GHENT
UNIVERSITY

FACULTY OF ENGINEERING
AND ARCHITECTURE

DEPARTMENT OF TELECOMMUNICATIONS AND INFORMATION PROCESSING (TELIN)
RESEARCH GROUP IMAGE PROCESSING AND INTERPRETATION (IPI)

SIMULATING N-BODY SYSTEMS

Simon Donné, November 9th 2016

GHENT
UNIVERSITY

OVERVIEW OF THE PRESENTATION

Problems and solutions

Full-on simulation

Attraction fields

Octree approximation

Out of memory problems

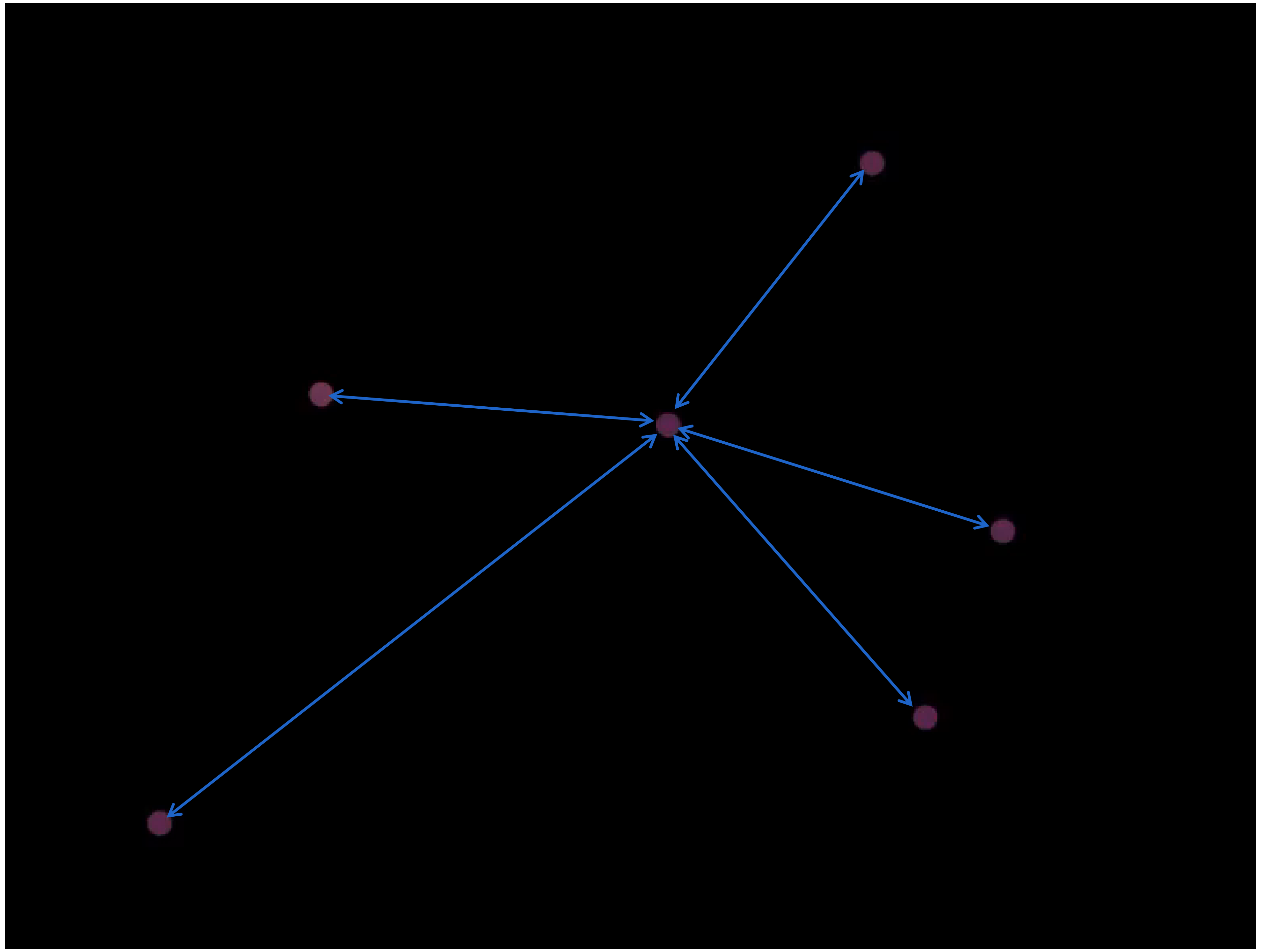
PROBLEMS AND SOLUTIONS

Every particle influences every other particle

6 particles: 15 interactions

81’920 particles: 3’355’402’240 interactions

Calculating them all is a lot of work: $O(n^2)$!



GHENT UNIVERSITY

PROBLEMS AND SOLUTIONS

All implementation is done using Quasar

An in-house developed programming language

GPU programming through automatic code generation

Automatic optimization of kernel parameters

Perfect for prototyping!

