Adaptive Grid Refinement and Load Balancing Techniques for Particulate Flow Simulations

Christoph Rettinger, Ulrich Rüde
Friedrich-Alexander-Universität Erlangen-Nürnberg, Chair for System Simulation

Motivation: Decrease Time to Solution

- Bulk region: coarse resolution
- Interface & particle region: fine resolution

- Only some regions of the computational domain require a fine resolution, others can be coarser
- Often, these regions change dynamically
- Refining / coarsening changes the workload
- Workload varies locally and temporally in multiphysics simulations

Adaptive Grid Refinement

- Block-structured refinement in waLBerla [1,2]
- Refinement criteria
  - Coupling-based: finest grid around particles
  - Fluid-based: scaled vorticity magnitude

Example: Settling Sphere
- Reduce time to solution from 7h (8192 fine blocks) to 27min (ca. 160 fine blocks)
- Same results are obtained

Load Balancing Results

Setup: Hopper Clogging
- 4300 particles settle in hopper and clog the narrow bottom
- Uniform grid with $12 \times 12 \times 16$ blocks of size $32^3$ cells
- 256 processes used on SuperMUC cluster (LRZ, Munich)
- Load balancing is triggered every 100 time steps

- Using fitted load predictor with space-filling curves reduces time to solution by 15% compared to standard one (constant workload)
- Space-filling curves yield better results than ParMETIS
- Load balancing overhead is 2.7% of total runtime

Load Prediction

- Goal: try to find a function that predicts the workload generated by a block, depending on block-local information
- total cells $C$, fluid cells $F$, near-boundary cells $NB$, local particles $P_L$, shadow particles $P_S$, contacts $K$, sub steps $S$

1. Define fit functions for the different parts of the algorithm, here the coupled Lattice Boltzmann method with particle simulation [3]

2. Perform simulations of characteristic setup to generate data
   - Horizontally periodic box filled with randomly placed particles
   - Settling leads to dense packing at bottom wall

3. Apply curve fitting to determine coefficients and combine functions to obtain complete load predictor

Load Distribution

- Task: distribute the predicted workload evenly among the available processes
- Space-filling curves, like Hilbert or Morton
- ParMETIS library [4]
  - Different graph partitioning algorithms available
  - Optimizes communication (edge cut) and load distribution

References

[1] open source simulation framework waLBerla (walberla.net)
[4] ParMETIS - Parallel Graph Partitioning and Fill-reducing Matrix Ordering