

Molecular Dynamics - Assignment 3

Alex Hocks Jan Hampe Johannes Riemenschneider

Technische Universität München

TUM CIT

Lehrstuhl für wissenschaftliches Rechnen

7. Dezember 2022

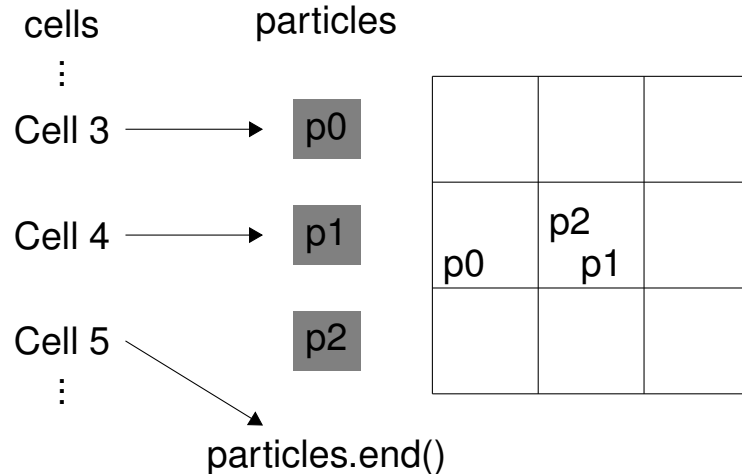


TUM Uhrenturm

The Cell Data-Structure - Approach 1

Idea:

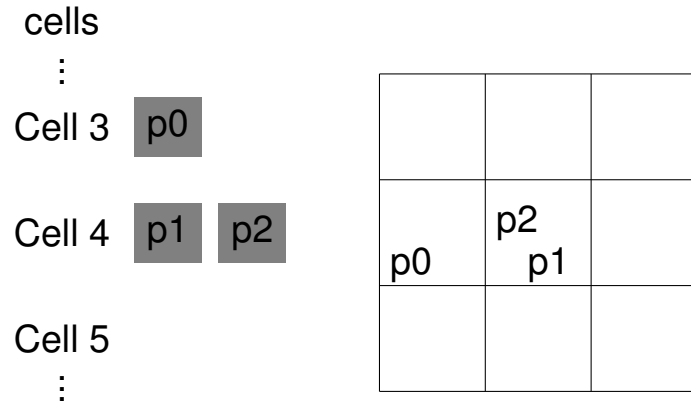
- Sort Particles in accordance to their Cell Position
- save which part of the particles-Vector corresponds to which cell



The Cell Data Structure - Approach 2

Idea:

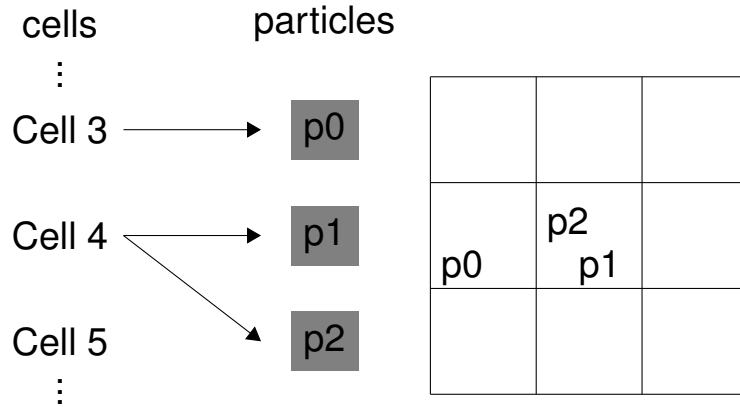
Approach 1.1 stored multiple virtual vectors in one vector → let's actually store the particles in vectors corresponding to their cell



The Cell Data Structure - Approach 3

Idea:

- Each Cell only keeps references to their members
- No sorting or copying of entire particles required



Approach Comparison

Approach 1	Approach 2	Approach 3
<ul style="list-style-type: none">+ Easy to implement+ Interface for old Assignments remains unchanged– Expensive struct swaps during sorting+ Direct access to particles for calculations	<ul style="list-style-type: none">+ Easy to implement+ New Implementation of some methods needed– Expensive struct copies with potential reallocs needed+ Direct access to particles for calculations	<ul style="list-style-type: none">+ Easy to implement– Interface for old Assignments remains unchanged+ References are cheap+ Dereferencing needed

