

Strategies for an Optimized simulation of granular particles in a Newtonian Fluid, Part 1: Basics

Hans-Georg Matuttis,

University of Electro-Communications, Department of Mechanical Engineering and Intelligent Systems, Chofu, Chofugaoka 1-5-1, Tokyo 182-8585, Japan

Abstract: In this article, an outline is given for an "ideal" simulation of granular particles in fluids with respect to CPU-effort, accuracy etc, which features should be incorporated in the underlying simulation, and which numerical techniques should be used with respect to the time integration.

1 Introduction

Though many features of granular materials, especially those in connection with static or quasi-static regimes (heap-formation, arching), a whole class of granular phenomena is influenced by the surrounding fluid:

Fluidized beds and Sedimentation are the terms used to describe systems where grains in various concentrations interact with the surrounding fluid under the influence of gravity.

Pattern formation processes like rippling and dune formation take place in air and under water. Hydrological phenomena like the formation of coastlines, the formation and destruction of river banks and the change of river beds also belongs in this class.

Dust avalanches of powder snow, where the avalanche is supposed to ride on a cushion of compressed air, are much faster, far reaching and more destructive than the fluid-like avalanches of "sherbet-like" snow.

Pneumatic transport is the field which deals with the intentional or unintentional movement of granular materials by fluids, ranging from vacuum cleaners over devices in chemical engineering to sandstorms.

Porous Media is the field where the surrounding "granular matrix" is considered to be stationary, and only the fluid it contains in the pore space is supposed to move.

Landslides are the result of an interaction between a granular matrix of a slope and a pore fluid which destabilizes the whole system.

We will outline which features must be incorporated in a simulation to access the widest range of the above phenomena with the minimum amount of computational effort while retaining physical validity. The granular materials shall be modeled on the particle level (Lagrangian method), to be able to investigate micro-mechanic mechanisms of macroscopic phenomena. The surrounding fluid should be simulated in a grid-based discretization as Newtonian fluid (Eulerian method), so that artifacts introduced by some grid-based methods (Lattice-Gas automata[9], Lattice Boltzmann simulations[6, 17], see Fig. 1) are avoided. Particle based methods (moving particle semi-implicit MPS[16, 15], smoothed particle dynamics, SPH[10, 22]) for the modelization of the fluid will

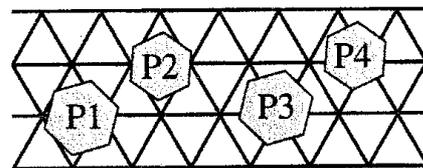


Figure 1: Grid artifact: Though the relative position of particle P1 versus P2 is the same as that of particle P3 versus P4, in the first case the path between the particles is blocked, only in the latter case there is a path for the fluid.

not be considered here, as they lead to shot-noise due to the discrete nature of the flow and will not be practicable. The resulting combination of granular particles and Newtonian Fluid will of course not necessarily behave like a Newtonian Fluid. The aim is to develop a simulation for granular materials in fluids which is "optimal" in the sense that

1. it allows the input of realistic conditions (particle shape, friction, etc),
2. is able to reproduce the macroscopic (rippling, ...) and mesoscopic (saltation, collapse of fluid-filled cavities, ...) behavior,
3. can be used for low (sedimentation) and high (porous media) granular density alike,
4. uses the minimum amount of computer time and storage in comparison with other computational methods and
5. maybe is even faster (in real time) than a purely "dry" simulation or a pure liquid due to a-priori considerations which will be explained below.

