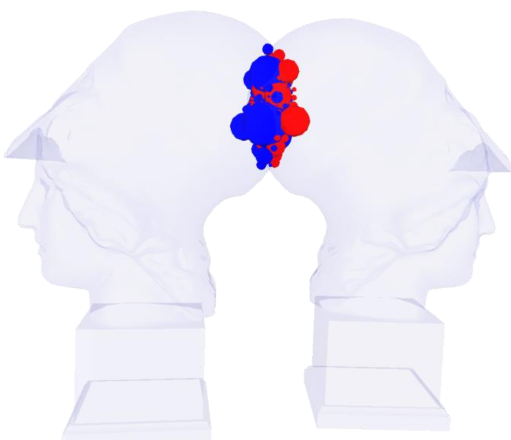
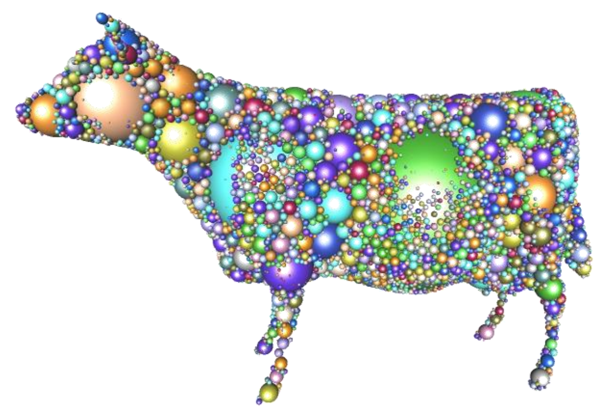
Volumetric Collision Detection and Response for Haptic Rendering

René Weller

University of Bremen
cgvr.cs.uni-bremen.de



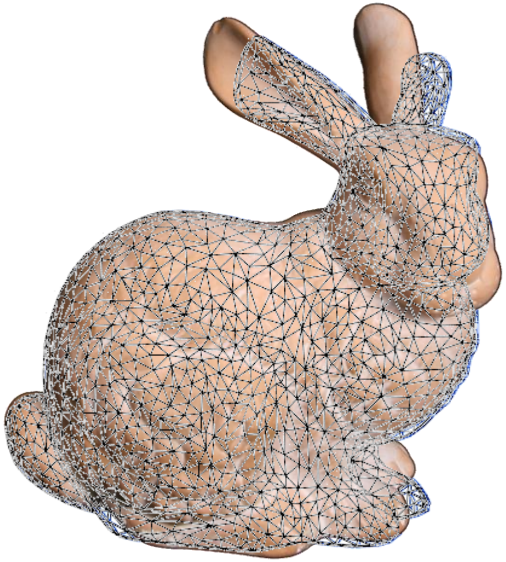


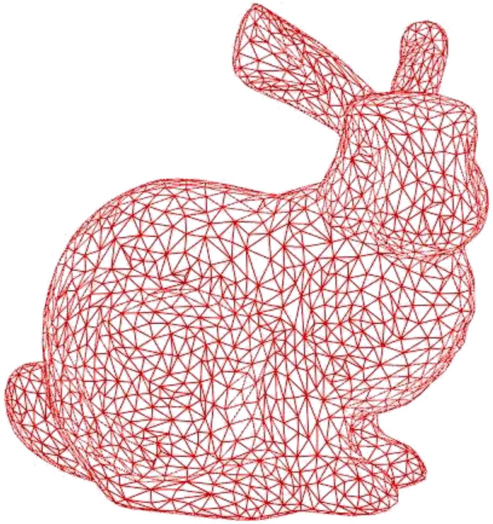
Workshop on Haptic Methods and Technologies for Virtual Assembly Simulations

IEEE World Haptics, June 2017

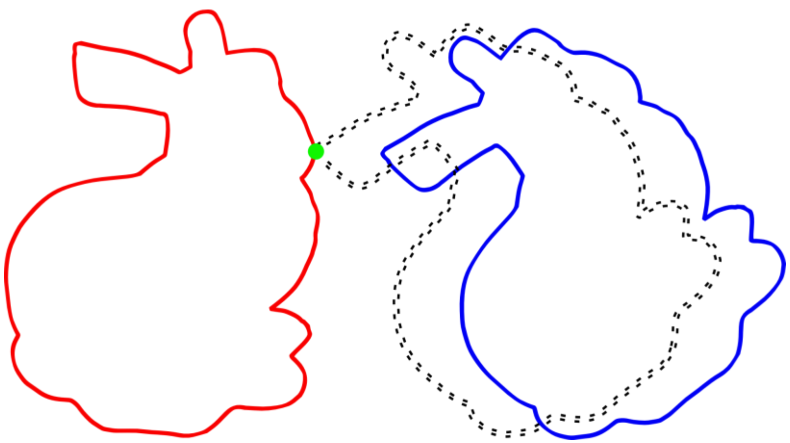
Motivation

Motivation: Collision Detection

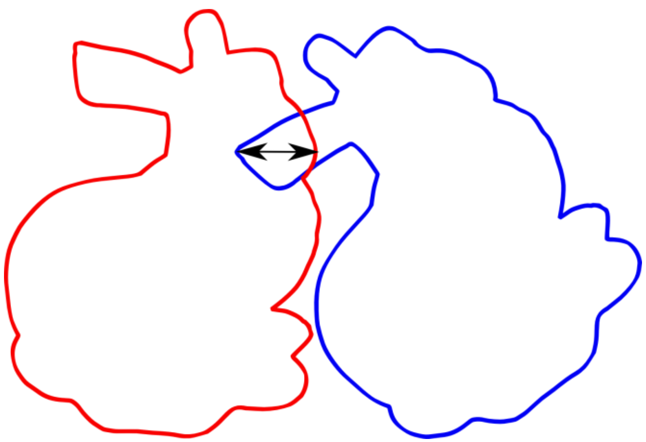




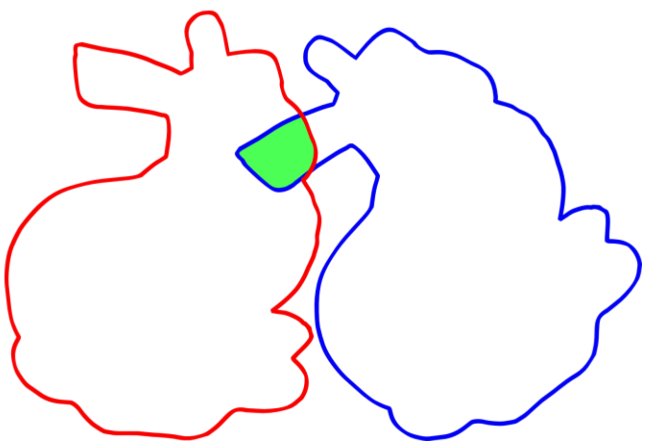
Penetration Measures



Continuous collision detection

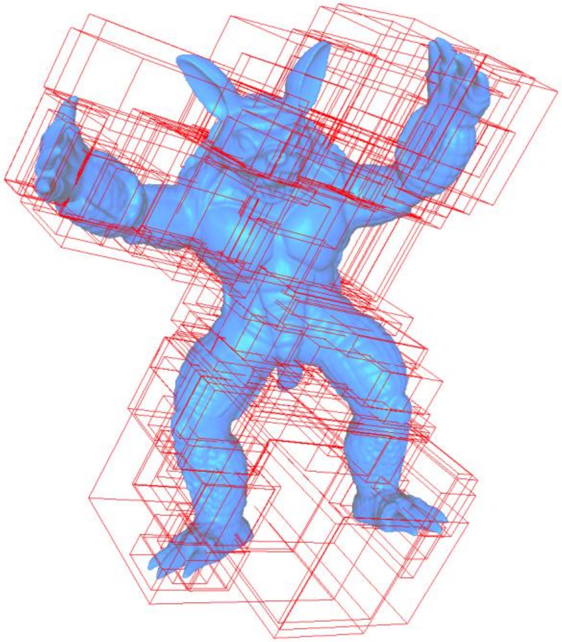


Translational penetration depth



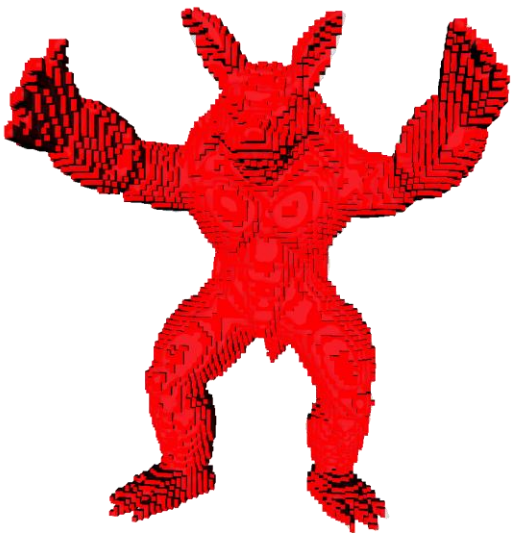
Penetration volume - “the most complicated yet accurate method” [Fisher and Lin, 2001]

BVHs vs Voxels



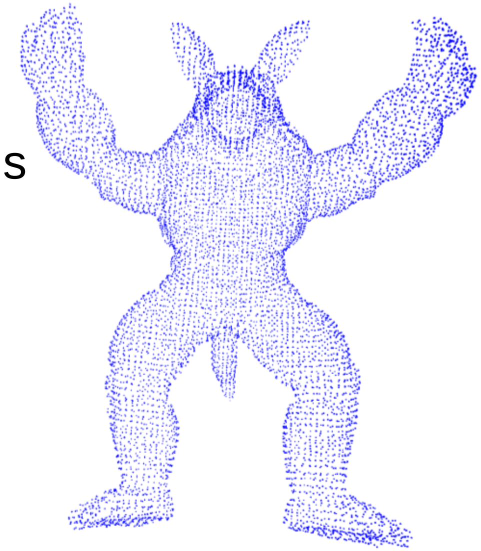
- BVHs
  - Easy to build
  - Fast, robust and exact
  - Complicated to compute penetration depth
  - Not fast enough for haptic applications

