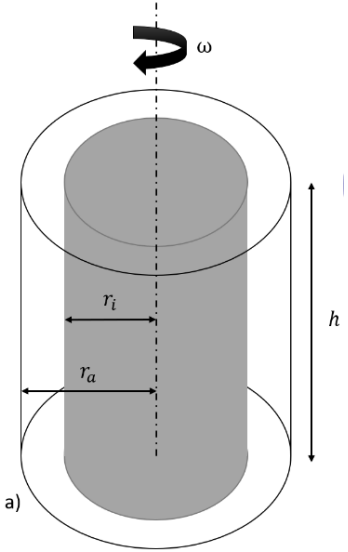
Best practice für Semi-Resolved CFD-DEM

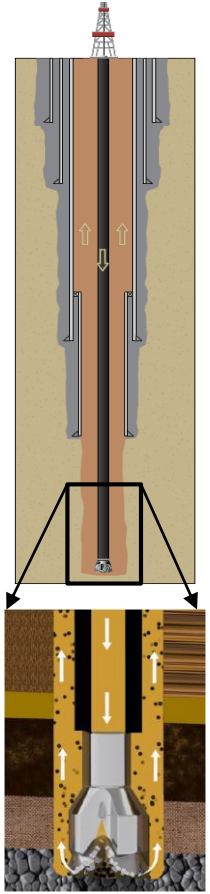
Kathrin Skinder, TU Clausthal

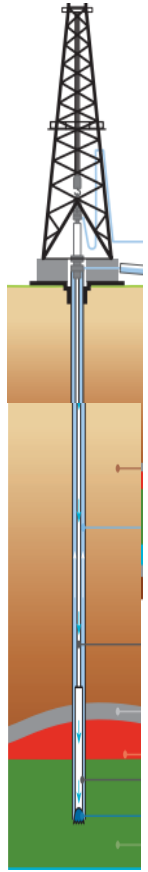
OpenFOAM Stammtisch
15. April 2026

Hintergrund: Bohrkleintransport

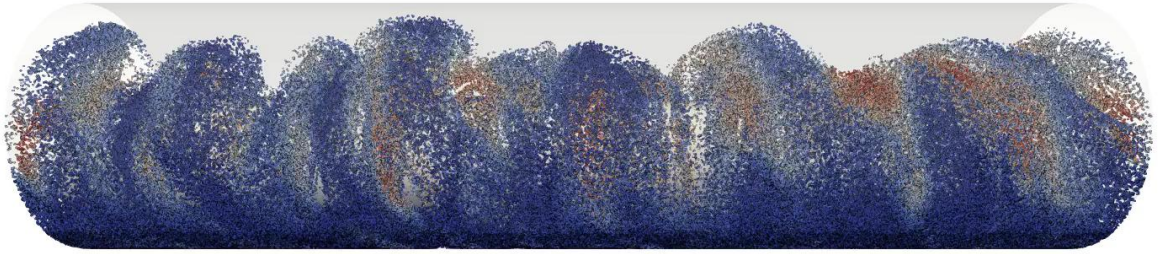
- Partikelbeladene Strömung im Ringspalt
- Gesteinspartikel im Bereich von µm bis zu ~10 mm
- Fragestellungen:
  - Bohrkleinaustrag sicher/effizient
  - Integrität der Bohrung (Druckfenster)