Previous simulations have dealt with the incorporation of particles into a fluid by using a straightforward approach, using the standard square grid with MAC[12] (marker and cell) and the standard granular particle simulation with round particles. Apart from the problem that the blocking of flow by particles cannot be treated with round particles, most of the algorithmic effort went into the treatment of the incommensurability of the straight grids and curved grain boundaries. Kajishima[14] used a brute-force approach, putting sub-meshes around the particles several orders of magnitude smaller than the particle diameter; the accuracy gained in the description of the particle outline lead to a huge blowup of the number of mesh-points, and accordingly, the CPU-effort is gigantic. Schwarzer et al[29, 13, 30] tried to match the particles with a similar-sized mesh, coupling the particle motion to tracer particles in the fluid: In a sense, the fluid goes "through" the particle, which are "fixed" to the flow lines with "springs". These "springs" (coupling constants) introduce additional degrees of freedom (stiffness, damping), for which the timescale is difficult to predict, which leads to numerically instable choices of the time-step. Moreover, most of the Computer time (99 %) goes in the solution of the MAC-pressure-iteration for the incompressibility-condition, due to incommensurability of particle boundaries and grid positions. In this article, we will explain how to circumvent the above problems by not choosing the most straightforward approach in the modeling of the primitives of the simulation (round particles, quadrilateral grid), but with the advantage that the treatment of the particle-fluid interface becomes straightforward.

1.1 Continuum/ Fluid part

The sound velocity c of a continuum solid can be calculated from its Young modulus Y and its density ρ as $c = \sqrt{Y/\rho}$. The sound velocity of a chain made of spherical particles is about 10 % of that of the continuum material the particles are made of: $c_{chain} \approx 0.1c$. The sound velocity of a three-dimensional disordered packing of mono-disperse particles c_{dis} is again on order of magnitude less, so that the relation $c_{dis} \approx 0.1c_{chain} \approx 0.01c$ hold[25]. Therefore, the sound velocity of the granular part can be considered considerably lower than that of the fluid part, which itself is about that of a continuum solid. Therefore, the fluid part will be best approximated as an incompressible fluid. If, as a starting point, the fluid is to be approximated as a Newtonian fluid, this means that the incompressible Navier-Stokes equation can be used to model the fluid part.

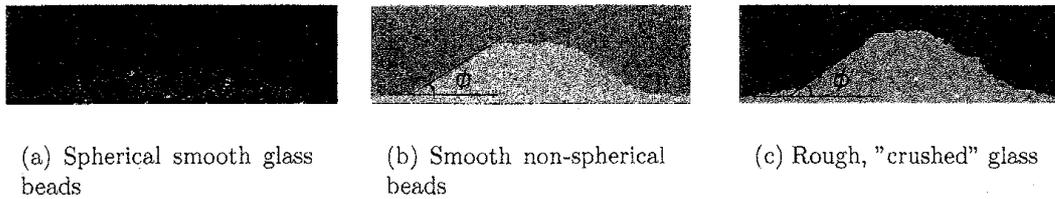


Figure 2: Effect of elongation and surface structure of glass beads on the critical angle

1.2 Particle Modeling

The bulk properties of assemblies of circular /spherical granular particles differ considerably from that of non-spherical particles: Whereas the angle of repose for dry spherical glass beads in Fig. 2(a) is about 20 degrees, if "straight" slopes can be identified at all, for the elongated non-spherical particles in Fig. 2(b) one obtains a more sand-like angle of repose of about 30 degrees. Surface roughness of the grains and the size dispersion relation also effects the angle of repose, as the angle of repose for the rough glass's in Fig. 2(c) shows. Another quantitative difference can be seen in the stress-strain diagram, where the maximal yield strength of elongated polydisperse particles is twice that of round particles (Fig. 3(a)), and the differences in the density-strain diagram also shows that the internal structure must be different (Fig. 3(b)). For these reasons, a simulation of granular particles in a fluid should make use of non-spherical particles, so that e.g. for sedimentation, the resulting slopes can be computed with acceptable stability and precision.