[Mendoza et al, 2006], [Zhang et al, 2007], …



- Voxel based algorithms
  - Fast enough for haptic interactions
  - Independent of object complexity
  - Memory consuming
  - Aliasing artifacts

[McNeely et al., 1999]

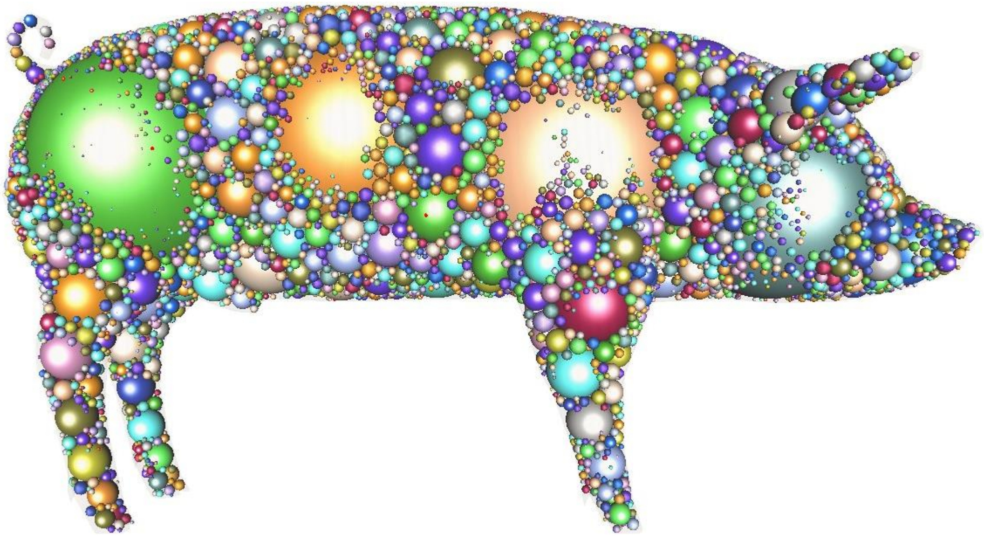


Goal: Keep the Best of Both Worlds

- Keep a single consistent data structures for moving and fixed objects
- Near constant running time and independency of objects complexity
- Stable model to compute continuous feedback forces
  - Keep the high accuracy of BVH algorithms
  - Avoid aliasing

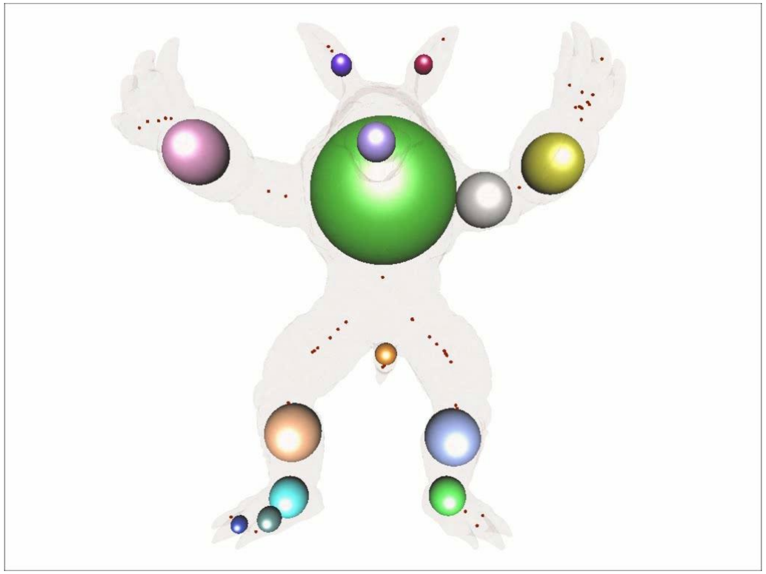
Inner Sphere Trees: Basic Idea

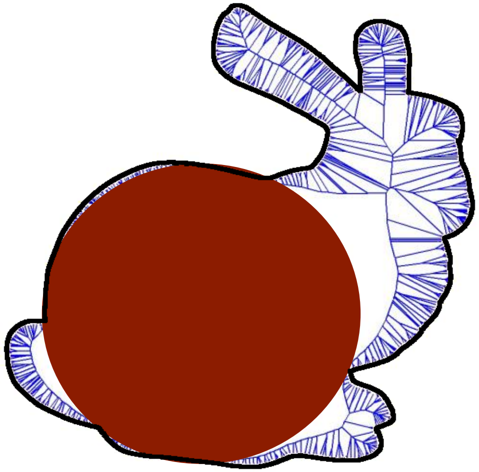
- Fill the object
  - from the *inside*
  - with *non-overlapping* spheres
- Build sphere hierarchy on *inner* spheres



Generating Spheres

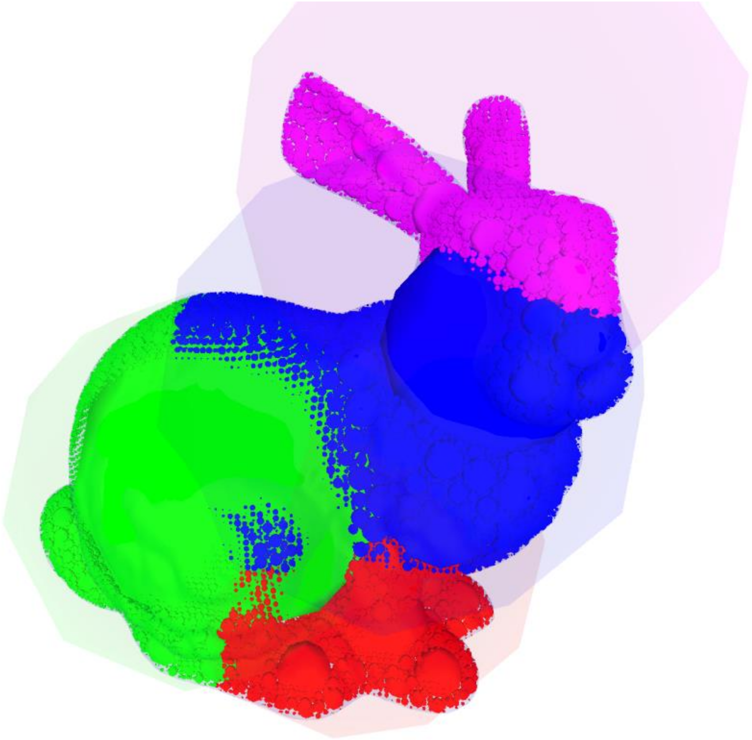
- Requirements:
  - Space-filling sphere packing
    => polydispersity
  - Support of arbitrary objects
- Protosphere
  - Basic idea:
    - Prototype-based approximation of Voronoi diagram
    - Greedy insertion of spheres
  - Massively-parallel implementation on the GPU