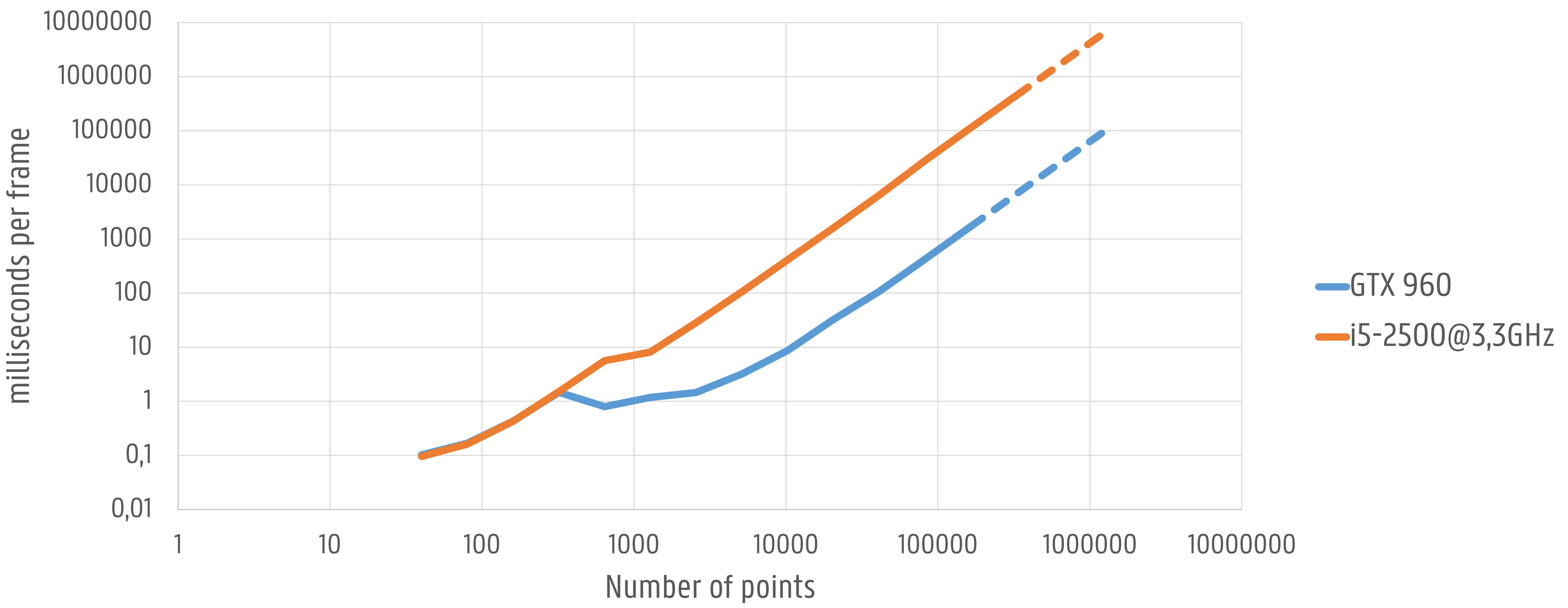
Incremental implementation times given for all methods

FULL-ON SIMULATION

Full implementation time: 3 hours

The ground-truth approach

Calculate all n(n-1) attractions



GHENT UNIVERSITY

ATTRACTION FIELDS

Each point throws an attraction field in the space around it

We approximate these attraction fields

ATTRACTION FIELDS

Incremental implementation time: 0,5 hours

Attempt 1: DxDxD voxel cube

For each voxel, calculate the effects of all points on it

Then each point simply looks up its location in the cube

Calculating the field: $O(nD^3)$ TOO SLOW

Field lookup: $O(n)$

ATTRACTION FIELDS

Incremental implementation time: 1,5 hours

Attempt 2: a hierarchy of voxel cubes (height h, subdivision D)

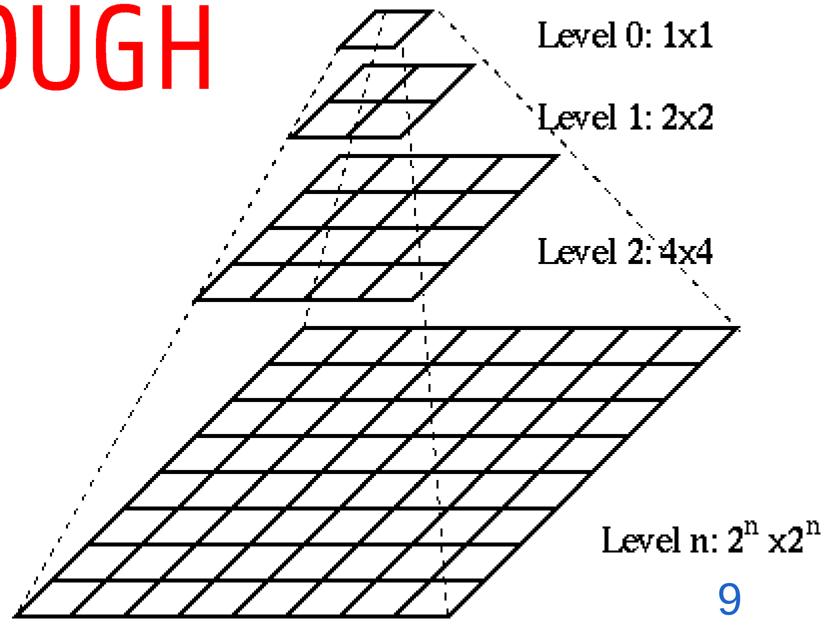
we use h=9, D=2

At each level, we calculate the exerted attraction on all voxels we are not in, and continue recursively into the voxel we *are* in.