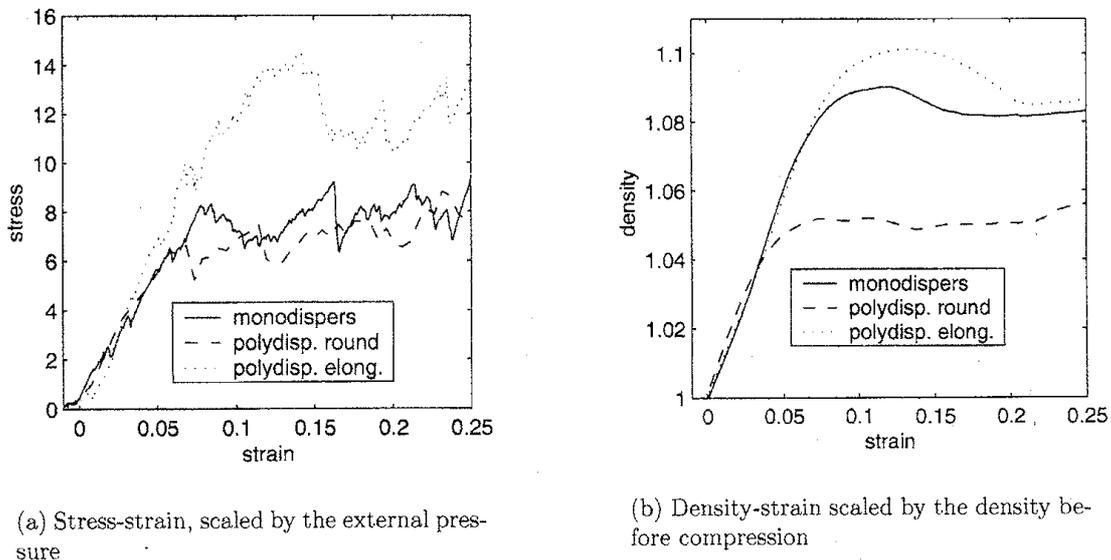


Figure 3: Granular properties of a typical single run of biaxial compression of monodisperse round particles (full line), polydisperse round particles (dashed line) and polydisperse elongated particles (dotted line) [21].

As the friction coefficient has also considerable influence on the bulk properties, the

appropriate lubricated solid friction for particles under water has to be implemented: This means that there is still static friction as in the dry case, but the coefficient of friction will be smaller. For the modeling of static friction, the computation of the contact point/area between the particles is necessary, no matter whether a tangential dash-pot-spring model[7] or the numerical exact implementation via differential algebraic equations ("Contact dynamics" cite moreau88, moreau89, or Ref. cite Hairer:93, p. 199) is used: This excludes the possibility of modeling the particles with elliptic potentials[32], which gives only a force direction, but not a traceable contact point. All in all, the use of polygonal (polyhedral in three dimensions) particles in connection with a surrounding fluid is preferable to e.g. particles with curved boundaries, because the geometric information is geometrically better defined.

Finally, the use of soft particle models (with finite young modulus) is preferable over "rigid particles" (infinite young modulus) in event-driven particle simulations[19, 20] or in contact mechanics[23, 24]: Event-driven simulations allow only the simulation of binary collisions, and as there is a suspicion that the net interaction of particles submerged under water is attractive, this would disturb agglomeration. Contact mechanics can treat multiple contact (at considerable larger algorithmic effort than the event-driven method); A drawback is that modeling the granular particles with rigid contacts results in an infinite sound velocity, which is physically dubious and in the presence of an incompressible fluids may lead to serious numerical instabilities due to the interference of two infinite signal velocities.

1.3 Statistical Physics

In accuracy benchmarks in fluid mechanics, e.g. the flow behind an obstacle, "idealized" boundary conditions for the likewise idealized Newtonian fluid are chosen so that solutions can be given or at least defined in arbitrary precision; in the case of Karman vortices, this will be an ideal circle or sphere as obstacle. For low Reynolds numbers, this will be the flow lines, for higher Reynolds numbers, where no stationary solution exists, one can still expect the existence of probability distributions for e.g. the size of vortices in the Karman vortex street which should be reproduced exactly by simulations and experiments alike. Physically, the relevance of such definitions may be actually doubtful, as experimentally already for Reynolds numbers as low as 20, huge deviations have been found in the streamlines of flow through a small orifice for polar and non-polar-simulations, on the one hand, and the numerical practically exact simulation on the other hand ([31] and References therein); A-priory, it is unclear why the experimentally used fluids (distilled water, ethanol and liquid paraffin) deviate so strongly from ideal Newtonian behavior.