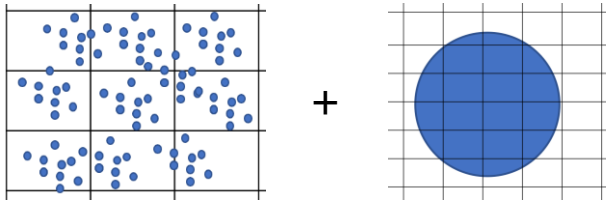




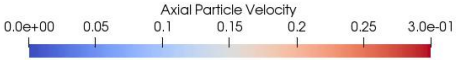
Modellierungsansatz OpenFOAM v8

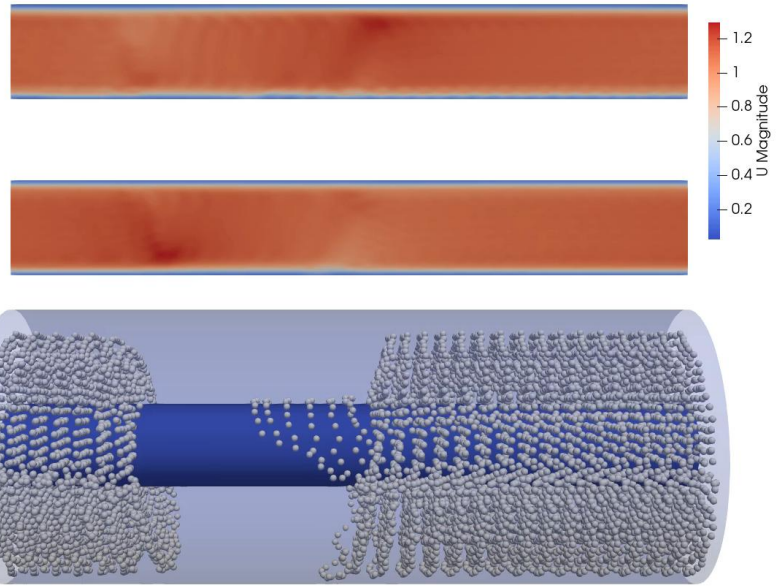


- Euler/Lagrange, MPPICFoam (bis $\frac{d_{particle}}{d_{cell}} = \frac{1}{4}$)
- Immersed Boundary (ab $\frac{d_{particle}}{d_{cell}} = 10$)
- Vision: Kombination beider Ansätze



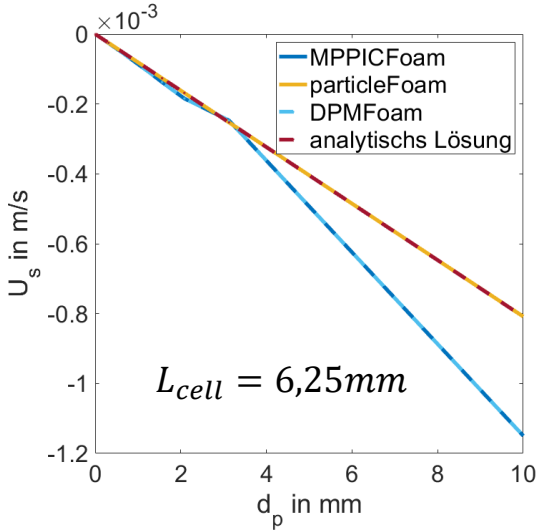
  - Breite Partikelgrößenverteilung in der Tiefbohrtechnik
  - „Lücke“ im Semi-Resolved Bereich $\frac{1}{4} < \frac{d_{particle}}{d_{cell}} < 10$

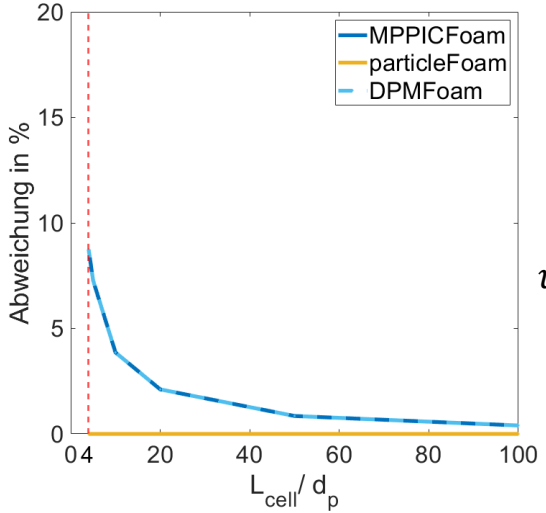




Problem: Interessenkonflikt zwischen Euler und Lagrange

- Für Euler-Phase ist ein möglichst feines Rechengitter nötig
- Für Lagrange-Phase müssen Partikel ca. 4x kleiner sein, als ein Gitterelement