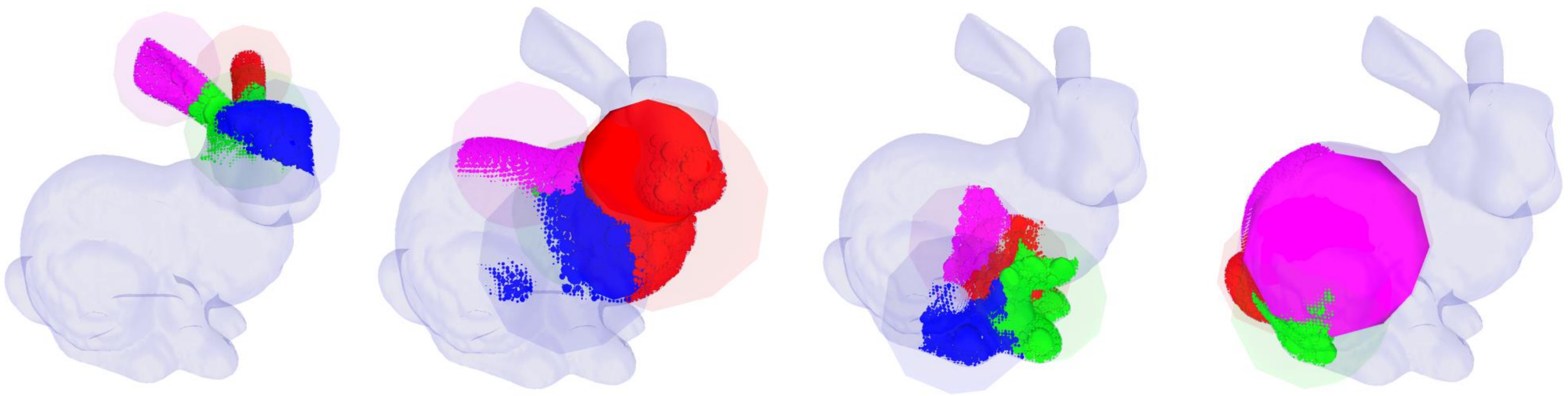




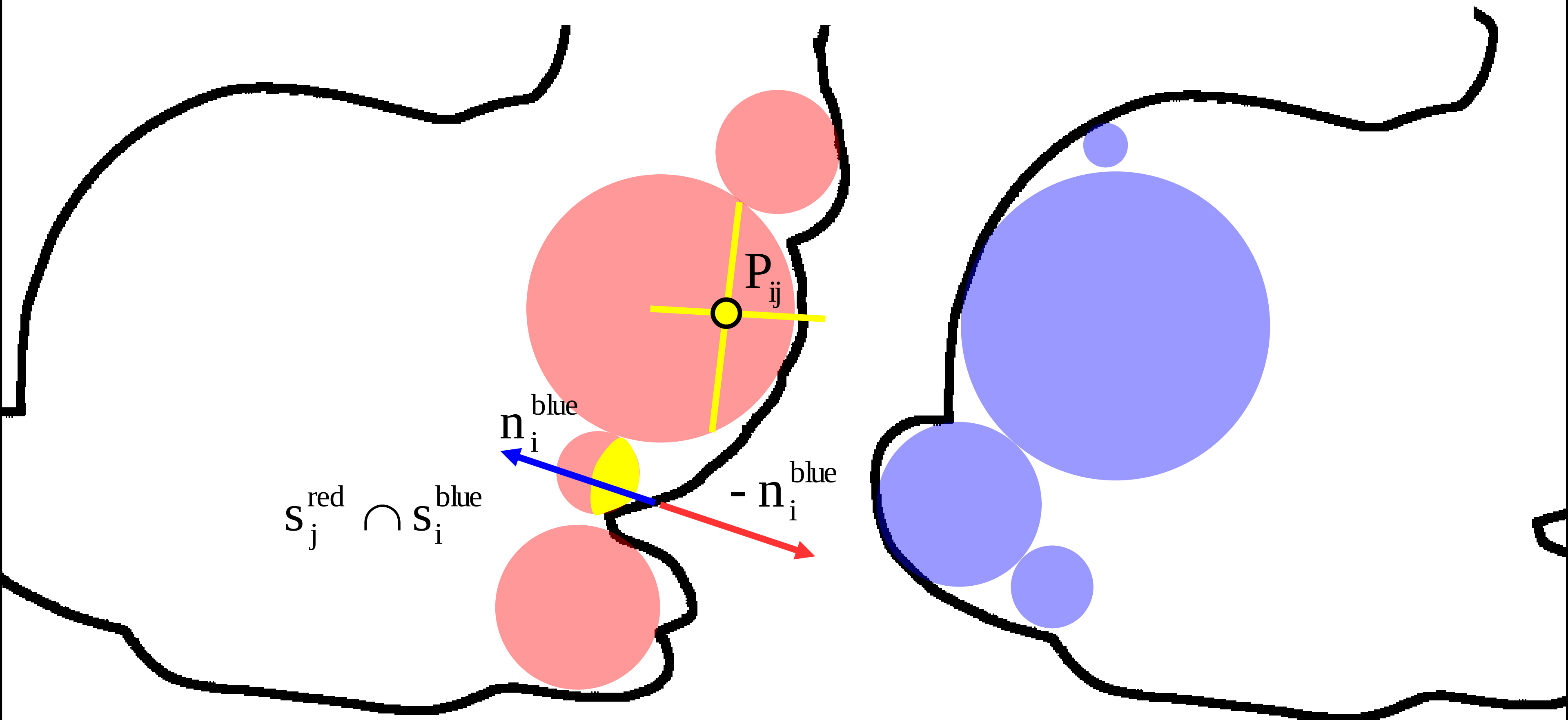
Bounding Volume Hierarchy for Inner Spheres

- Construction: massively parallel
  Batch Neural Gas on the GPU
- SIMD-accelerated traversal





Collision Response: Forces and Torques



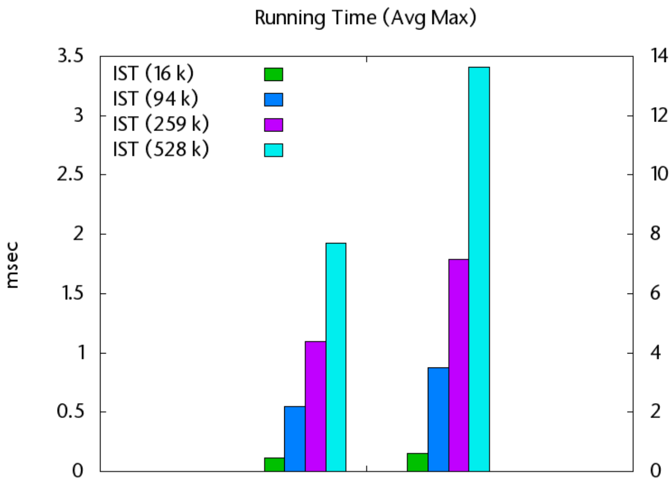
$$f_{ij}^{blue} = (s_j^{red} \cap s_i^{blue})(-n_i^{blue})$$

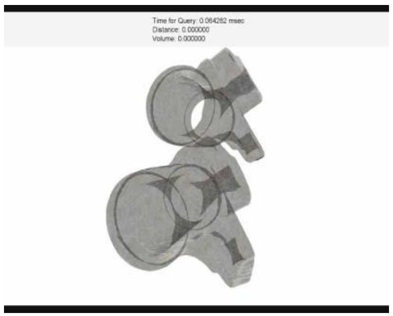
$$\tau_{ij}^{blue} = (P_{ij} - C_m) \times (f_{ij}^{blue})$$

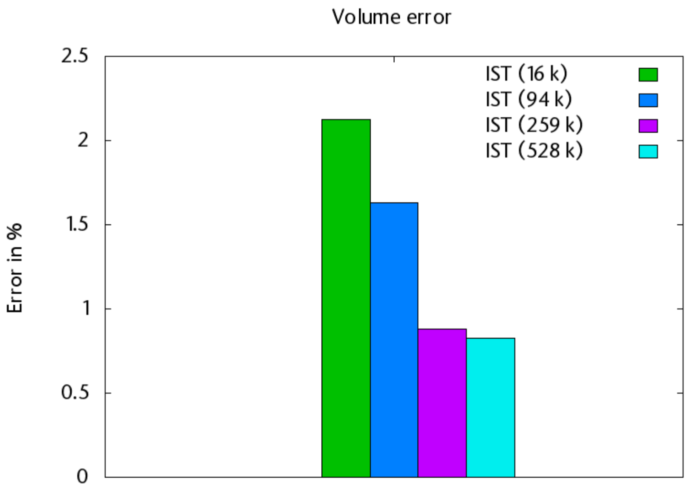
$$f_{total}^{blue} = \sum f_{ij}^{blue}$$

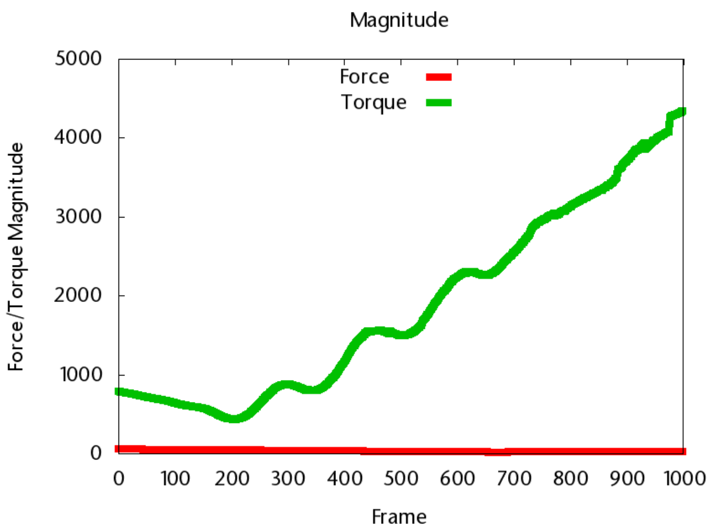
$$\tau_{total}^{blue} = \sum \tau_{ij}^{blue}$$

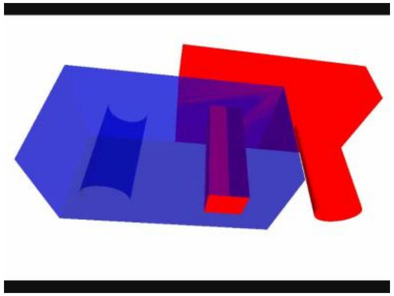
Results

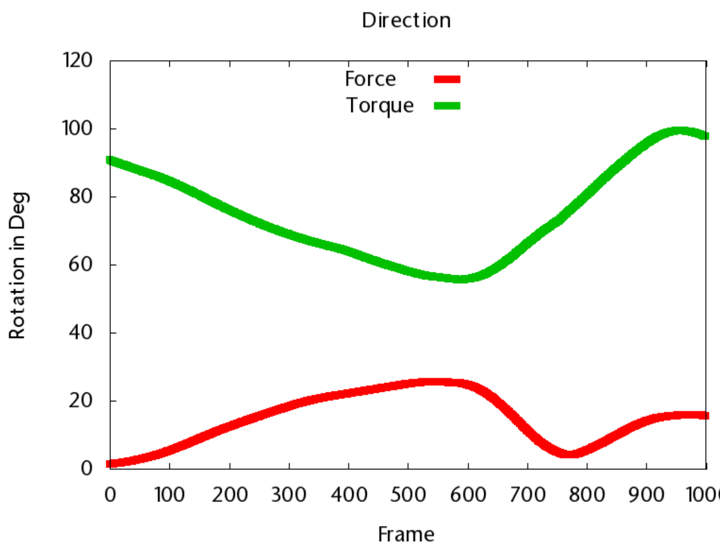

Running Time (Avg Max)
msec
IST (16 k)
IST (94 k)
IST (259 k)
IST (528 k)
0
0.5
1
1.5
2
2.5
3
3.5
0
2
4
6
8
10
12
14




Volume error
Error in %
IST (16 k)
IST (94 k)
IST (259 k)
IST (528 k)
0
0.5
1
1.5
2
2.5