In granular material research from the point of statistical physics, on the other hand, one is not so much interested in single particle problems with arbitrary precision, but tries to derive the macroscopic quantities based on the microscopic mechanisms. In that respect, for the sedimentation of granular particles in a fluid, the experimental verification should not be a verification of the particle trajectory (impossible, because it is obviously a nonlinear-chaotic problem), but the speed of the sedimentation fronts, and the angle of the resulting slopes.

Whereas, as discussed in the previous paragraph, "high accuracy" is not an issue due to the lack of "high accuracy reference data", "high stability" has to be retained at all

costs: No fluctuating noise terms due to lack of precision in the solution of the continuity equation should be introduced, because they might lead to unphysically oscillatory force laws, so that the granular phase would become fluidized more than is physically realistic.

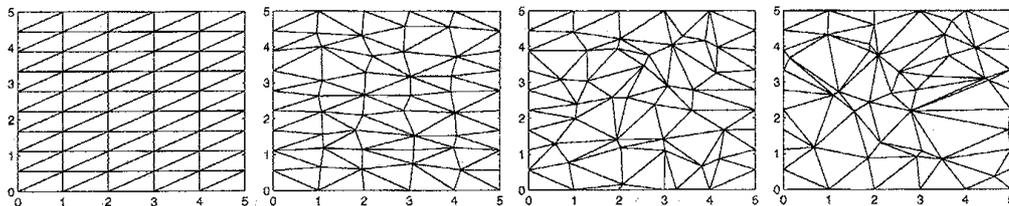


Figure 4: Delaunay-Triangulations of a square-grid deformed with Gaussian-distributed random numbers ($\sigma = 0/\text{Friedrichs - Keller - Grid}, 0.05, 0.1, 0.2$)

1.4 The Grid

Structured grids have the advantage that the resulting system of equation is characterized by a band Matrix, so that conventional LU-Solvers can be used. The disadvantage is that for complicated simulation geometries, the construction of the grid is an arbitrarily complex task, even more so if the neighborhood relation of the grid points must be taken into account as an additional constraint. If two granular particles approach each other, the space between them can become arbitrarily narrow: The treatment of the fluid with a more or less uniform grid spacing, as prevalent in the MAC-method, is not suitable for this case, because if the regular grid with the smallest grid size (smallest particle distance) is used, this leads to unnecessary many degrees of freedom.

Unstructured grids give the largest freedom for the distribution of the mesh-points, and can therefore be optimally adapted to the particle- and flow-geometry. The disadvantage is that "numerically exact" LU-Solvers become inefficient, and sparse matrix methods have to be employed, which are numerically considerably less stable.

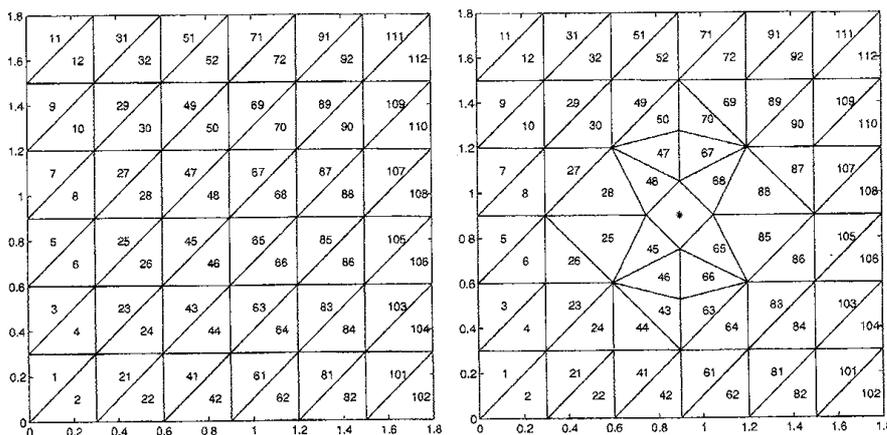


Figure 5: Friedrichs-Keller Grid and modification with a square particle in the middle, but the same number of elements.