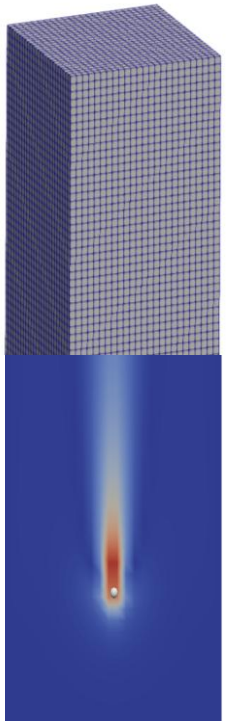




Sedimentationsmodell (Stokes)

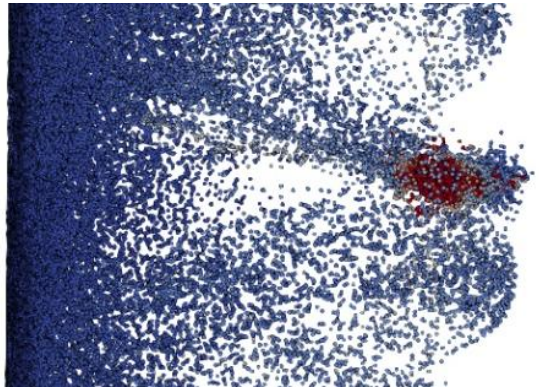
$$u_s = \frac{g\, {d_p}^2 (\rho_p - \rho_f)}{18\, \upsilon\, \rho_f}$$

