

# Universität Hamburg Projekt 2016 Particle Simulation

Olliver Heidmann Benjamin Warnke

February 4, 2017

## 1 Projektplan

### 1. startparameter (Oliver)

- verbose
- algorithmus
- autotuneing on/off
- timestep gröÙe
- abstand der datei speicherung
- optimierungsmethode festlegen (nur ohne autotuneing)
- seed für random

### 2. prototypen/interface für Optimierte Datenstrukturen(Oliver)

- insert(Particle):void
- next():Particle
- neighbours(Particle):Particlelist
- zusätzliche? weniger?
- dies würde die austauschbarkeit enorm vereinfachen ... autotuneing müsste nur noch eine variable ändern
- gemeinsam treffen, nachdem die vorherigen aufgaben abgeschlossen sind.

### 3. Debug+Benchmark funktionen (Benjamin)

### 4. importieren/generieren der Startdaten (Benjamin)

- aus Datei (Optional)
- generieren nach bestimmten verteilungen für tests, welches verfahren wann am besten ist
  - kugel in der mitte
  - verteilte kugeln jeweils gleichmäßig gefüllt
  - gleichmäßig
  - pseudorandom
  - wenige partikel
  - viele partikel
  - weit auseinander
  - eng zusammen
  - generieren nach mustern

### 5. implementierung der Lennard-Jones-Simulation (Oliver)

- lesen im kapitel des buches?!?
- was für daten werden benötigt?
  - distanz zum anderen

6. ausgabe der Daten (Benjamin)

- (a)
  - eigenes txt format für tests
  - exportieren in bekannte Dateiformate (mindestens 1)
    - LAMMPS
    - ESPRESSO
    - GROMACS
    - VMD
    - ParaView/vtk
- (b)
  - OpenGL (optioal)

7. autotuneing + analyse

- als Grid (Benjamin)
- mit Listen der Nachbarn (Oliver)
- gruppen zusammenfassen und nur den Schwerpunkt berechnen
- welches verfahren sollte "gelöscht" werden -> wiso?
- welches verfahren ist am besten -> wiso? abhängig von der eingabe??
- kriterien für analyse?
- eingaben kategorisieren um analysieren zu können

8. optimierung

- funtions Verfahren
- openmp
- Intel TBB (optioal)
- cuda (optional)
- opencl (optional)
- mpi (optional)

9. andere Simulationsverfahren

- Smoothed Particle Hydrodynamics
- Dissipative Particle Dynamics