To simulate polygonal particles with a shape-matching grid, triangular elements generated via Delaunay triangulation will be easier to handle than quadrilateral elements, because for each point distribution, a Delaunay-grid can be constructed (see Fig. 4). In

technical applications, usually for some prevalent flow directions, elements are often chosen longer the direction of the dominant flow. Because we cannot assume any prevalent anisotropic flow between the granular particles, a "good" computational grid should not be distorted, but be as regular as possible, as realized in the AGM-package[3, 4].

There is a philosophy in fluid dynamics of immersed bodies which tries to retain a rectangular grid structure for the fluid simulation at all costs, but introduce the interfaces between the fluid and the particle surface with computer science techniques (Ghost Fluid method, CIP/Multi-Moment, MARS, Level Set method). We refrain from implementing such methods, because each additional interface in a medium introduces a new wave resistance/ mechanical impedance for the signal- or momentum-propagation in that medium, and the scattering properties of the above interfaces are far from clear: In the worst case, it may serve as a source of noise which prevents the build-up of static configurations. Instead, particles form the boundaries of the fluid (see Fig. 1.4), and the additional, unphysical degrees of freedom are a nuisance anyway because they increase the computational complexity and the CPU-time alike. **Chimera-Grids** are popular for simulations of few particles of complicated shape inside a fluid, e.g. for the cross-section of wings, usually in connection with quadrilateral grids; due to our choice of granular particles as polygons and the flexibility of the finite element method, the implementation of Chimera Grids or finer meshes is not necessary.

1.5 Discretization Schemes for the fluid

For the grid part of the simulation, the algorithm with the least degrees of freedom is desirable: The least computation effort for lattice methods (Partial Differential Equations, Lattice fermions ...) for N lattice points can in some cases be obtained for a single timestep within $N \log N$ operations (if Fourier-Type methods can be employed), more realistic for general problems is a cost of N^2 operations per timestep. Therefore, the number of grid points should be reduced as much as possible. This can be accomplished by using larger meshes with higher-order discretization methods and by choosing the particle boundary as boundary of the fluid, so that the corresponding grid points are determined by the granular simulation, and inputted as boundary conditions into the fluid part of the simulation.

Finite Differences have the advantage of being "simple", in the sense that the discretized equations still resemble the original differential equations. The disadvantage is that they work best ("computationally most efficient") for rectangular grids, which don't work well with the polygonal particles we want to use.

The **Finite Volume** method has problems with strongly varying mesh sizes, and also maybe with topological changes of the grid. The whole formalism is considerably more complicated than the finite difference method.

The **Galerkin Finite Element Method** (GFEM) is, as complicated as finite volume method (but not more so, as has been argued in Ref. [26]): it uses the weak form of the PDE, which means that not the original differential equations are approximated, but their integral. This leads to more benign behavior for strongly varying solutions (the integral over a non-differentiable function is likely to be differentiable), but it also introduces additional degrees of freedom (oscillatory solutions additionally to the advective solutions in the original system), which have to be taken care of. Because the GFEM is the most

versatile method for complicated boundaries and unstructured meshes, we will base our simulation of granular particles in fluids on this method. The Navier-Stokes equations in the "strong formulation" can be written as

$$\frac{\partial u}{\partial t} = -\frac{\partial(u^2)}{\partial x} - \frac{\partial(uv)}{\partial y} - \frac{\partial\phi}{\partial x} + f_x + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \quad (1)$$