Calculating the field: $O(h(D^3-1)n)$

Calculating the field: $O(hn)$

NOT FINE ENOUGH

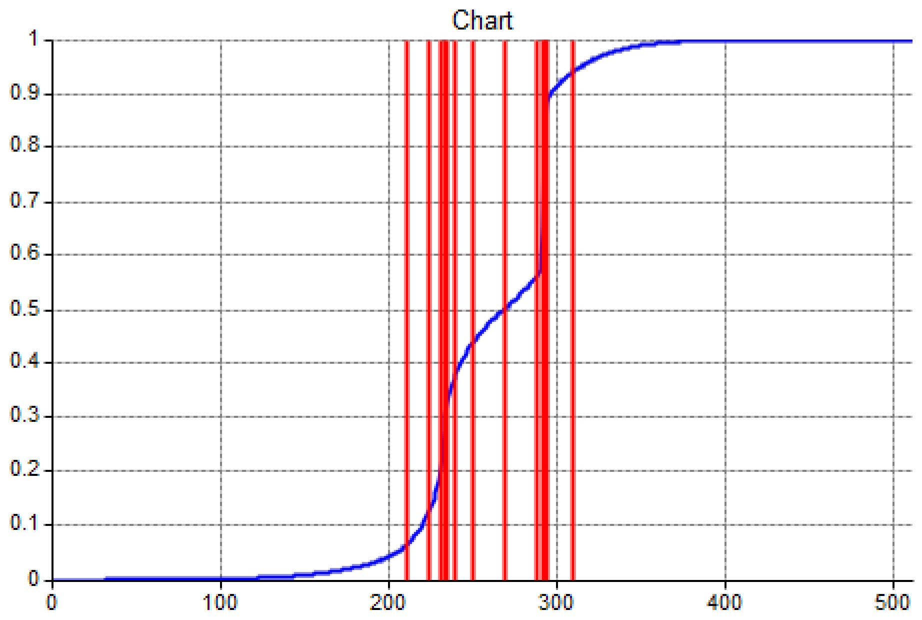


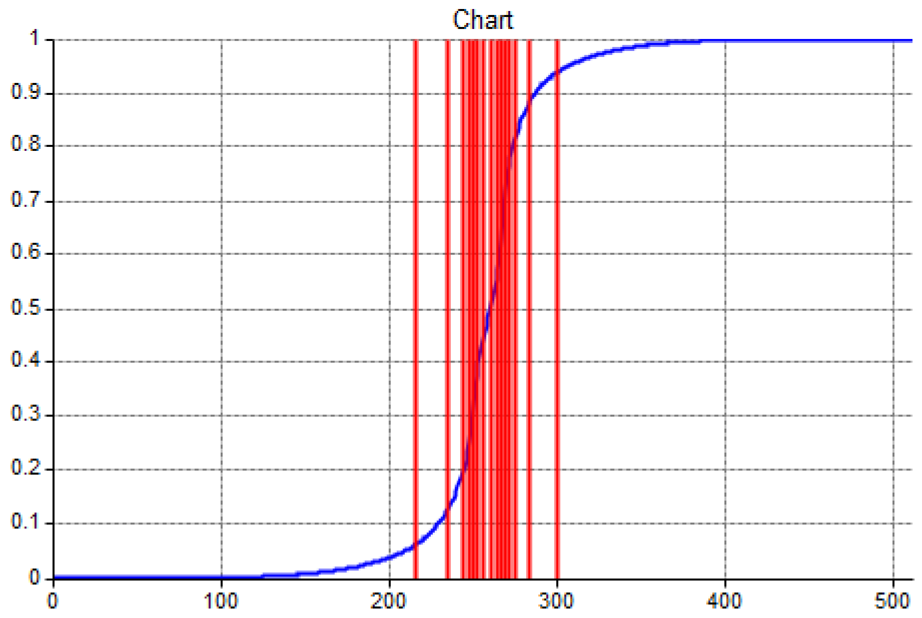
GHENT UNIVERSITY

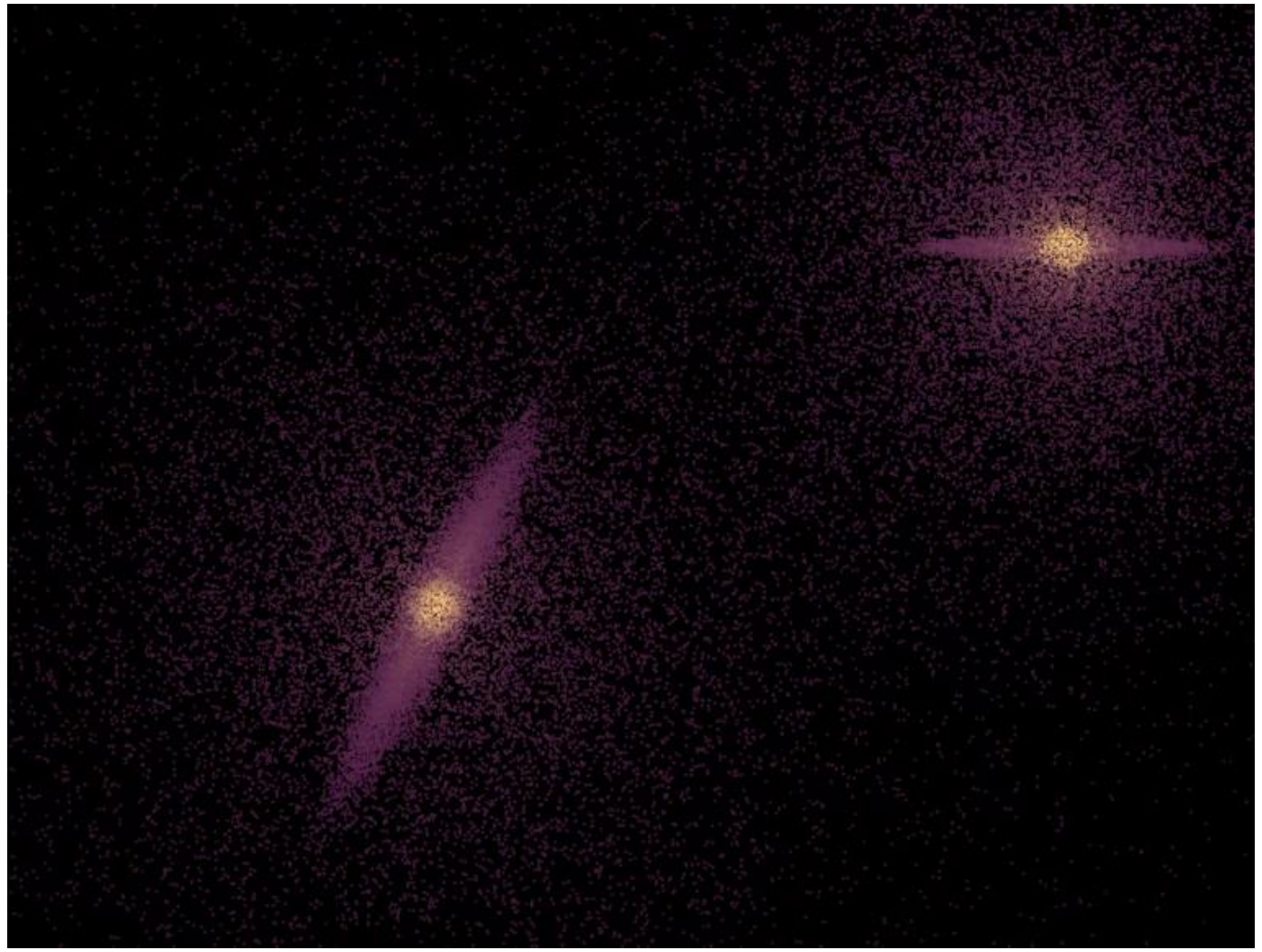
Incremental implementation time: 1,5 hours

