CDFC: Collision Detection Based on Fuzzy Clustering for Deformable Objects on GPUs

David Mainzer[1] and Gabriel Zachmann[2]

[1]Clausthal University, Germany, dm@tu-clausthal.de and [2]University of Bremen, Germany, zach@cs-bremen.de

Our Contributions

- **Reports intersections and self-intersections**
- **Works for arbitrary polygon soups**
- **Works for arbitrary & unknown motions of the primitives**
- **No need do distinguish between broad and narrow phase**

```
Algorithm 1 GPU-based Collision Detection
Each line is mapped to a massively parallel computation kernel
Input: primitives of all objects
Output: intersecting pairs of primitives
  subdivide scene into $c$ clusters using fuzzy C-means
  for all clusters do in parallel
      compute and apply PCA
      sort AABBs along longest principle axis
      collect all overlapping intervals
      for all overlapping intervals do in parallel
          if AABB intersect along y-axis then
              do primitive-primitive intersection test
          end if
      end for
  end for
```

Sweep-Plane Technique using PCA

- All primitives in a scene → huge amount of false positives if using a **fixed** world coordinate axis as sweep direction
- **Best** sweep direction separates the primitives as much as possible
- **Principle component analysis** (PCA): direction of first principal component maximizes variance

Segmentation Process

- Problem: sweeping essentially projects boxes down to intervals $\mapsto$ primitives are potentially colliding, even if there is a large distance inbetween
- Solution: subdivide scene into connected components using **fuzzy C-Means**
- Condition on clusters: Clusters in close proximity have to be connected, otherwise collisions across border of clusters would be missed

Results

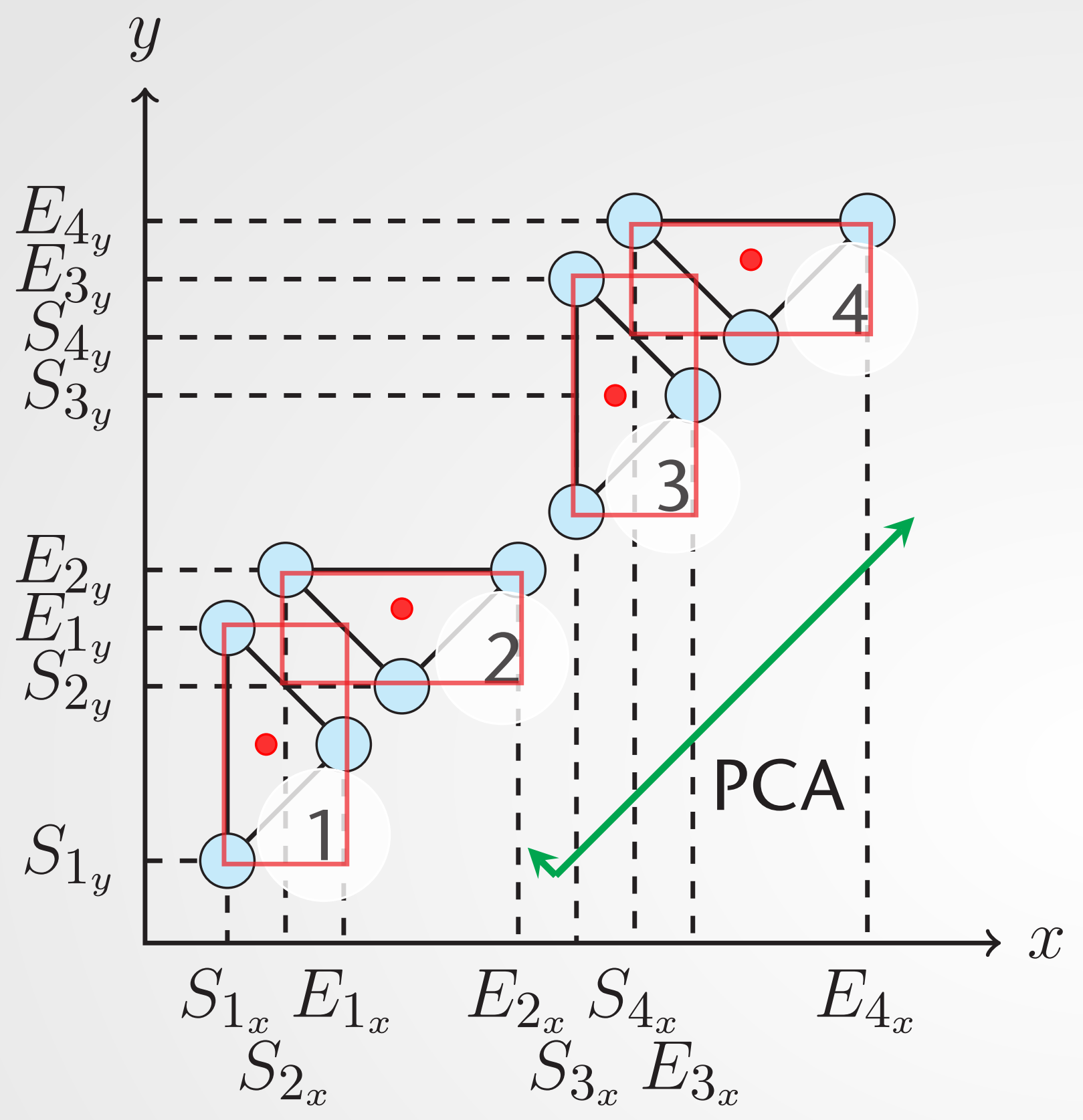
- NVIDIA GeForce GTX 480 & CUDA Toolkit 5.0
- External and self-collision detection