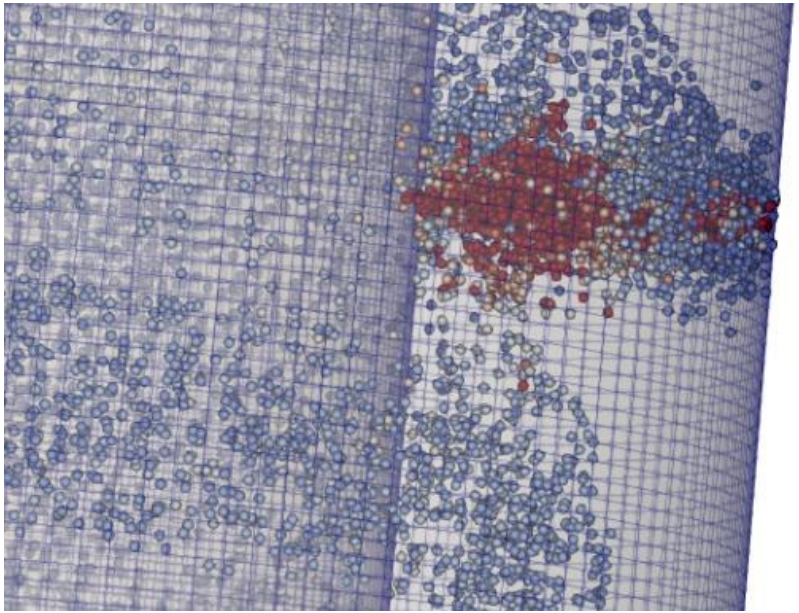
$$Re \ll 1$$



Konsequenzen

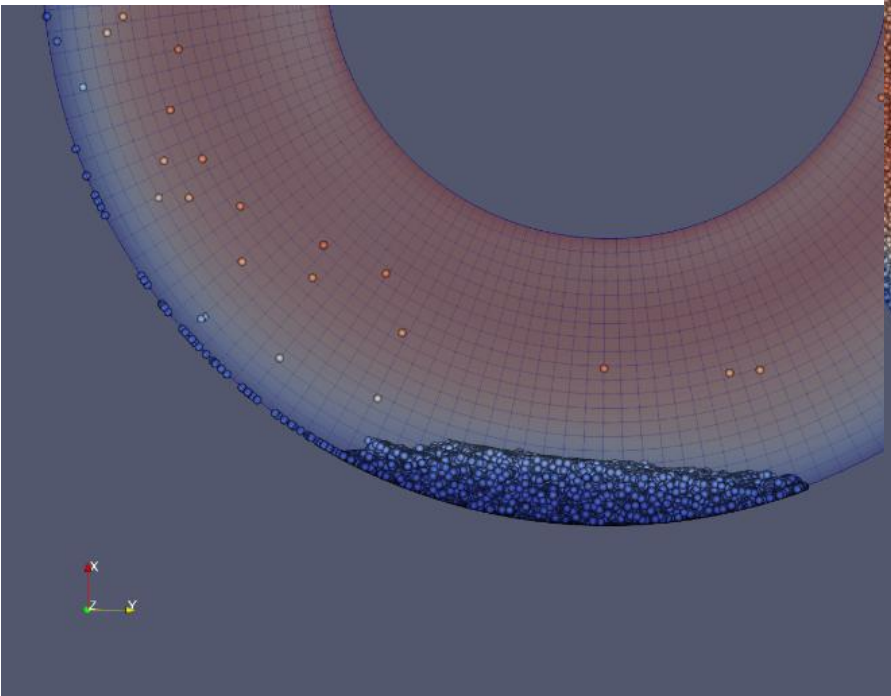
- Überschätzung der Partikelgeschwindigkeit
- Lokale unphysikalische Geschwindigkeits-Peaks und Cluster