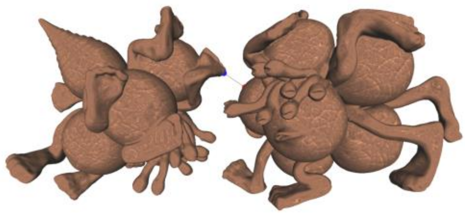

Magnitude
Force/Torque Magnitude
Force
Torque
0
1000
2000
3000
4000
5000
0
100
200
300
400
500
600
700
800
900
1000
Frame

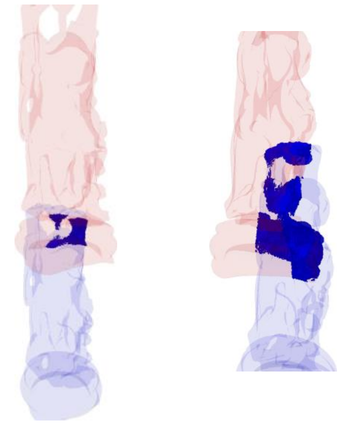



Direction
Rotation in Deg
Force
Torque
0
20
40
60
80
100
120
0
100
200
300
400
500
600
700
800
900
1000
Frame

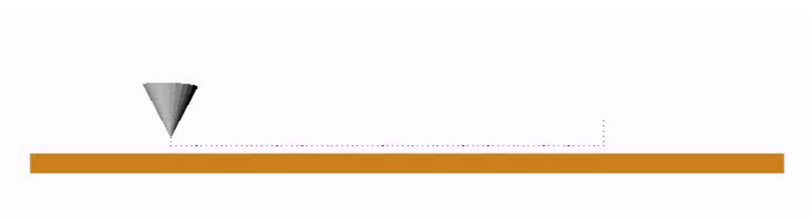
ISTs vs VPS

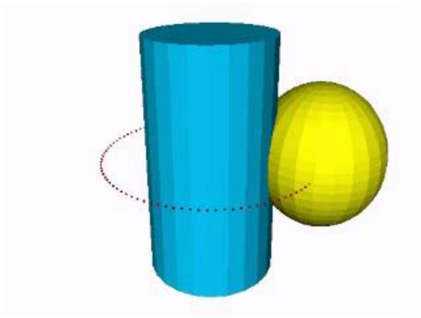
- Performance and quality benchmarks

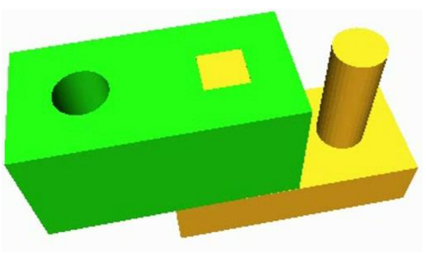




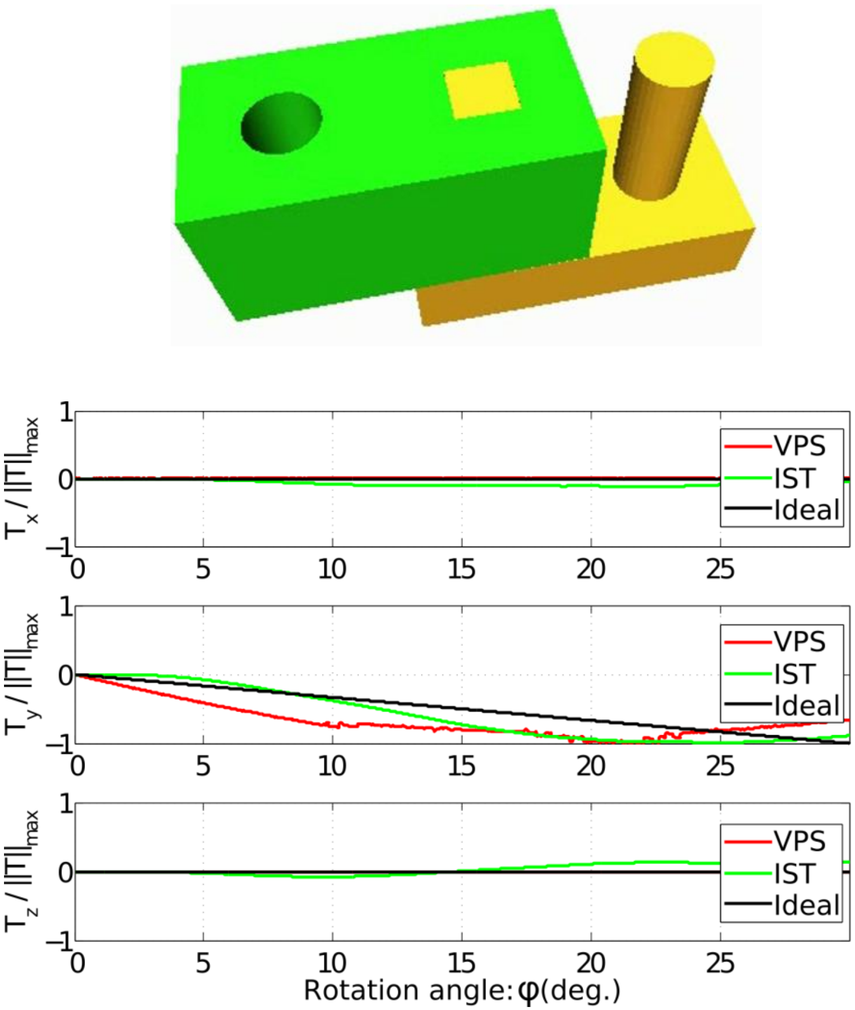
- Performance benchmark:
  - Cover large variety of different objects
  - and interesting contact scenarios
- Requirements quality benchmark:
  - Ground truth via analytical model
  - Typical contact configurations in force feedback or physically-based simulations