$$\frac{\partial v}{\partial t} = -\frac{\partial(vu)}{\partial x} - \frac{\partial(v^2)}{\partial y} - \frac{\partial\phi}{\partial y} + f_y + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \quad (2)$$

$$0 = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \quad (3)$$

with the pressure fields p , and the velocity field u, v for the flow in x/y direction, and with dynamic viscosity ν . The last equation, eqn,3, the "continuity equation", assures the incompressibility. as in the "weak formulation", the equations have been integrated over with basis functions (for first/second order finite elements, with polynomials of first/second order respectively) and then integrated out:

$$\int \phi \left(\frac{\partial u}{\partial t} \right) = \int \phi \left(-\frac{\partial(u^2)}{\partial x} - \frac{\partial(uv)}{\partial y} - \frac{\partial\phi}{\partial x} + f_x + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \right) \quad (4)$$

$$\int \phi \left(\frac{\partial v}{\partial t} \right) = \int \phi \left(-\frac{\partial(vu)}{\partial x} - \frac{\partial(v^2)}{\partial y} - \frac{\partial\phi}{\partial y} + f_y + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \right) \quad (5)$$

$$0 = \int \phi \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \quad (6)$$

There is considerable freedom in the choice of elements, i.e. which points on a grid should be integrated out, and in which order the approximation should be performed, as can be seen in Fig. 6. Nowadays, there is a consensus[27] that because the Navier-Stokes equation is of second order in velocity and first order in pressure, the approximation for the velocities should be one order higher than for the pressure, which at least eliminates the $P_1^+ P_1$ -element (first order in velocity, first order in pressure) in Fig. 6(a). This still leaves considerable freedom for second order elements. Because $P_2(P_1 + P_0)$ conserves the element mass, we expect problems due to the volume change of elements induced by moving boundaries (granular particles), so we will not use this element. The $P_2 P_{-1}$ -element has no pressure points on the element boundaries, so that no forces on granular particles which act as element boundaries can be computed, so the use of this element is also excluded. This leaves basically $P_2^+ P_1$ and $P_2 P_1$, the latter being simplest second order element, which we will choose for our implementation.

2 Time Integration

2.1 Time consumption

Meaningful criteria for "the fastest algorithms" are not easy to come by. In the field of Monte-Carlo simulations, "updates per second" (UPS) are a common criterion, albeit not a very meaningful one: Different updating methods come at different computational costs, but lead also to different relaxation times; a meaningful criterion would be the the

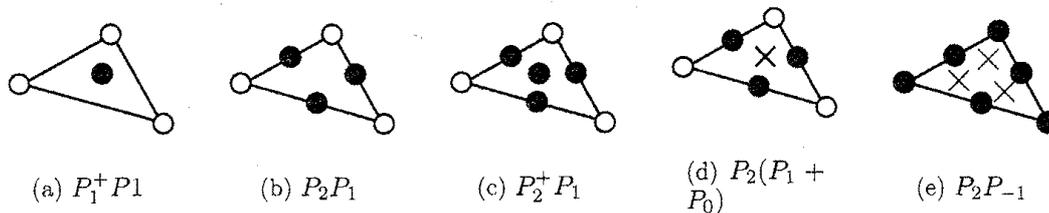


Figure 6: Various elements for fluid flow, where \circ indicates continuous velocities and continuous pressures, \bullet indicates continuous velocities and \times indicates discontinuous pressures.