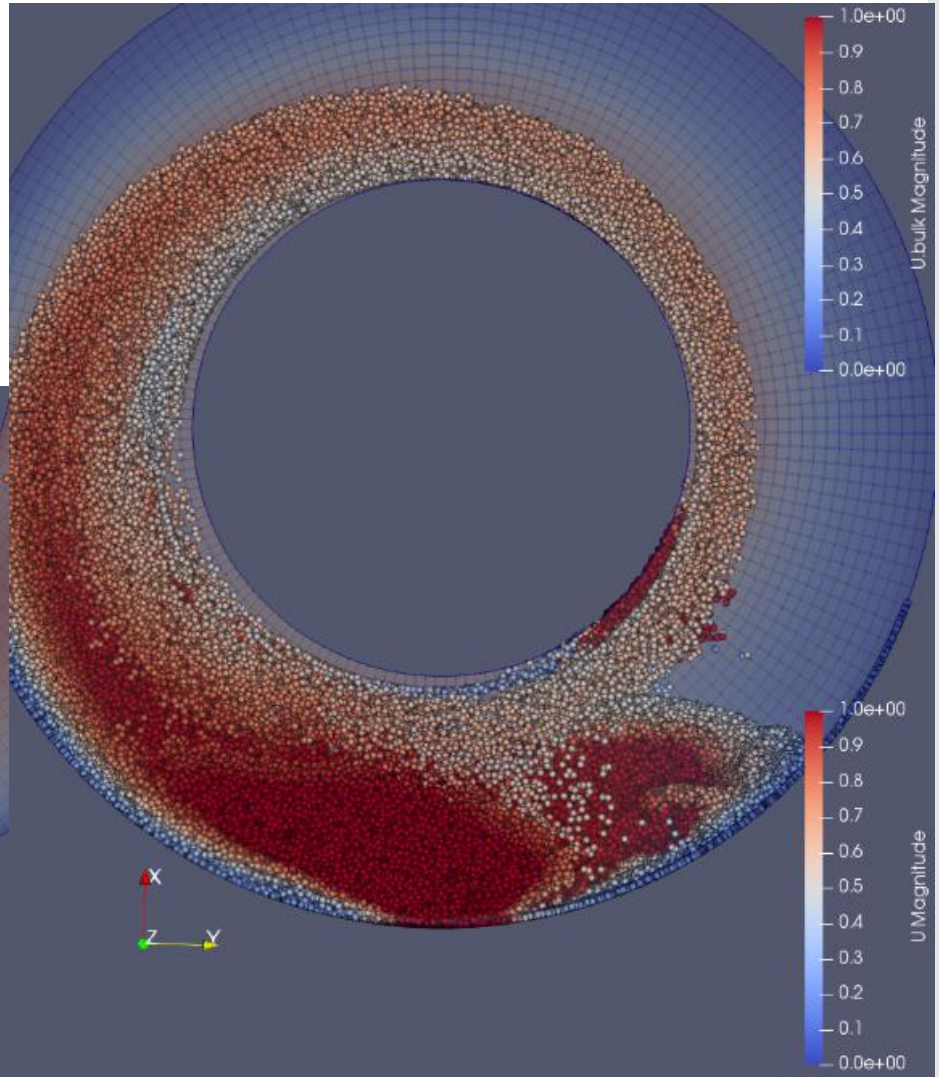




Konsequenzen

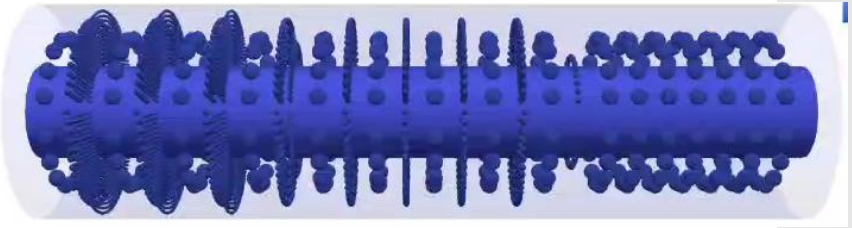
- Links: stärkere Rotation
- Rechts: geringere Rotation

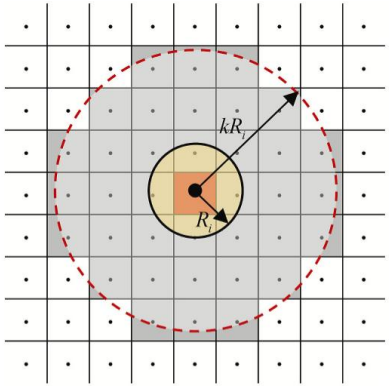




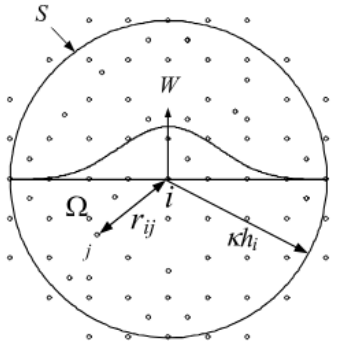
„Lösung“: Semi Resolved (Kernel based averaging/big particle models)

- CFDEM coupling (OpenFOAM + Liggghts)
- Custom implementations?
- Do‘s / Dont‘s?
- Validierung?





Wu et al., An improved semi-resolved CFD-DEM method for particle systems with wide mesh/particle size ratios, Computers and Geotechnics, 2026



Wang et al., A semi-resolved CFD–DEM approach for particulate flows with kernel based approximation and Hilbert curve based searching strategy,Journal of Computational Physics, 2019

Problem2: Periodische Ränder + DPMFoam

UcEqn.H

```
fvVectorMatrix UcEqn
(
    fvm::ddt(alphac, Uc) + fvm::div(alphaPhic, Uc)
  - fvm::Sp(fvc::ddt(alphac) + fvc::div(alphaPhic), Uc)
  + continuousPhaseTurbulence->divDevTau(Uc)
 ==
    (1.0/rhoc)*cloudSU
  + fvOptions(Uc)
  //+ alphac*fvOptions(Uc)
);
```

constant/fvOptions

```
meanVelocityConstraint
{
    type            meanVelocityForce;
    active          yes;

    selectionMode   all;
    fields          (U.bulk);
    patch           inlet;
    Ubar            (0 0 0.5);
    //relaxation      0.2;
}
```

- Impulsquellterm fvOptions(Uc) = GradP
- Multiplizieren mit alphac ändert den berechneten Druckgradienten GradP
- Was ist richtig?
- Ideen zur Validierung?