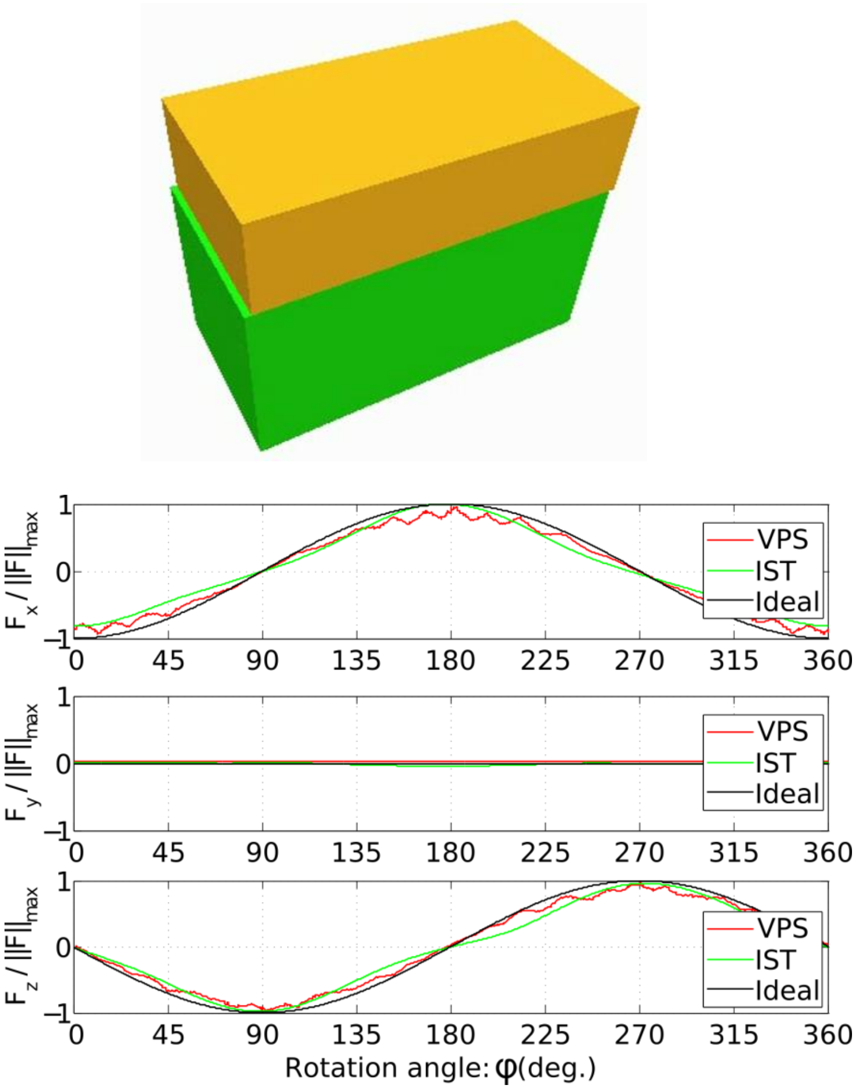




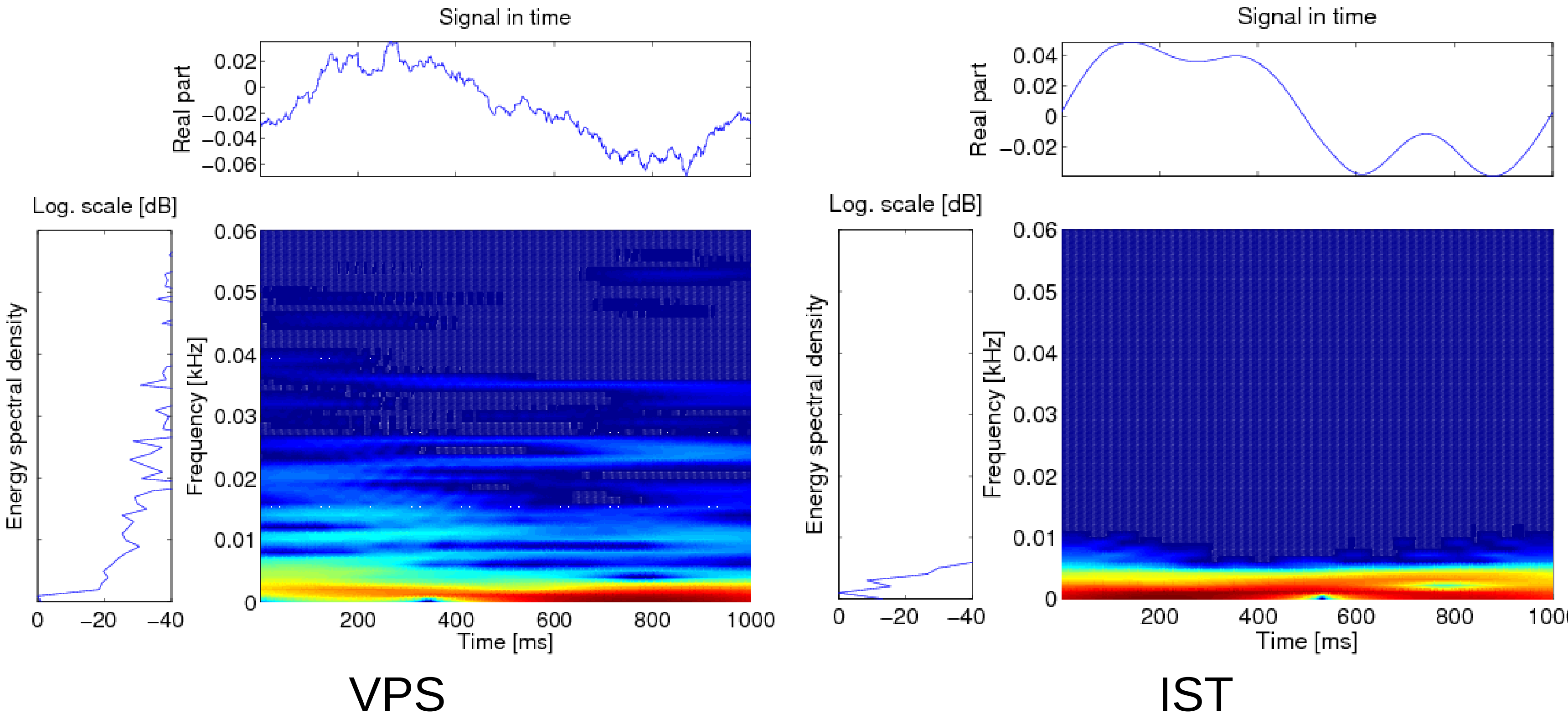


Results: Quality Benchmark



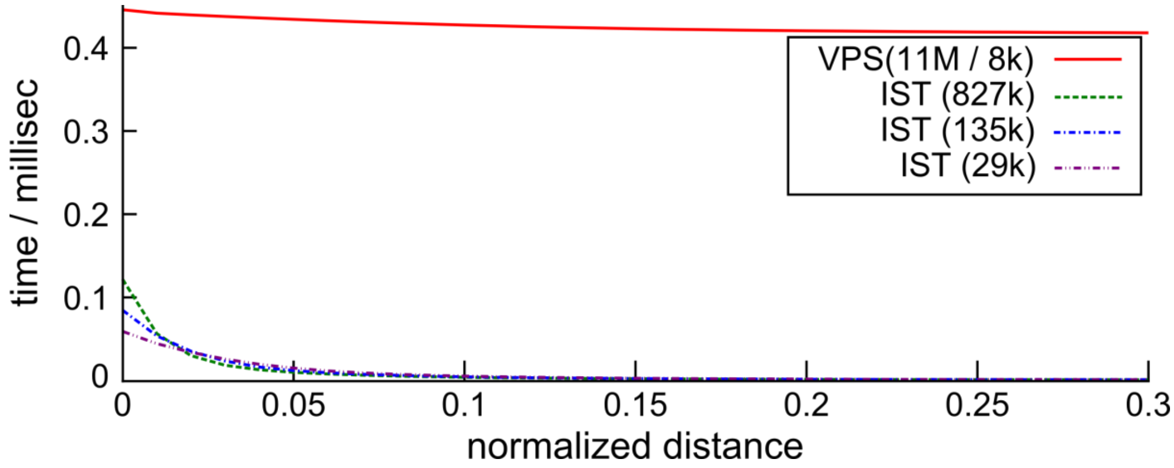


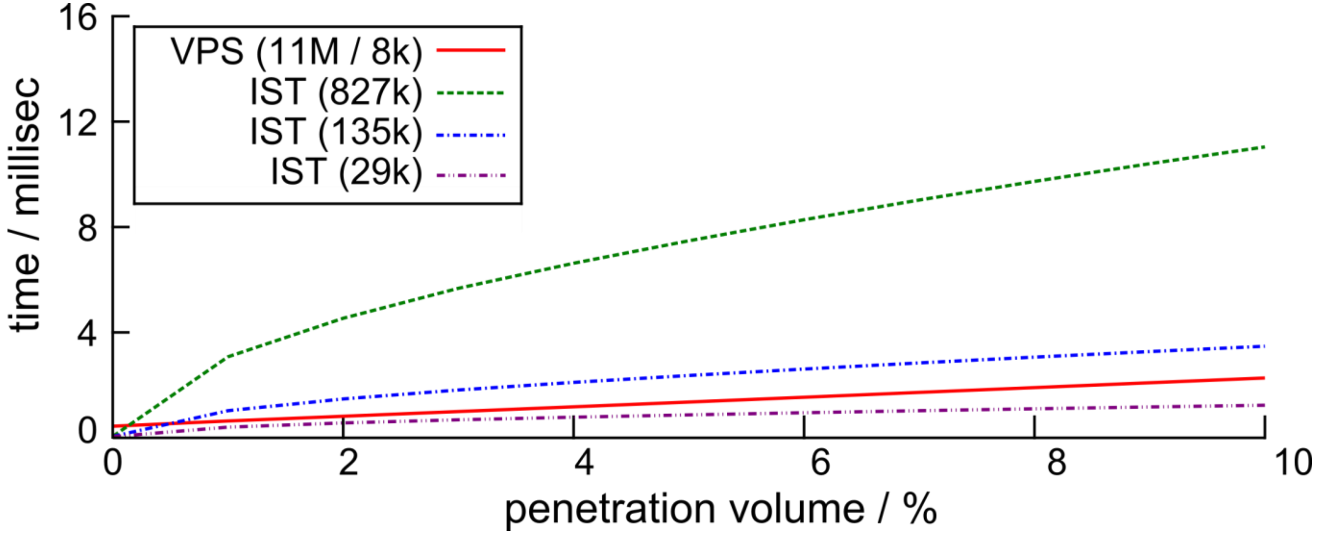
Results: Quality Benchmark



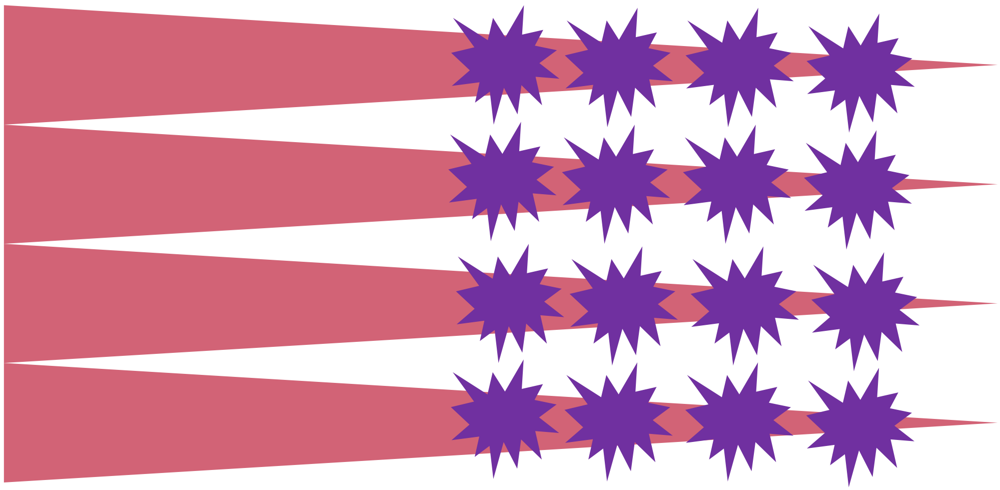
- Color coded intensity of frequency
  (dark blue represents intensity of zero)