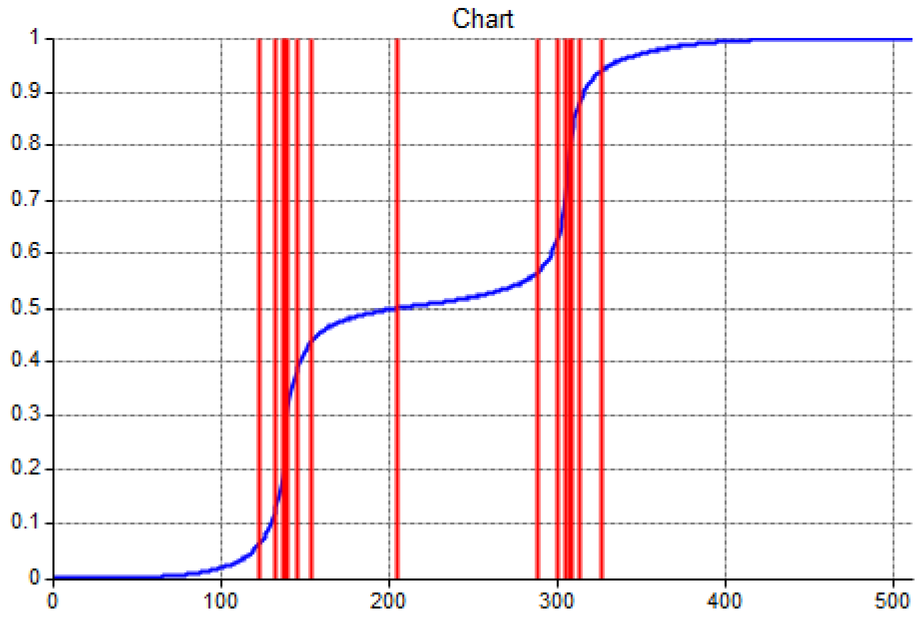
Attempt 2: a hierarchy of nonuniform voxel cubes

The voxel cubes are no longer uniform

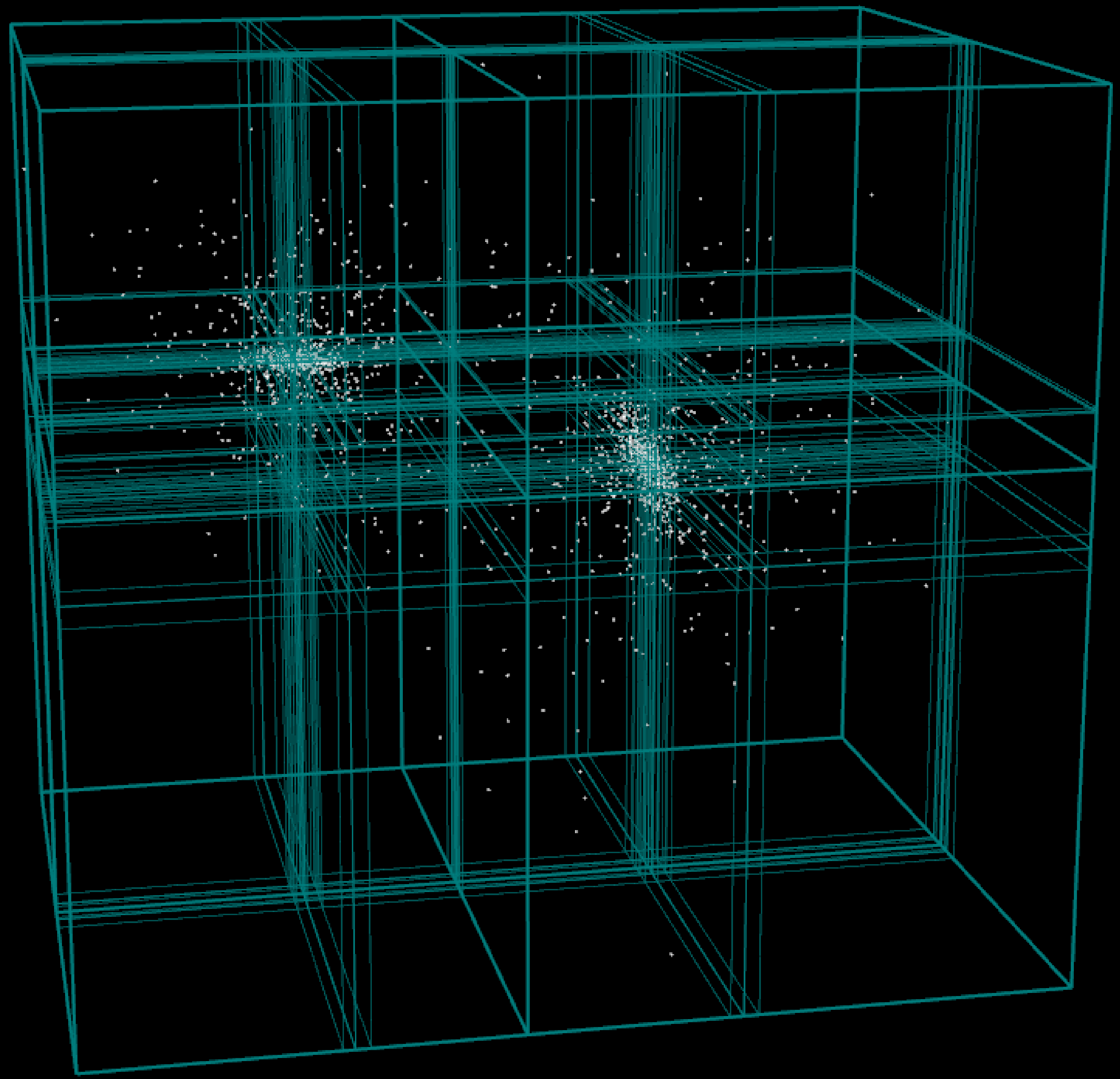








GHENT UNIVERSITY



ATTRACTION FIELDS

Incremental implementation time: 3,5 hours

Attempt 3: a hierarchy of nonuniform voxel cubes

The voxel cubes are no longer uniform

Nice properties:

Calculating the subdivisions: $O(nD^h)$

Calculating the field: $O(h(D^3-1)n)$

STILL NOT FINE ENOUGH

Calculating the field: $O(hn)$

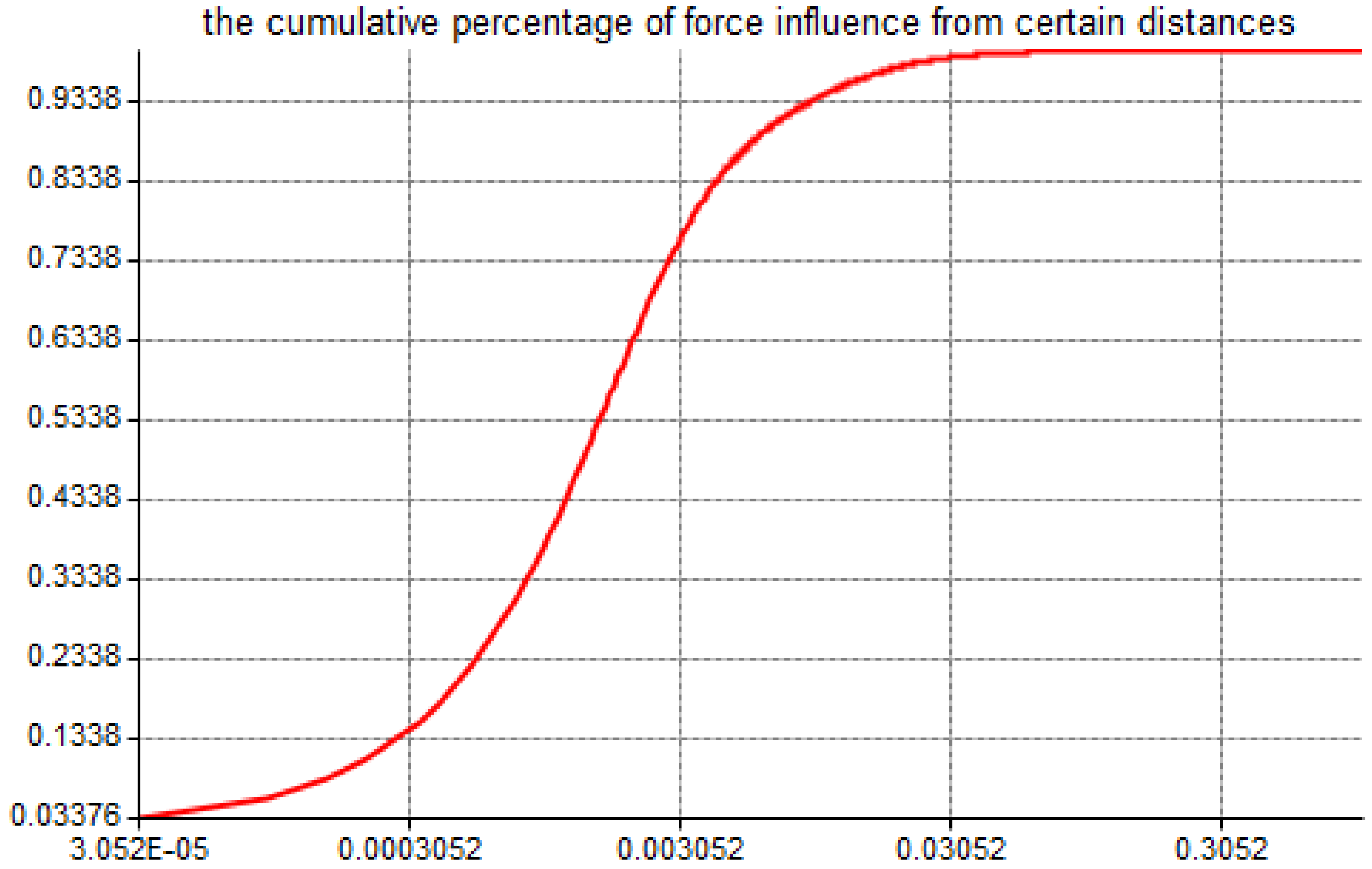
GHENT UNIVERSITY

OCTREE APPROXIMATION

The attraction field method failed

Gravitational attraction is ruled by close objects

Even though the milky way dataset has this distribution:

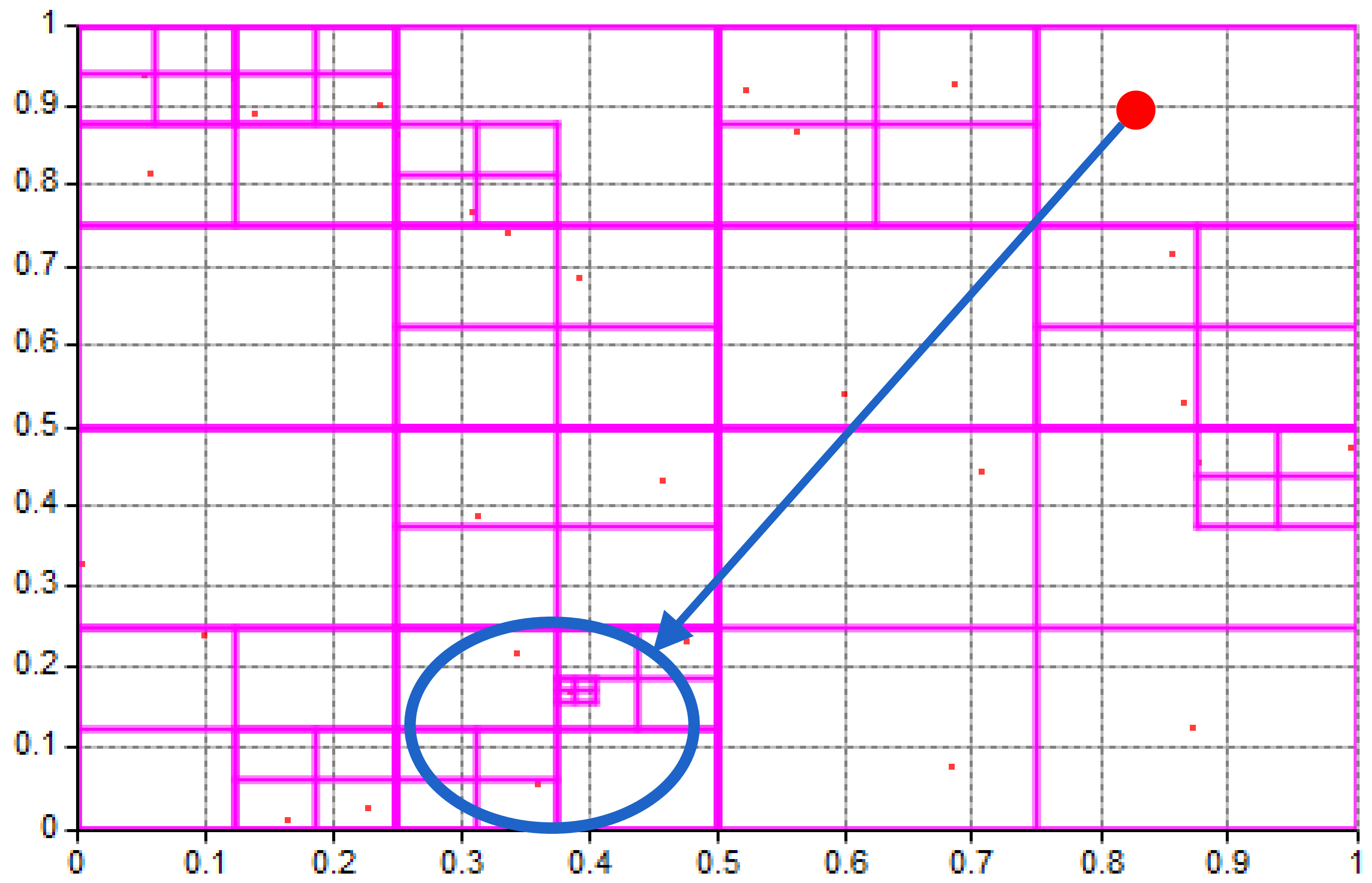


GHENT UNIVERSITY

OCTREE APPROXIMATION

Approximate large far-away clusters as a single point

Using oct-trees



GHENT UNIVERSITY

OCTREE APPROXIMATION

Octree implementation time: 6 hours

The oct-tree implementation is based on

"Parallel construction of quadtrees and quality triangulations"

1. Interleave the bits of the fixed-point float coordinates
2. Sort the list

Octant 4

GHENT UNIVERSITY

OCTREE APPROXIMATION

Octree implementation time: 6 hours

The oct-tree implementation is based on

“Parallel construction of quadtrees and quality triangulations”

3. From this list, the tree can be built in a straightforward manner

Total complexity: O(n log(n))

GHENT UNIVERSITY

OCTREE APPROXIMATION

Octree implementation time: 6 hours

4. Nice GPU optimization step

We have the points ordered spatially

i.e. points following the same path through the tree

Use these for selecting kernel blocks

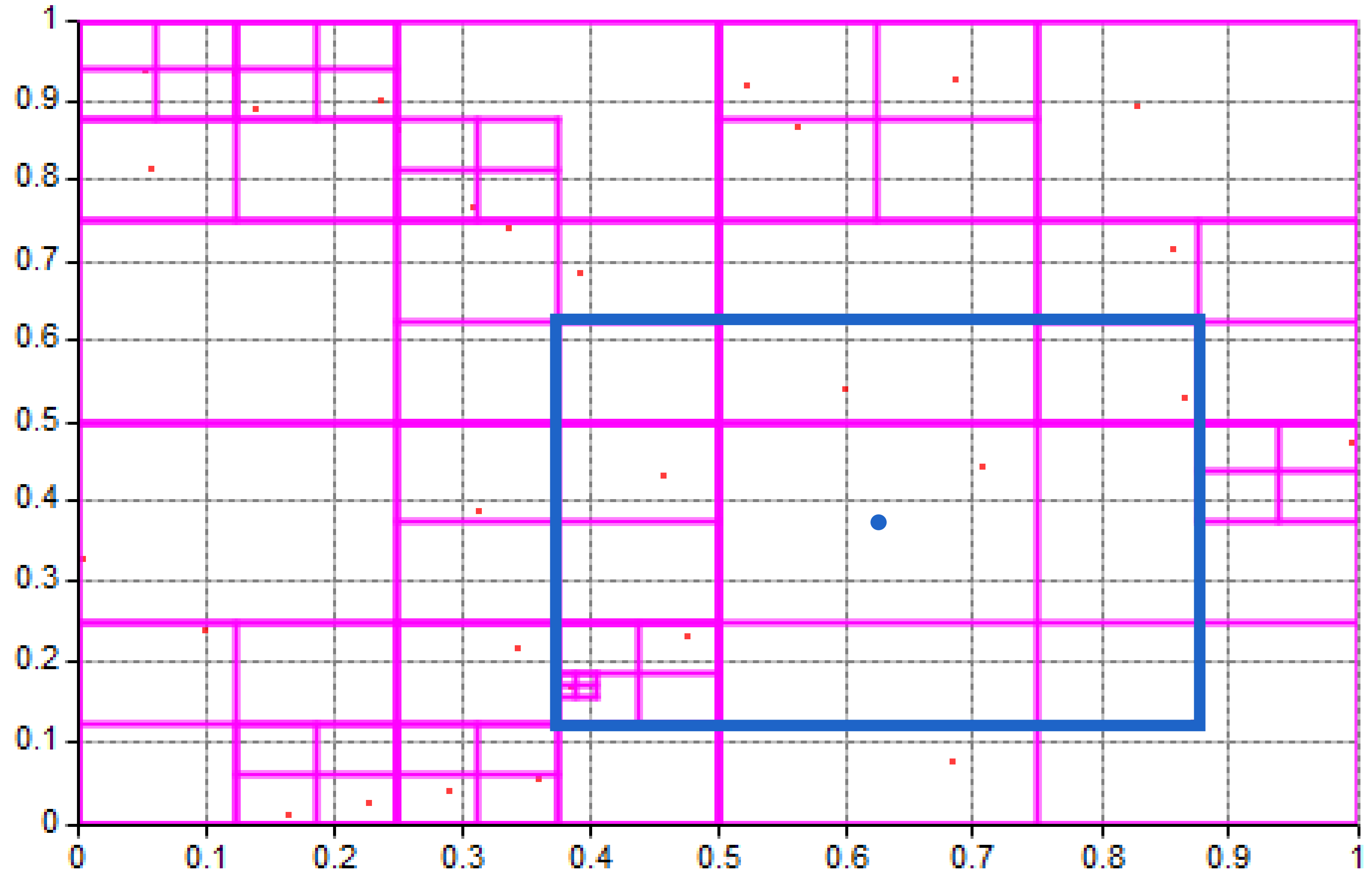
⇔ arbitrary ordering from original file read

