential Equations II, Springer.

[8] Griebel, M., Schweitzer, A. (2006). Lecture Notes in Computational Science and Engineering, Vol. 57, Meshfree Methods for Partial Differential Equations III, Springer.

[9] Iliev, O., Tiwari, S. (2002). A generalized (meshfree) finite difference discretization for elliptic interface problems, (I. Dimov, I. Lirkov, S. Margenov, and Z. Zlatev, eds.), Numerical Methods and Applications, Lecture Notes in Computer Sciences, Springer, 480.

[10] Kuhnert, J. (2010). Specialized Numerical Differential Operators for the Finite Point Set Method; preprint Fraunhofer ITWM

[11] Liszka, T, Orkisz, J. (1980). The finite difference method on arbitrary irregular grid and its application in applied mechanics, Computers & Structures, 11, 83.

[12] Liu, M. B., Liu, G. R., Lam, K. Y. (2003). Modeling incompressible flows using a finite particle method. Appl. Math. Model., 29(12), 1252.

[13] Lucy, L. B. (1977). A numerical approach to the testing of the fission hypothesis. Astron. J., 82, 1013.

[14] Monaghan, J. J. (1992). Smooth particle hydrodynamics. Annu. Rev. Astron. Astrop, 30, 543.

[15] Monaghan, J. J. (1994). Simulating free surface flows with SPH. J. Comput. Phys., 110, 399.

[16] Schäfer, M.; ,Kuhnert, J. (2010). Finite Pointset Method for Chip Formation; preprint Fraunhofer ITWM, 2010

[17] Tiwari S., Kuhnert J.(2002). A meshfree method for incompressible fluid flows with incorporated surface tension , revue aurope'enne des elements finis. J.Comp. Appl. Math. 11, 965.

[18] Tiwari S., Kuhnert J. (2007). Modeling of two-phase flows with surface tension by Finite Pointset Method (FPM). J. Comp. Appl. Math., 203, 376.

[19] G. R. Johnson, W. H. Cook. (1983) A constitutive model and data for metals subjected to large strains, high strain rates and high temperatures. 7th International Symposium on Ballistics.

[20] F. J. Zerilli, R. W. Armstrong. (1987) Dislocation-mechanics-based constitutive relations for material dynamics calculations. J. Appl. Phys., 61, No.5, (1987), 1816-1825.