Results: Performance Benchmark

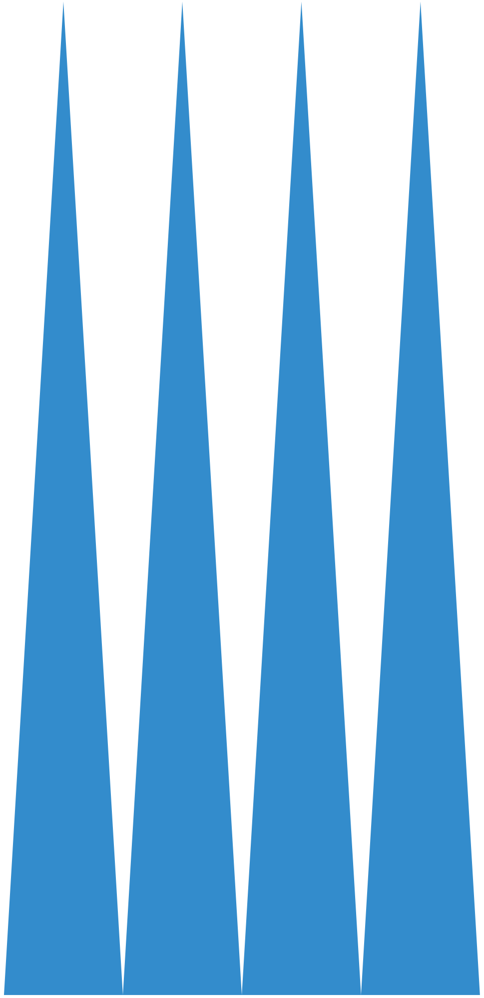




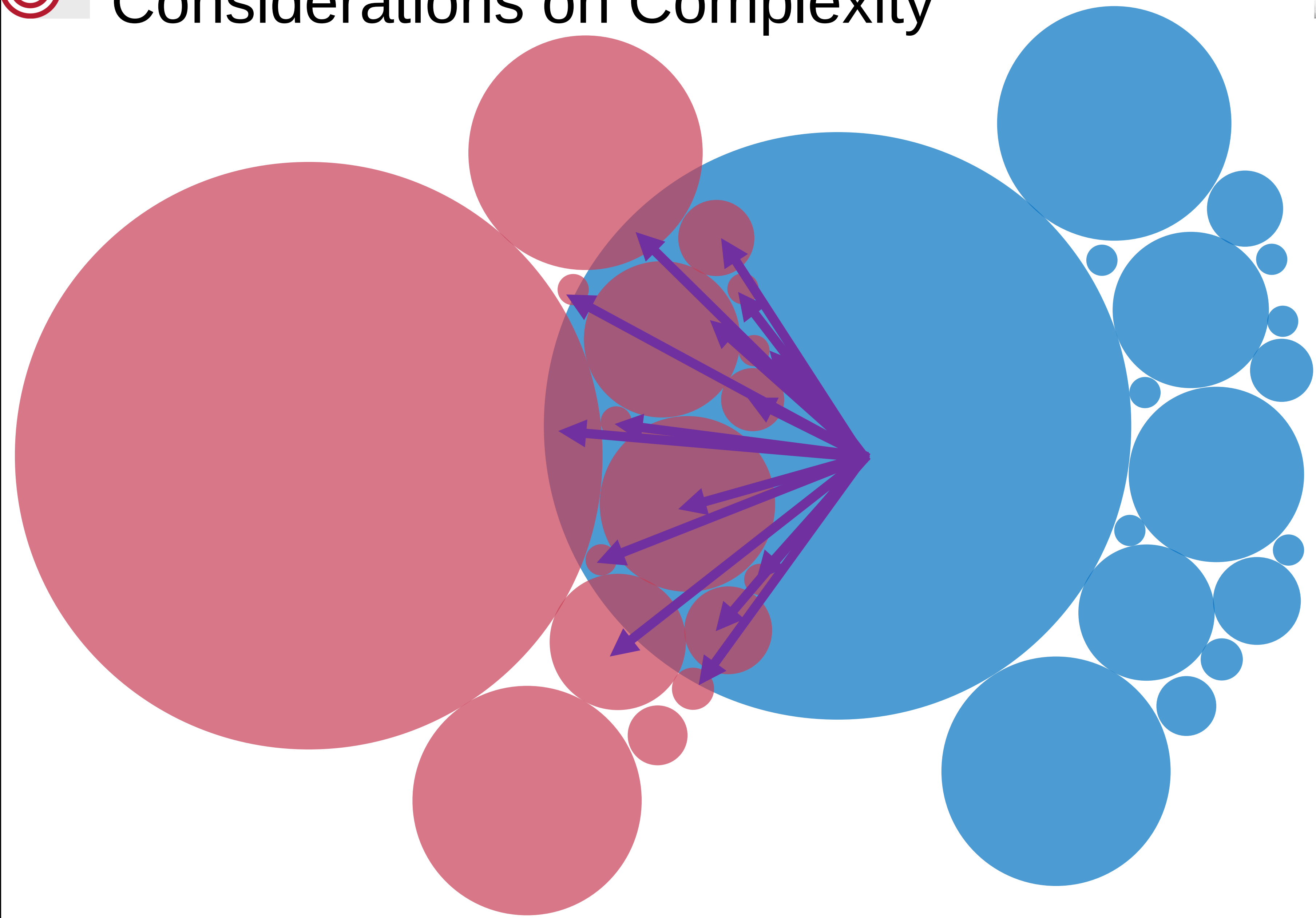
Considerations on Complexity



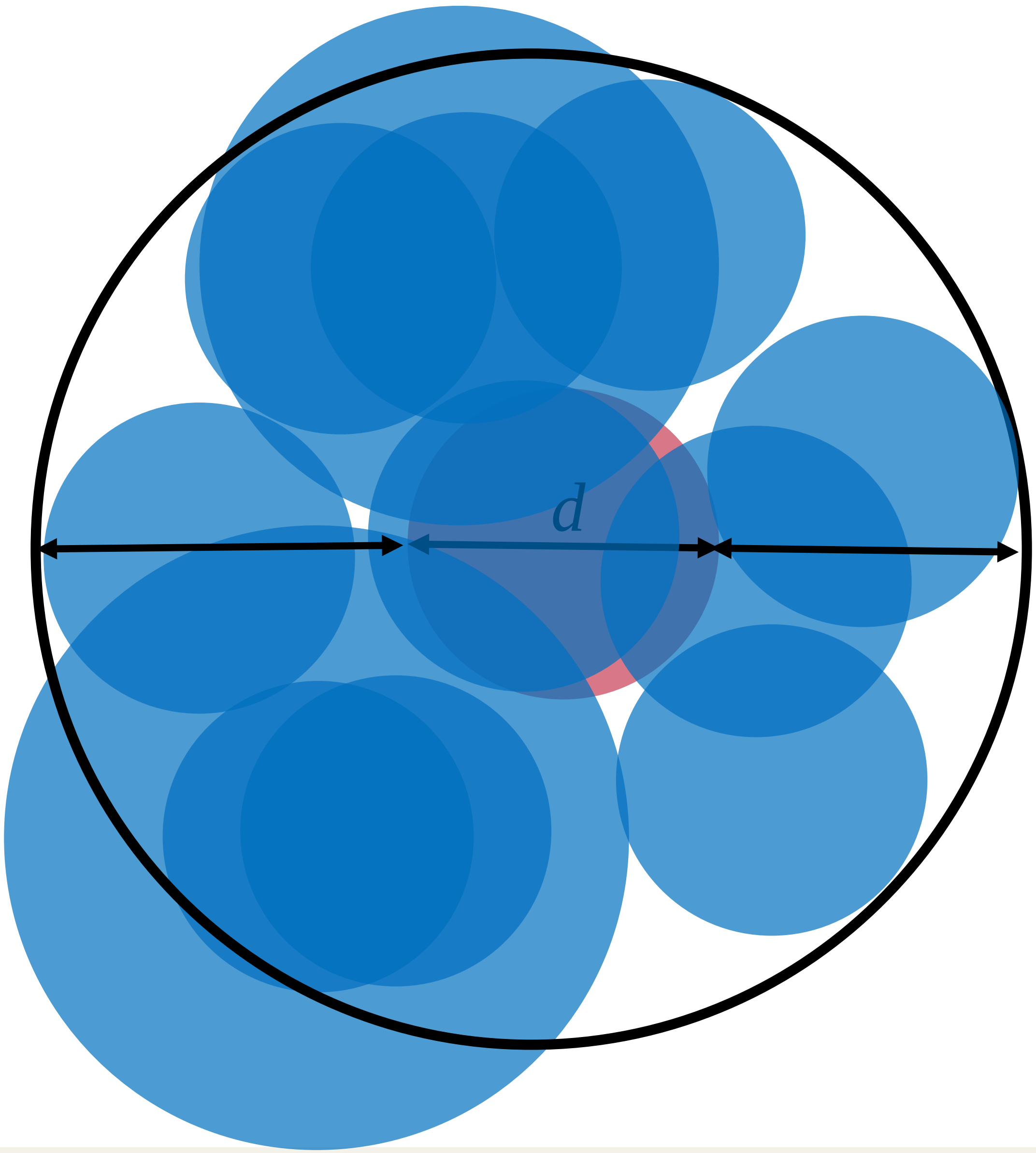
$$O(n^2)$$



Considerations on Complexity



Considerations on Complexity

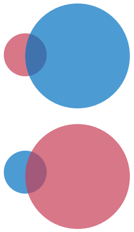


Theoretical Fundament

- Lemma: A single sphere s can intersect only a constant number of disjoint spheres A that have at least the same radius.
- Theorem: The maximum number of intersecting pairs of spheres of two polydisperse sphere packings A and B with n spheres is in $O(n)$.
  - Proof:

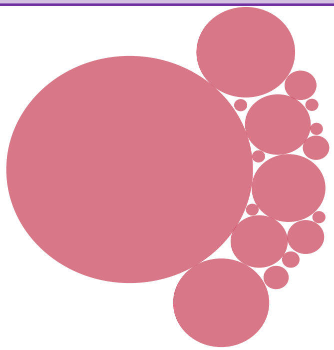
Check each sphere $s_i \in A$ against larger spheres in B

