Modelling of spheroidal particles in viscous flow

T. Kempe, S. Schwarz, J. Fröhlich

Institute of Fluid Mechanics
Dresden University of Technology
Dresden, Germany

Academy Colloquium IB Methods, Amsterdam 2009 / 06 / 16
Motivation

- Goal: DNS of flow with heavy and light “particles”
  e.g.
  → Sedimentation processes
  → Particle transport
  → Cyclones, ....

- Production runs with very many particles

Here

- Method of Uhlmann (2005) fails
  for density ratios ~ 1 → solve that

- Extension to ellipsoidal particles

Formation of sand ripples under turbulent water flow

Turbulence over granular bed
[Stoesser, Fröhlich, Rodi 2005]
PRIME (Phase-resolving Simulation Environment)

Basic Fluid Solver

- Incompressible Navier-Stokes equations
  \[ \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \frac{1}{\rho} \nabla p = \nu \nabla^2 \mathbf{u} + \mathbf{f} \]

- 2\textsuperscript{nd} order Finite Volumes
- Staggered Cartesian grid
- Explicit 3-step Runge-Kutta for convective terms
- Implicit Crank-Nicolson for diffusive terms
- Parallelization
  → Domain decomposition
  → Libraries PETSc and HYPRE

Speedup PRIME (channel flow).
- solid: 8.4 Mio gridpoints
- dash-dot: 4.2 Mio gridpoints
IBM [Uhlmann, 2005]

- Outline of method:
  - Phase coupling by additional volume forces
  - Interpolation of velocities to marker points
  - Direct Forcing [Fadlun 2000]
    \[ F = \frac{U_r - U_i}{\Delta t} \]
  - Spreading to gridpoints via regularized \( \delta \)-functions [Peskin 1977]

- Advantages:
  - extremely simple
  - good stability
  - order of convergence about 1.7

T. Kempe, S. Schwarz, J. Fröhlich
Immobile sphere in channel flow

- Re$_b$ = 540 ; Re$_d$ = 130
- 128 x 128 x 128 gridpoints
- 3020 Marker points on sphere

- PRIME, MGLET (TU Munich), SUSPENSE [Uhlmann 2005]

u in x - direction

u in y - direction

Flow around a fixed sphere, Re=130, BC u=0 is verified
Mobile particles

- Linear momentum balance:
  \[ m_p \frac{d\mathbf{u}_p}{dt} = \rho_f \int_\Gamma \mathbf{\tau} \cdot \mathbf{n} \, dS + (\rho_p - \rho_f) V_p \mathbf{g} \]

  Pressure & viscous forces  buoyancy

- Angular momentum balance:
  \[ I_c \frac{d\mathbf{\omega}_p}{dt} = \rho_f \int_\Gamma \mathbf{r} \times (\mathbf{\tau} \cdot \mathbf{n}) \, dS \]

- Runge–Kutta (3rd order) for time integration

- Fully parallel implementation in PRIME

- Master & Slave strategy
Sedimentation of a single particle

- Experiment [Mordant, Pinton 2001]
- 128 x 1024 x 128 grid points
- 874 marker points on surface
- Sedimentation velocity:

874 Marker points on sphere [Leopardi 2006]

\[
\text{Re}_p = 41 \quad \text{Re}_p = 280
\]

→ Good prediction of the particle dynamics

T. Kempe, S. Schwarz, J. Fröhlich
Original Method: Problems with $\rho_p \approx \rho_f$

- Direct evaluation of viscous forces costly and imprecise
- Smoothed delta-function smears gradients
- Better: Viscous forces from fluid motion

**Particle momentum:**

$$m_p \frac{d\mathbf{u}_p}{dt} = \rho_f \int \mathbf{\tau} \cdot \mathbf{n} dS + (\rho_p - \rho_f)V_p \mathbf{g}$$

**Fluid momentum:**

$$\rho_f \frac{d}{dt} \int \mathbf{u}_f dV = \rho_f \int \mathbf{f} dV + \rho_f \int \mathbf{\tau} \cdot \mathbf{n} dS$$

**Rigid body approximation:**

$$\rho_f \frac{d}{dt} \int \mathbf{u}_f dV \approx \rho_f V_p \frac{d\mathbf{u}_p}{dt}$$

**Final particle momentum:**

$$\frac{d\mathbf{u}_p}{dt} = \frac{\rho_f}{V_p(\rho_p - \rho_f)} \int \mathbf{f} dV + \mathbf{g}$$

- Singularity for $\rho_p = \rho_f$ in original method
- Numerical problems for $\rho_p \approx \rho_f$ and unstable for $\rho_p < \rho_f$
Modified scheme for $\rho_p \approx \rho_f$

- As before:

- Now evaluation of integral via quadrature

- Need volume fraction $\alpha$

**Option 1:** analytical

\[
\rho_f \int_{\Gamma} \mathbf{\tau} \cdot \mathbf{n} \, dS = \rho_f \frac{d}{dt} \int_{\Omega} \mathbf{u}_f \, dV + \rho_f \int_{\Omega} \mathbf{f} \, dV
\]

\[
\int_{\Omega} \mathbf{u}_f \, dV \approx \sum_{i} \sum_{j} \sum_{k} u_{i,j,k} \, V_{i,j,k}^{cell} \alpha_{i,j,k}
\]

**Option 2:** cut-cell

→ Expensive for 3D moving boundaries
Method here:

- Signed-distance level set of corner points
- Volume fraction approximated by

\[ \alpha_{i,j,k} \approx \frac{\sum_{m=1}^{8} - \phi_m H(-\phi_m)}{\sum_{m=1}^{8} \|\phi_m\|} \]