UPS, divided by the half-life time τ which characterizes the relaxation time. Likewise, for ordinary differential equations, some time integration methods are more costly per time-step, than others, e.g. in general implicit Runge-Kutta methods are more costly than explicit Runge-Kutta methods, though implicit predictor corrector methods (Gear/ Backward Difference/BDF) are not necessarily more costly than explicit (Adams-Bashforth) methods. Higher order Runge-Kutta (RK) methods are more costly than lower methods because they need more function evaluations per timestep than lower order methods (n -th order methods RK methods need at least n function evaluations), whereas for Predictor-Corrector methods usually a single function evaluation per time step is sufficient. Nevertheless, the decisive factor is the maximal size of the time step dt_{\max} (called "radius of stability" in the numerical literature) which can be used, so that a meaningful criterion for the fastest integrator is the the largest possible timestep for a given problem dt , divided by the number of necessary function evaluations. For particles of mass m and Young modulus Y , the characteristic frequency of the corresponding undamped harmonic oscillator is $\omega = \sqrt{Y/m}$. That means that over a wide range of velocities, the contact time T for a single collision can be approximated well as $T = 2\pi/\omega$. We have found that for suitable integrators (Gear-Predictor-Corrector/BDF), the timestep dt_{\max} can be chosen of the order of $dt \approx T/10$. For particles in a fluid, the motion is additionally damped: The contact time becomes larger and larger time steps seem possible. Under the assumptions that the surfaces and the motion granular particles erase vortices of small diameter/ small time scale, larger grid sizes and larger time steps seem possible for the fluid part of the simulation than for the simulation without particles.

2.2 Method of Lines

The discretization in the time direction will not be done by finite elements with a component in the time domain but with the method of lines[28], i.e. the partial time derivative is treated as an ordinary derivative, and the solution can then be obtained with conventional ordinary differential equation solvers, making use of the substantial theory with respect to accuracy and stability of that field[11, 8]. For the simulation of partial differential equations with explicit integrators (linearization of the time evolution), the size of the maximal timestep is usually proportional to the lattice spacing. This criterion based on von Neuman stability analysis (linearization of the time evolution operator for each eigenmode) is usually not valid for implicit integrators, which are derived without linearization and offer therefore additional potential for performance gains.

2.3 Relaxation vs. "numerically exact" methods

In the MAC-method, the time integration is performed without taking the incompressibility into account. Instead, at the new time step, the velocity is equilibrated with the relaxation condition

$$v^{n+1,k+2} = v^{n+1,k+1} - \delta t \nabla (p^{n+1,k+1} - p^{n+1,k}), k \geq 0.$$

This relaxation may take "arbitrarily long" because the non-locality of the Navier-Stokes equation may lead to P-U-V-configurations with very large error; moreover, the relaxation step k does not have the meaning of a real time, but is unphysical.

Nobody would treat a particle on a string/ rigid pendulum with a "relaxation dynamics" where first the pendulum is advanced to a point which corresponds to a change of the length, and afterwards the length is adapted again. Instead, the "numerically exact" treatment of the time integration which conserves the constraint with "zero error" is the Lagrange multiplier formalism in Fig. 7.

<p>For a pendulum of mass m of unit length at position $\mathbf{q} = (\mathbf{x}_1, \mathbf{x}_2)$ and unit length, the constraint equation (center at the</p> $g(\mathbf{q}) = \frac{1}{2} (\mathbf{q} \cdot \mathbf{q} - 1) = 0.$ <p>Additional equations follow for</p> $\dot{g}(\mathbf{q}) = \mathbf{q} \cdot \dot{\mathbf{q}} = 0,$ $\ddot{g}(\mathbf{q}) = \ddot{\mathbf{q}} \cdot \mathbf{q} + \dot{\mathbf{q}} \cdot \dot{\mathbf{q}} = 0,$ <p>by taking the derivative of the constraint $g(q)$ with respect to time. From Newton's equation of motion follows that the acceleration $\ddot{\mathbf{q}}$ depends on the constraint force \hat{f} and the external forces f as</p> $\ddot{\mathbf{q}} = f + \hat{f}/m.$	<p>Inserting this in the equation for $\ddot{g}(\mathbf{q})$, one obtains</p> $\ddot{g}(\mathbf{q}) = \frac{f + \hat{f}}{m} \cdot \mathbf{q} + \dot{\mathbf{q}} \cdot \dot{\mathbf{q}} = 0.$ <p>From the principle of virtual work (constraint forces may not perform work on a system) follows that</p> $\hat{f} \cdot \dot{\mathbf{q}} = 0,$ <p>so that the constraint force \hat{f} must be parallel to the coordinate vector \mathbf{q}, i.e. $\hat{f} = \lambda \mathbf{q}$, with the Lagrange multiplier λ so that</p> $\lambda = \frac{-f \cdot \mathbf{q} - m \dot{\mathbf{q}} \cdot \dot{\mathbf{q}}}{\mathbf{q} \cdot \mathbf{q}} \quad (7)$
--	---