Approach Comparison

Approach 1	Approach 2	Approach 3
<ul style="list-style-type: none">+ Easy to implement+ Interface for old Assignments remains unchanged– Expensive struct swaps during sorting+ Direct access to particles for calculations	<ul style="list-style-type: none">+ Easy to implement+ New Implementation of some methods needed– Expensive struct copies with potential reallocs needed+ Direct access to particles for calculations	<ul style="list-style-type: none">+ Easy to implement– Interface for old Assignments remains unchanged+ References are cheap+ Dereferencing needed

In the end we decided to implement approach 3. ↴

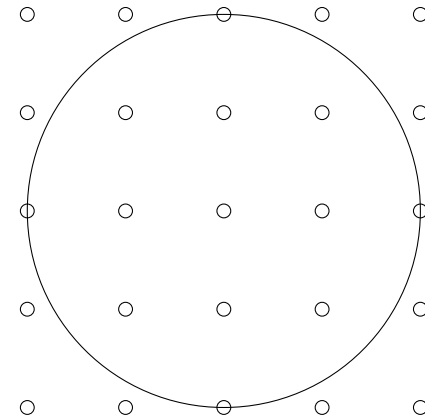
Spheres

Expansion of the Body-struct utilized in Assignment 2

```
enum Shape {cuboid , sphere};
struct Body {
    Shape shape;
    Eigen::Vector3d fixpoint;
    Eigen::Vector3d dimensions;
    double distance;
    double mass;
    Eigen::Vector3d start_velocity;
} ;
```

$$\sqrt{x^2 + y^2 + z^2} \leq r$$

$$\iff x^2 + y^2 + z^2 \leq r^2$$

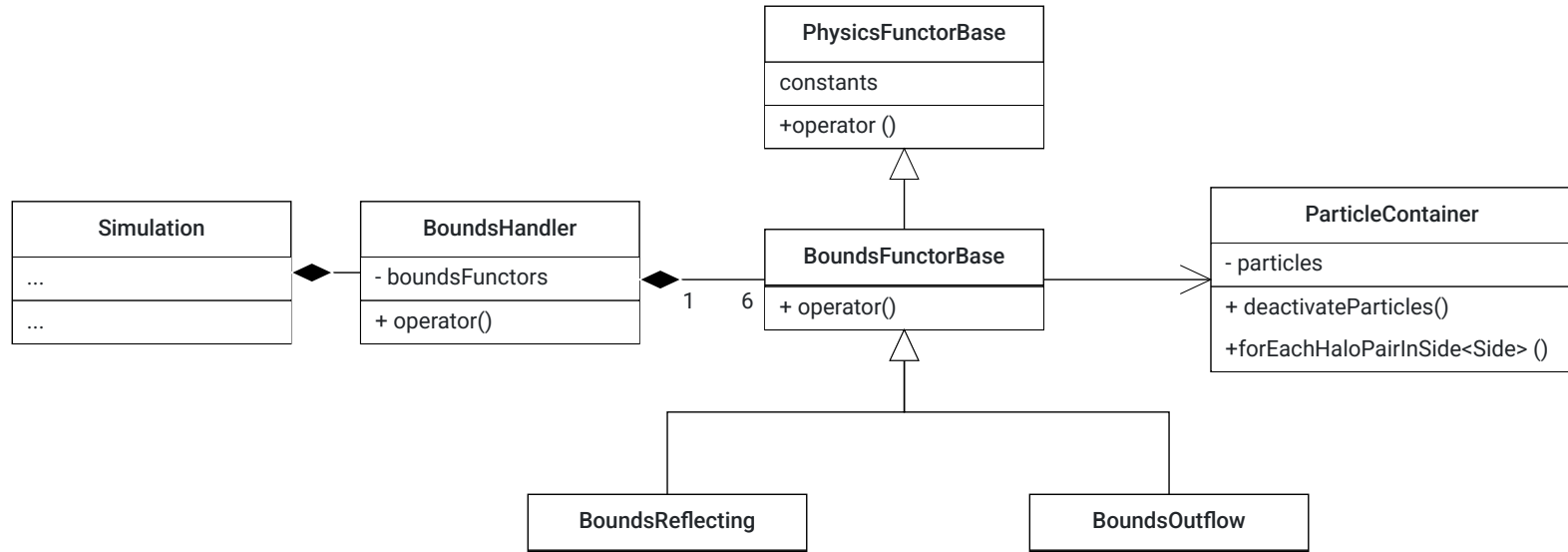


ParticleContainer's new methods

```
class ParticleContainer{  
    ...  
  
    void forAllPairsInSameCell(void (*fun) (Particle& p1, Particle& p1));  
  
    void forAllPairsInNeighbouringCell(void (*fun) (Particle& p1, Particle& p1));  
  
    void forAllCells(void (*fun)(...));  
  
    void forAllDistinctNeighbouringCells(void (*fun) (...));  
  
    ...  
}
```