Check each sphere $s_j \in B$ against larger spheres in A

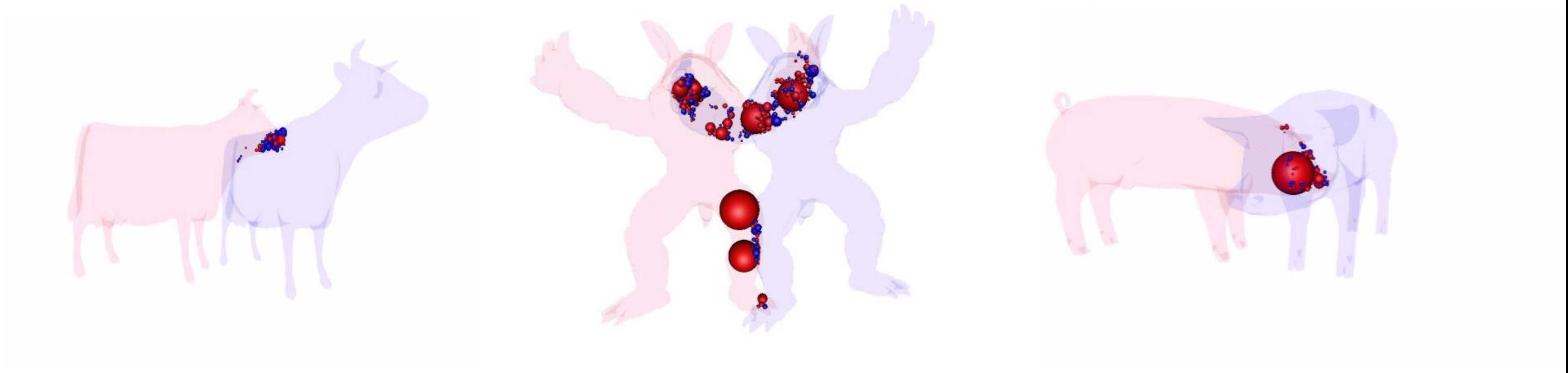


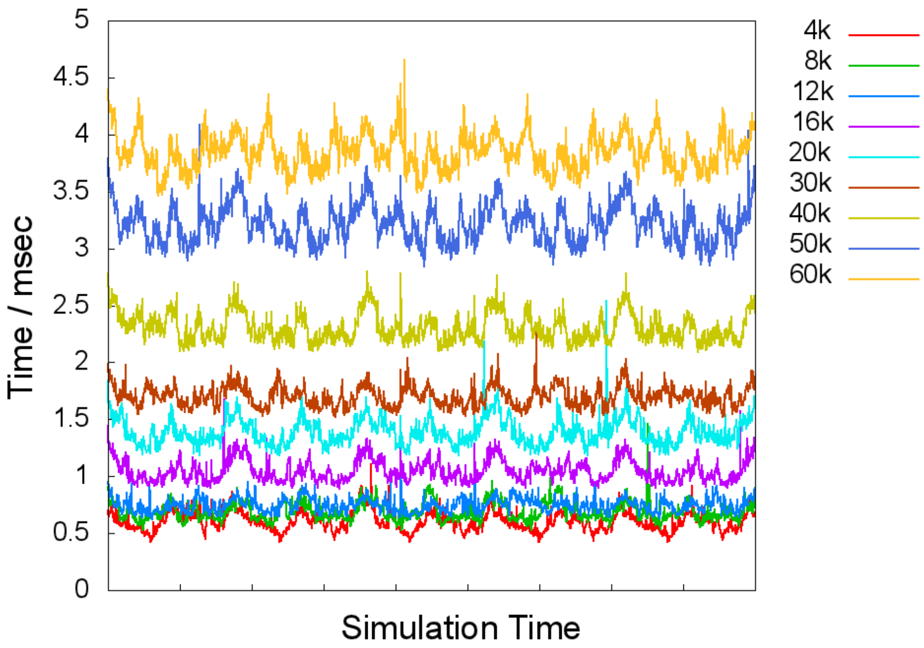
$\Rightarrow O(1)$ for each $s_i \in A$ and $s_j \in B$

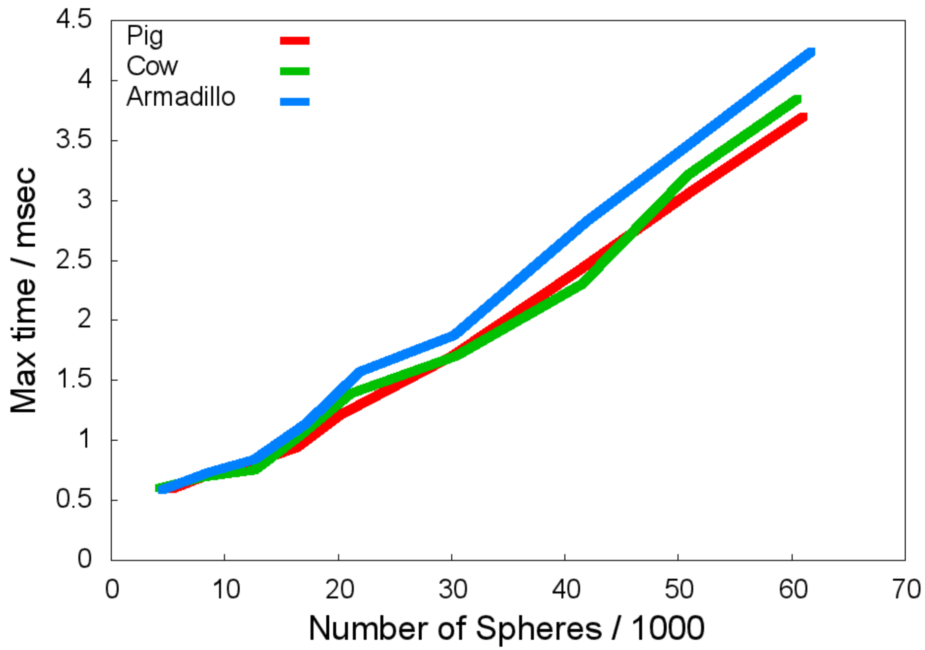
$\Rightarrow O(n)$ in total



Results




Time / msec
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Simulation Time
4k
8k
12k
16k
20k
30k
40k
50k
60k

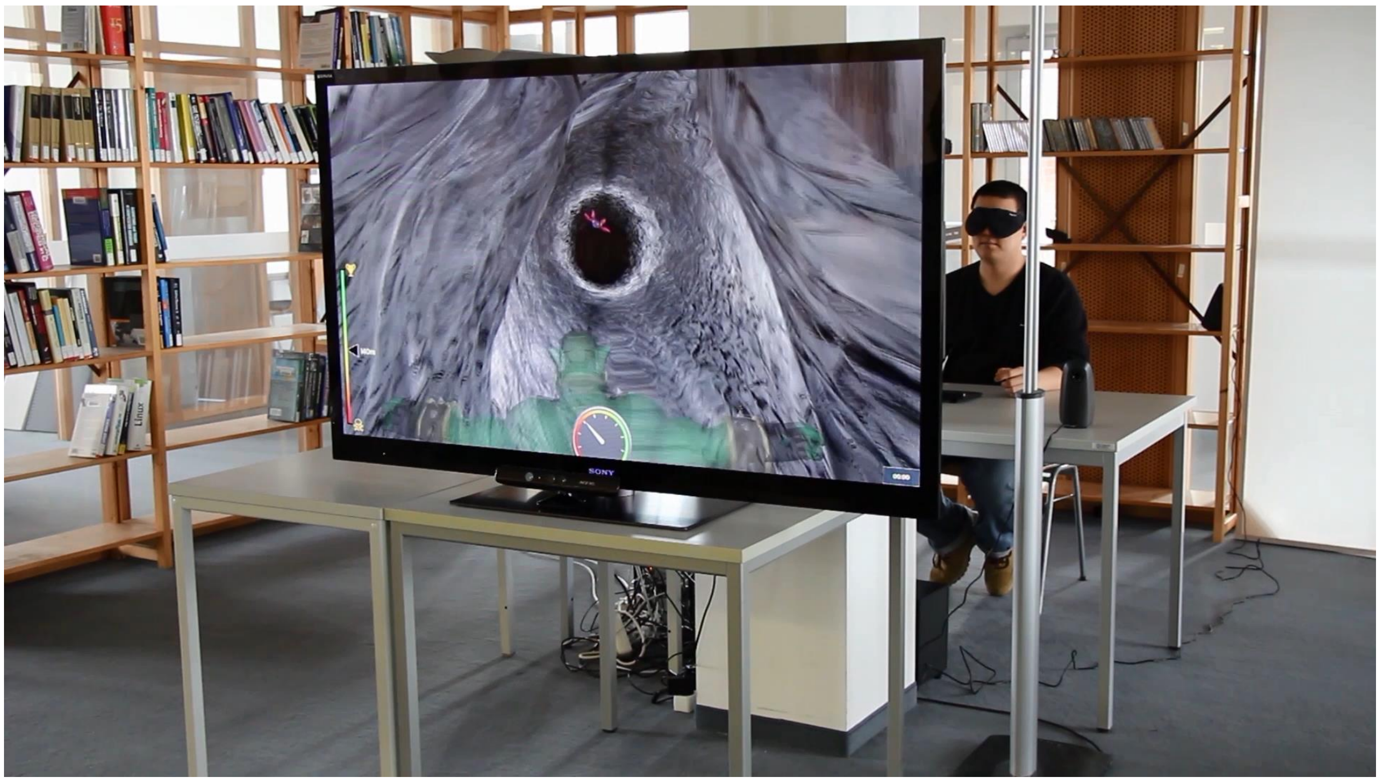

Max time / msec
Pig
Cow
Armadillo
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
0
10
20
30
40
50
60
70
Number of Spheres / 1000