OCTREE APPROXIMATION

Incremental implementation time: 5 hours

Accuracy: when do we approximate tree nodes as a single cluster?

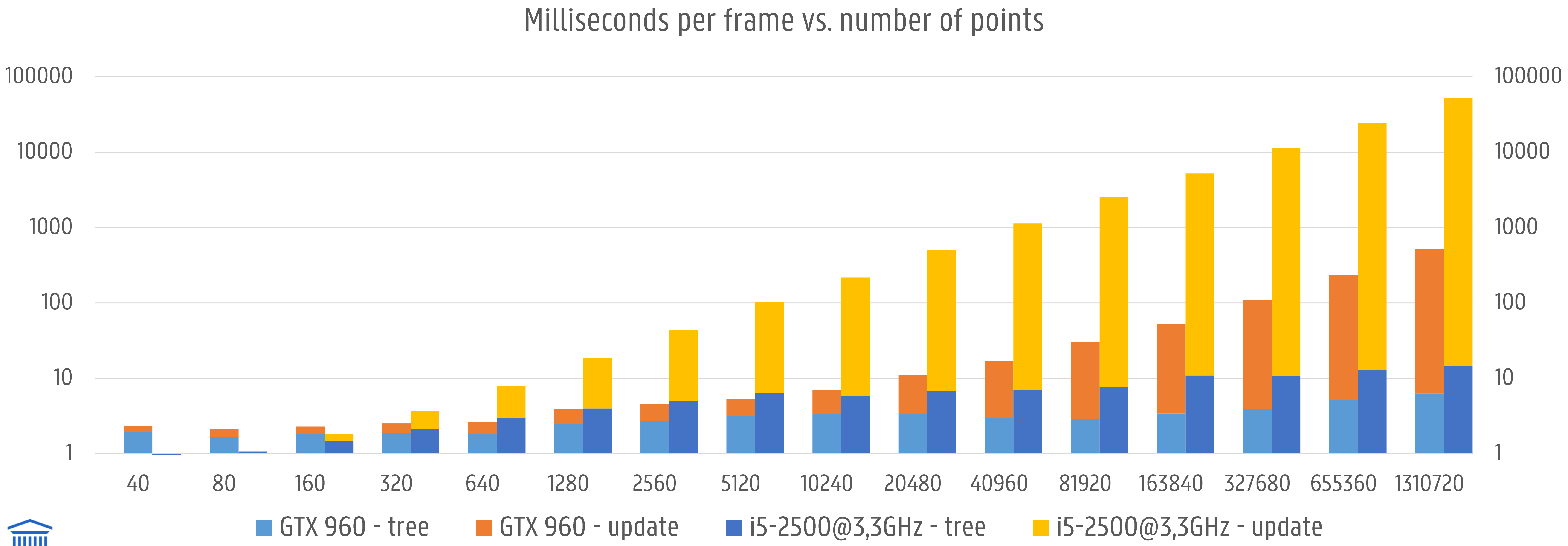
Heuristic rule: point is not in the twice-as-wide cube