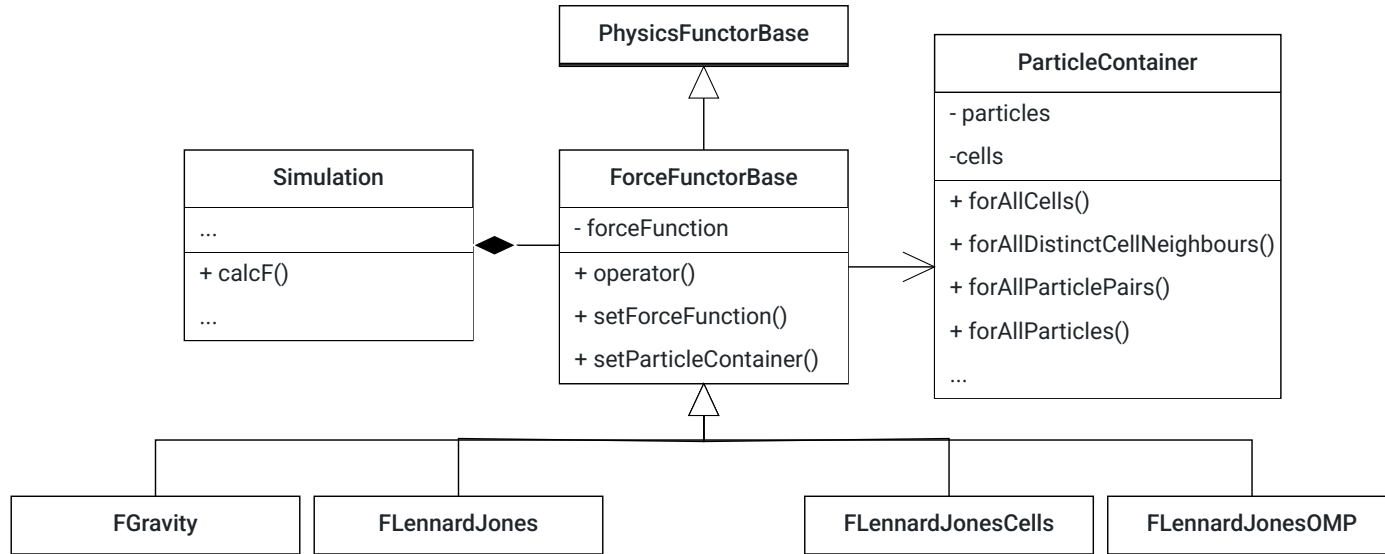
- Functionality of first two methods is sufficient but hard to optimize
- Functionality of last two methods results in higher cohesion, but potential for runtime improvement