Applications: A Multi-User Haptic Workspace

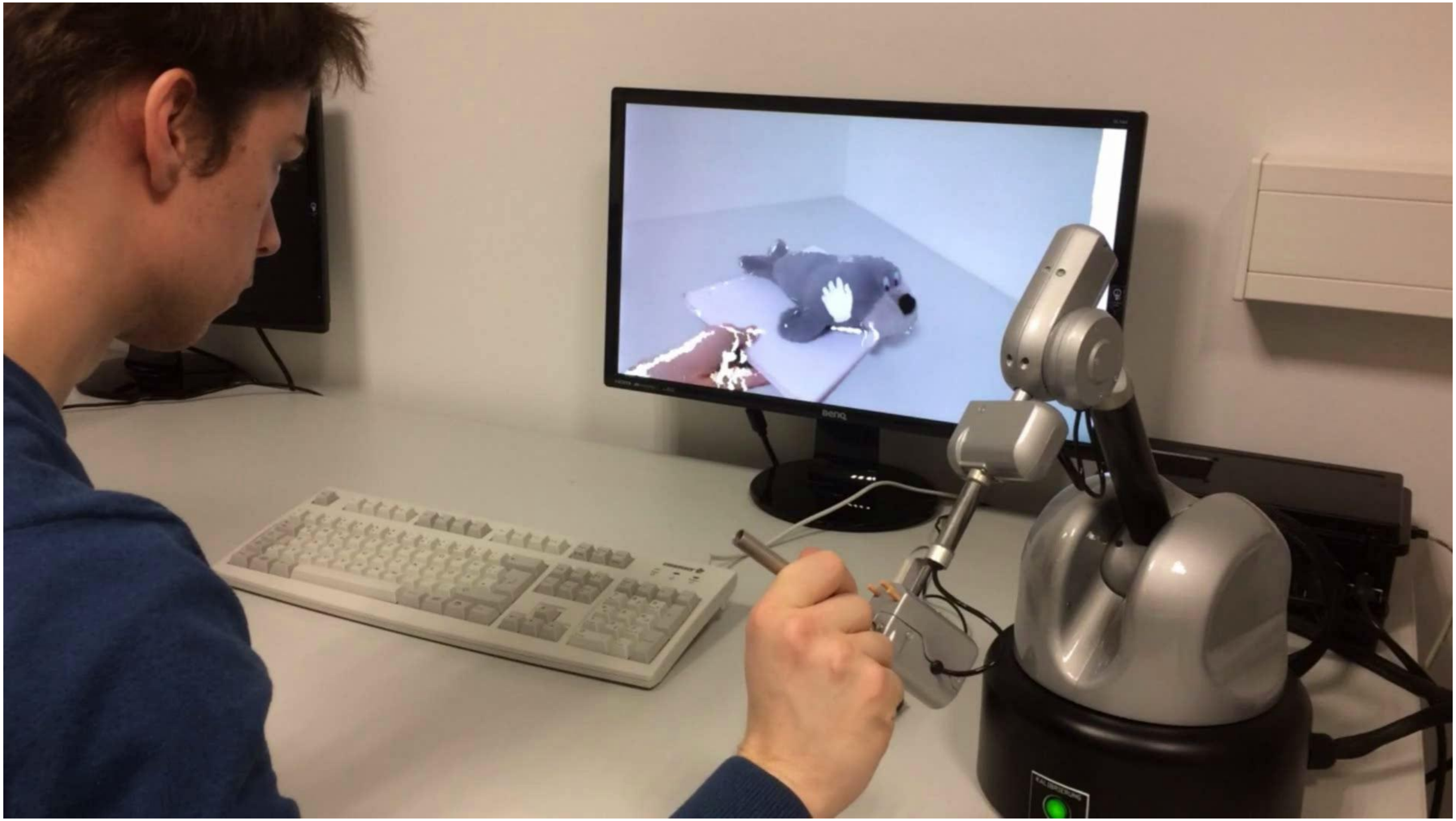
- Two user bi-manual interaction
- >12 dynamic objects,
  triangle count > 2 Millions
- Stable 1000Hz simulation rate on normal consumer PC
- Game to evaluate the influence of the degrees of freedom in haptics

Applications: KINAPTIK

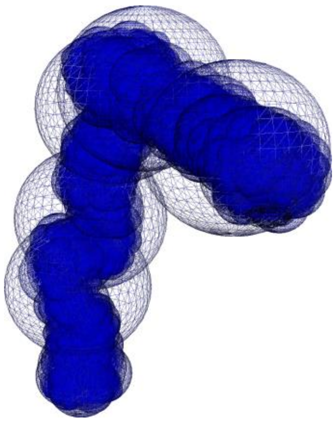
- The first competitive 3D multiplayer game for both sighted and blind people

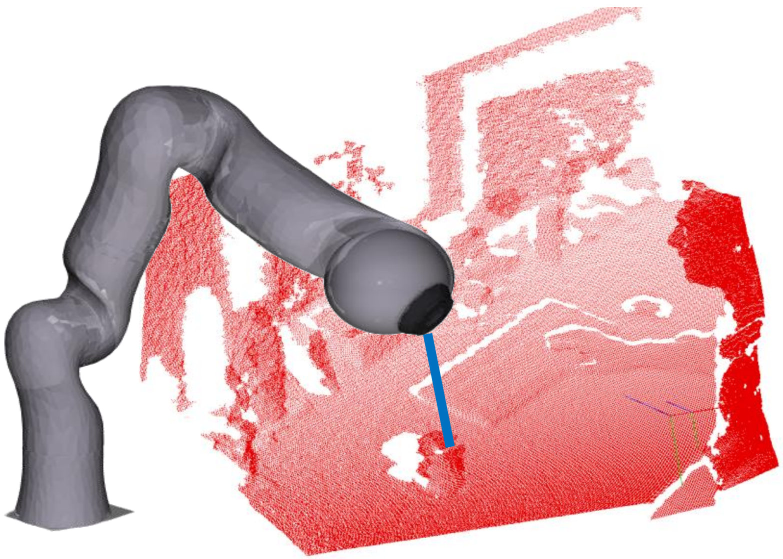


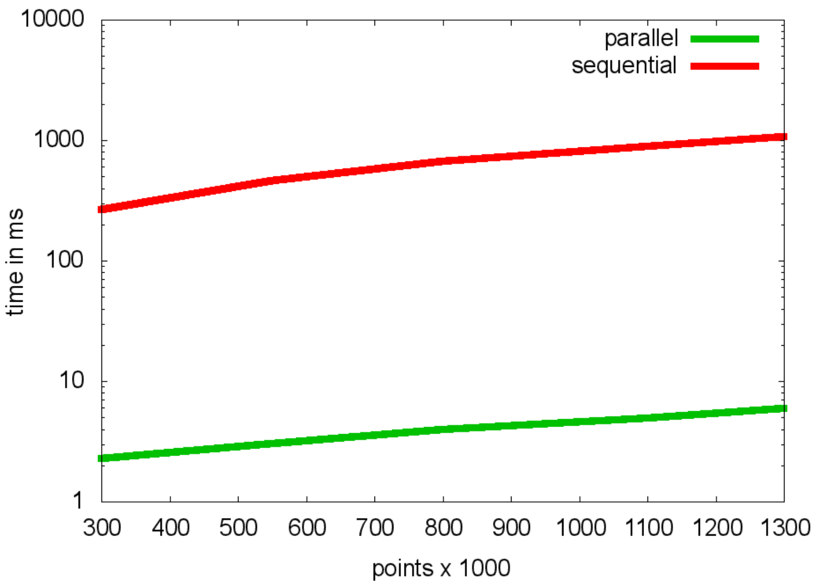
6-DOF Haptics for Streaming Point Clouds



Applications: Collision Avoidance in Robotics

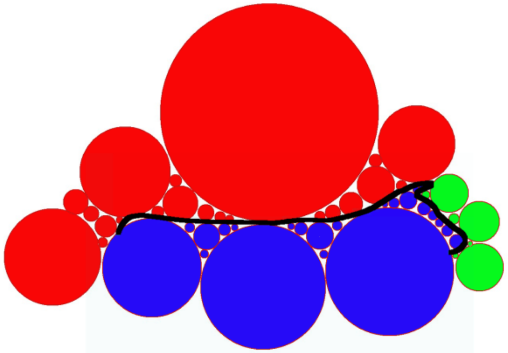


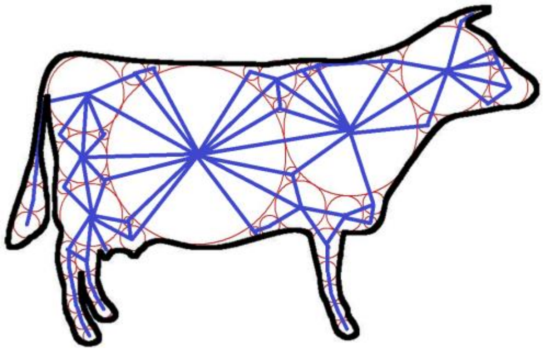




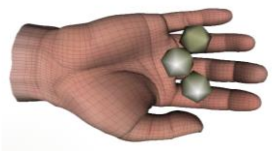
Conclusions and Future Challenges

- First algorithm to compute the penetration volume efficiently
- Independent of object complexity
- Worst case O(n) overlapping spheres
  - O(1) parallel time complexity
- <1 msec for > 200k spheres
- Continuous forces and torques
- Future works
  - Thin sheets
  - Deformable objects
  - Rendering
  - Application to other problems
    - Segmentation, Reconstruction,...

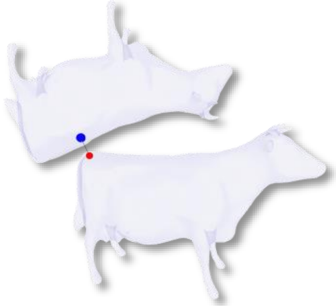




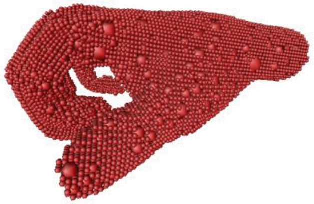
Thank You!



Protosphere



Kinetic CollDet



KINAPTIK

René Weller

weller@cs.uni-bremen.de
http://cgvr.cs.uni-bremen.de