GHENT UNIVERSITY

OCTREE APPROXIMATION

The resulting approach is translated well to GPU

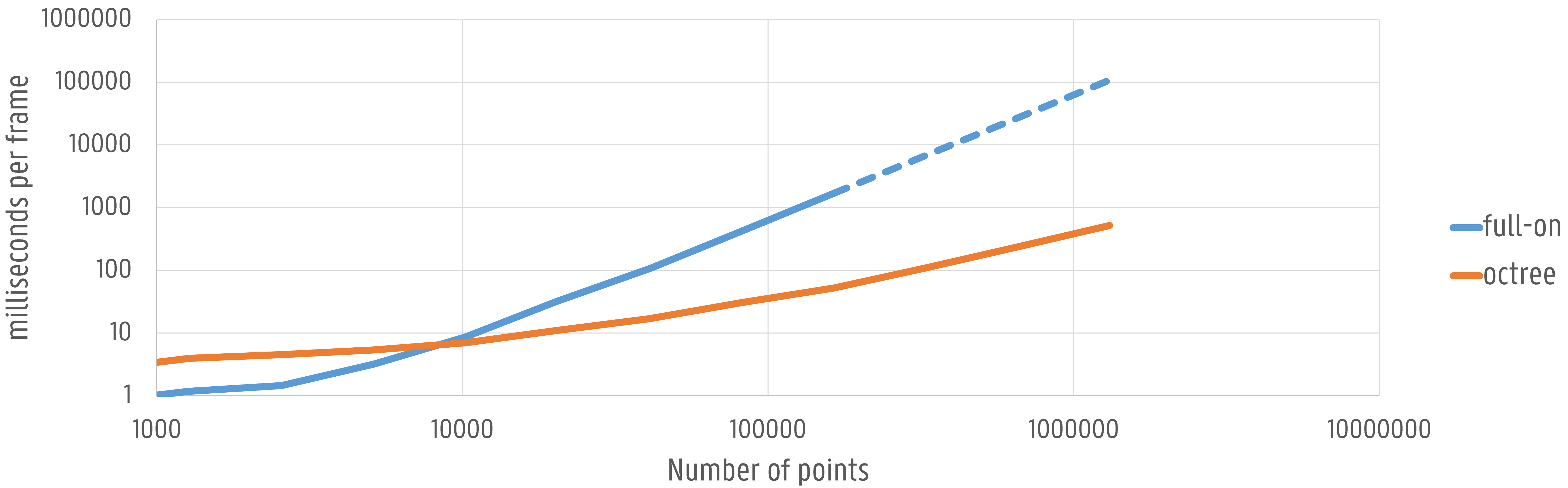


GHENT UNIVERSITY

Fractions represent the actual fraction of the runtime

OCTREE APPROXIMATION

Much faster than full-on calculations!



GHENT UNIVERSITY

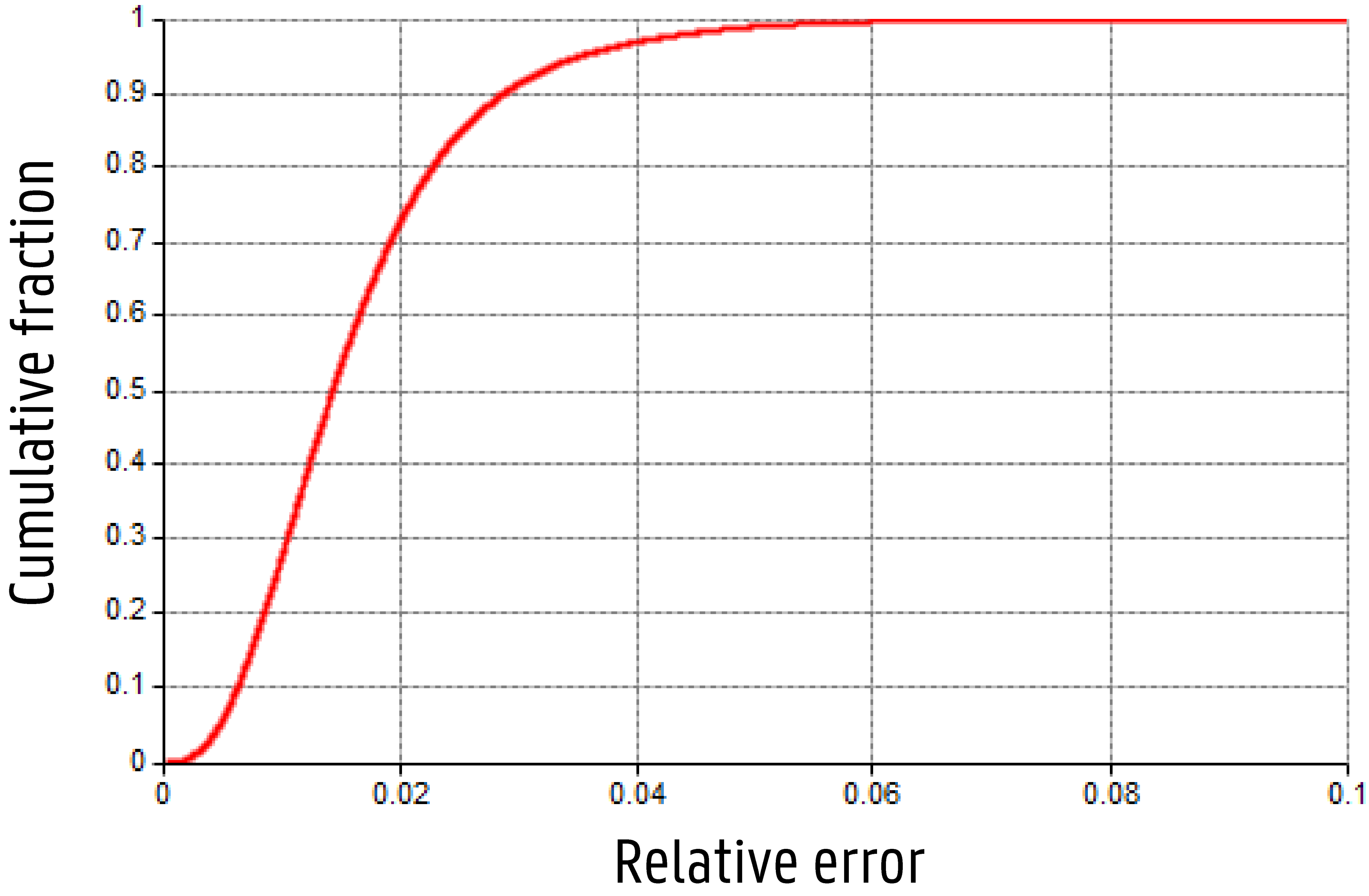
OCTREE APPROXIMATION

The resulting simulation is close to the full-on simulation

For the Milky Way/Andromeda cluster collision:

mean error ~1,5%

stddev error ~1%



GHENT UNIVERSITY

OUT OF MEMORY PROBLEMS

Two possible problems:

- Memory constraint
- CUDA kernel runtime (max 5s)

SPLIT UP DATASETS INTO SEGMENTS

Or, a positive reason:

- Multiple GPUs available

OUT OF MEMORY PROBLEMS

Forces are additive, so:

1) Make a separate tree for each segment of points

   one kernel per segment

2) Calculate the attractions of each tree on each segment

   one kernel per segment per tree