\( \phi \) – Level Set
\( H \) – Heaviside function

- Exact for:
  - \( \alpha = 0 \ldots 1 \)
  - \( \alpha = 0.5 \)
  - \( \alpha = 0 \) or \( \alpha = 1 \)
Application to sphere

- Signed-distance level set function

\[ \phi_{i,j,k} = \sqrt{(x_{i,j,k} - x_p)^2 + (y_{i,j,k} - y_p)^2 + (z_{i,j,k} - z_p)^2} - r_p \]

- Validation: Volume of a sphere in comparison to analytical solution

<table>
<thead>
<tr>
<th>D / ( \Delta x )</th>
<th>( V_{an} - V_{num} )</th>
<th>Relative Error [%]</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>5.239e-3</td>
<td>1.954</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1.374e-3</td>
<td>5.12e-1</td>
<td>1.96</td>
</tr>
<tr>
<td>40</td>
<td>3.474e-4</td>
<td>1.29e-1</td>
<td>2.00</td>
</tr>
<tr>
<td>80</td>
<td>8.520e-5</td>
<td>3.18e-2</td>
<td>2.02</td>
</tr>
<tr>
<td>160</td>
<td>2.157e-5</td>
<td>8.05e-3</td>
<td>1.99</td>
</tr>
<tr>
<td>320</td>
<td>5.608e-6</td>
<td>2.09e-3</td>
<td>1.98</td>
</tr>
<tr>
<td>640</td>
<td>1.327e-6</td>
<td>4.95e-4</td>
<td>2.05</td>
</tr>
</tbody>
</table>

\[ V_{ex} = \frac{4}{3} \pi r^3 \]

\[ V_{num} = \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \sum_{k=1}^{N_k} (\Delta x)^3 \alpha_{i,j,k} \]

\( \rightarrow \) 2nd order convergence

T. Kempe, S. Schwarz, J. Fröhlich
Results for settling and buoyant spheres

Velocity of buoyant & sedimenting particles

→ Stable time integration for previously inaccessible density ratios

Symbols: Experiment [ten Cate 2002]
Performance

- Vertical channel flow
- 16.8 Mio gridpoints
- 742 forcing points on each sphere
- \( D/\Delta x_i = 15.4 \)
- 32 processors on SGI Altix 4700

<table>
<thead>
<tr>
<th># Particles</th>
<th>( t_{\text{fluid solver}} ) [s]</th>
<th>( t_{\text{particles}} ) [s]</th>
<th>( t_{\text{particles}} ) [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>4.826</td>
<td>0.067</td>
<td>0.085</td>
</tr>
<tr>
<td>500</td>
<td>4.830</td>
<td>0.130</td>
<td>0.194</td>
</tr>
<tr>
<td>2000</td>
<td>4.943</td>
<td>0.404</td>
<td>0.589</td>
</tr>
</tbody>
</table>

→ Numerically efficient scheme
Arbitrary Geometry – Forcing Points

- Equidistant partition of surface cumbersome
- Distribution of forcing points either by
  - Sphere: Equi-Sphere Partitioning [Leopardi 2006]
  - Explicit prescription „by hand“ and triangulation
  - Advancing front algorithm (interface to Gambit)

- Within spreading of forces ensure that
  \[ \sum_{L=1}^{N_f} F_L \Delta V_L = \sum_{i,j,k} f_{i,j,k} \Delta V_E \]
  where \( \Delta V_E = \Delta x^3 \) for unstretched Euler grid

- For each forcing point determine (shell–element of thickness \( \Delta x \))
  \( \Delta V_L = A_L \Delta x \)
Arbitrary Geometry – Eq. of Motion

- Solve equation of motion in laboratory system
- Transformation of tensor of inertia

\[ I_p = A I_{p,0} A^{-1} \]

with rotation matrix

\[ A = A(\phi_{x,y,z}) \]

- Runge-Kutta method for

\[ \frac{d(I_p \omega_p)}{dt} = T \]

with

\[ I_p(t) = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix} \]

- Direct solution of linear system for \( \omega_p \)
- Calculate new orientation \( \Phi_{x,y,z} = f(\omega_p) \)
Vortex Structures

- Wake structures and associated trajectory

**Straight**

- PRIME, $y$-velocity, $Re_a \approx 100$

**Zig-Zag**

- PRIME, Vorticity $\omega_y$, $Re_a \approx 200$

**Spiral**

- PRIME, Vorticity $\omega_y$, $Re_a \approx 300$, Rotation around fixed position

**Zig-Zag, rocking, chaotic**

- Air-Water, Schlieren optics [Veldhuis, 2008], $Re \approx 1500$

- PRIME, Vorticity $\omega_y$, $Re \approx 4000$
Fixed swinging ellipsoid

Problem setup

Rotation angle $\Phi_z$ vs. time, $Re_{eq} = 150$, aspect ratio = 2.5

Iso-contours of streamwise vorticity $|\omega_x| = 3$, $Re_{eq} = 200$, aspect ratio = 2.5

→ Zig-Zag trajectory generated by wake vortices
Under way

→ Contact modelling

→ Sediment erosion

Thanks for your attention.

Questions?
Arbitrary Shape

A) time-dependant

- Forcing points with *prescribed* oscillation in aspect ratio
- Velocity $u_{osc}$ of the interface forced within IBM

\[
    \mathbf{u}^{(l)}_{osc} = \frac{\partial \mathbf{x}^{(l)}}{\partial t}
\]

B) Very complex

- Data structure allows arbitrary shapes
- feasibility study: Car in crossflow

T. Kempe, S. Schwarz, J. Fröhlich