Figure 7: Lagrange Multiplier Formalism for a single pendulum

2.4 Generalized Formulations

In the above example for the single pendulum, the Newton equations of motion have been rewritten from a second-order equation to first order equations

$$\begin{aligned} M\dot{u} &= f(q, u), \\ \dot{q} &= u, \end{aligned}$$

for the masses M coordinates q , velocities u and forces $f(q, u)$. The constraints $g(q)$ were introduced via their Jacobian $G^T(q) = \nabla g(q)$,

$$\begin{aligned} M\dot{u} &= f(q, u) - G^T(q)\lambda \\ \dot{q} &= u \\ 0 &= g(q), \end{aligned}$$

where, λ is the vector of Lagrange multipliers and M must be invertible. This differential algebraic equation (DAE, ordinary differential equation with algebraic constraints) can be rewritten in a so-called index-1 formulation as

$$\begin{pmatrix} M & G^T(q) \\ G(q) & 0 \end{pmatrix} \begin{pmatrix} u' \\ \lambda \end{pmatrix} = \begin{pmatrix} f(q, u) \\ -g_{qq}(u, u), \end{pmatrix}$$

where M must not be invertible. It can be shown[27] that the GFEM-discretization of the Navier-Stokes equation with implicit-Euler-time discretization takes the form

$$\begin{bmatrix} \frac{1}{\Delta t_n}M + K + N(u_{n+1}) & C \\ C^T & 0 \end{bmatrix} \begin{pmatrix} u_{n+1} \\ P_{n+1} \end{pmatrix} = \begin{pmatrix} \frac{1}{\Delta t_n}Mu_n + f_{n+1} \\ g_{n+1} \end{pmatrix}.$$

In other words, one sees that in the generalized index-1 formulation for DAE's, the pressures in the incompressible Navier-Stokes equations take the role of Lagrange-multipliers. Similar forms are obtained for other time integrators, see Ref. [27] The choice of initial conditions for the DAE's is much more complex than for Ordinary Differential Equations (ODE's). For the ordinary differential equations of a spring moving in two dimensions, any initial conditions can be specified; for the rigid pendulum in eq. 7, only initial conditions make sense where the absolute value of the position is equal to the length of the pendulum, and the velocity vector is orthogonal on the position vector. If these initial conditions are not given, the solution diverges rapidly (within few time-steps) towards infinity, DAE's need consistent initial conditions, in contrast to ODE's. For the solution of the Navier Stokes equation, this means that for initial state the continuity equation must be fulfilled. The initial condition can be computed as the solution of the stationary Navier Stokes equation via Newton-Raphson iteration (details in the part II of this paper).

3 Summary and Conclusions

For polygonal/polyhedral granular particles, we have shown that there is a geometrically ideal discretization of the surrounding incompressible flow in the framework of triangular Galerkin Finite Elements. Though this approach is geometrically more demanding than the conventional pairing of quadrilateral grids with round particles, one is rewarded with a much more straightforward combination of particles and fluid. The time integration performed with the method of lines yields differential algebraic equations with the pressure as Lagrange-parameter, so that no pressure iteration for the incompressible Navier Stokes equations is necessary. The whole concept will be only fruitful if implemented on unstructured grids, so the whole concept hinges on the availability of "cheap" and accurate solution of the nonlinear system in each timestep. Such a solution method will be discussed in the following article.

(References at the end of the second part of the article on the following pages)