Bounds Handling

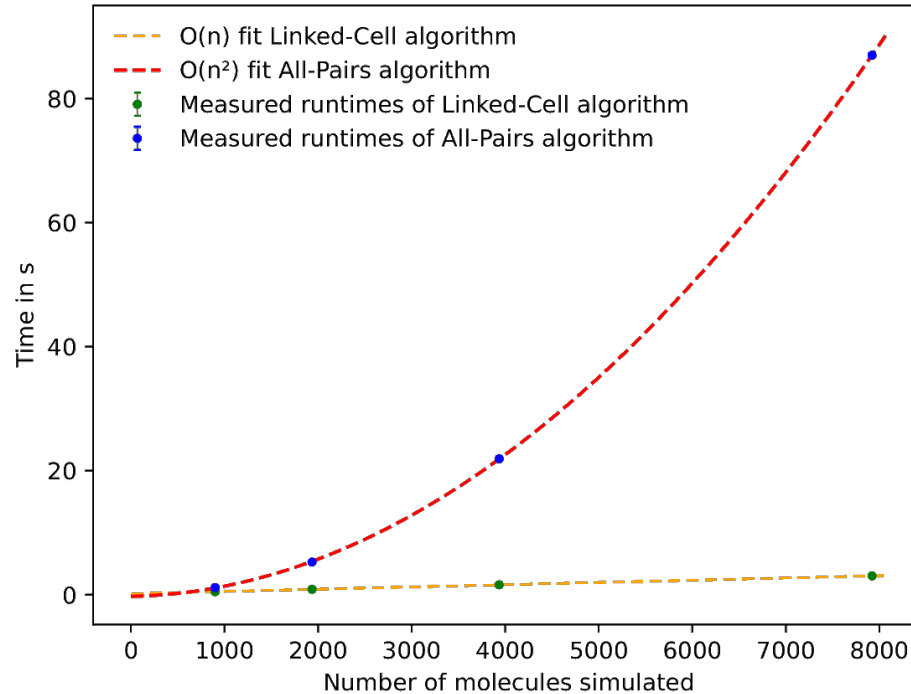


ForceFuncctors

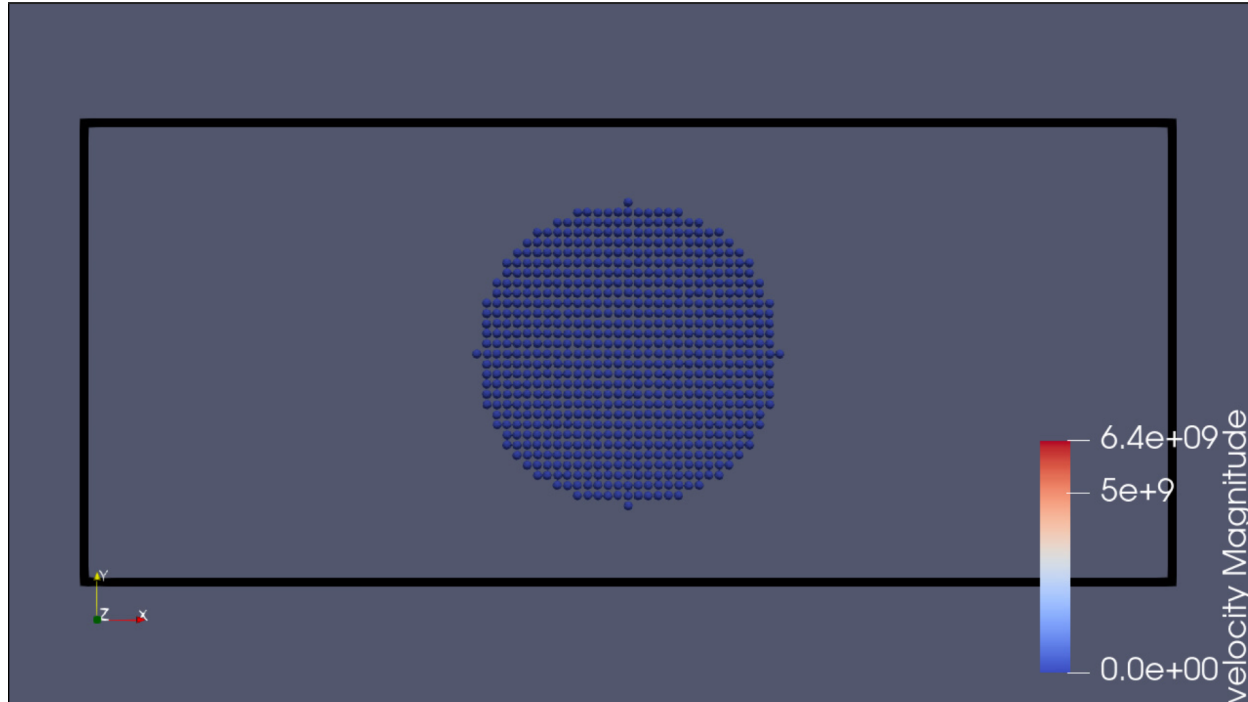
- Force function used gets determined on runtime via input parameters
- Force functor defines the algorithm (Linked-Cell algorithm/ All-Pairs algorithm) used



Runtime Comparison of different algorithms



Our Simulation



We were just kidding

Of course we've had a Simulation that actually
worked

Of course we've had a Simulation that actually
worked

...

Of course we've had a Simulation that actually
worked

...

right?

Our actual Simulation