Bench.	Our	CSt.[3]	Pab.[2]	HP[1]	MC[4]
Cl. on Ball	20.24ms	18.6ms	36.6ms	23.2ms	32.5ms
Funnel	6.53ms	4.4ms	6.7ms	–	–

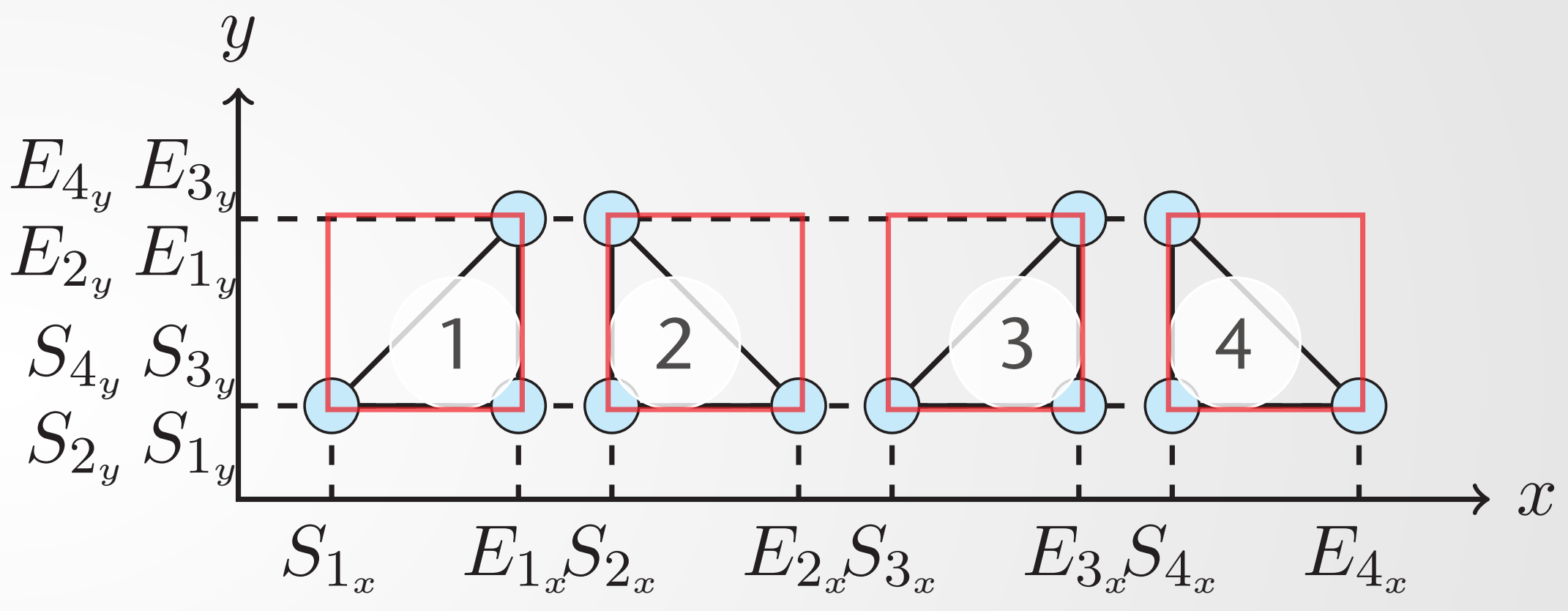
- Benchmark **Cloth on Ball**: Cloth (92k triangles) drops down on a rotating ball (760 triangles)
- Benchmark **Funnel**: Cloth (14.4k triangles) falls into a funnel (2k triangles) and passes through it, due to the force applied by a ball (1.7k triangles)

Sweep-Plane Approach via PCA

- Compute covariance matrix over all centroid points of all primitives

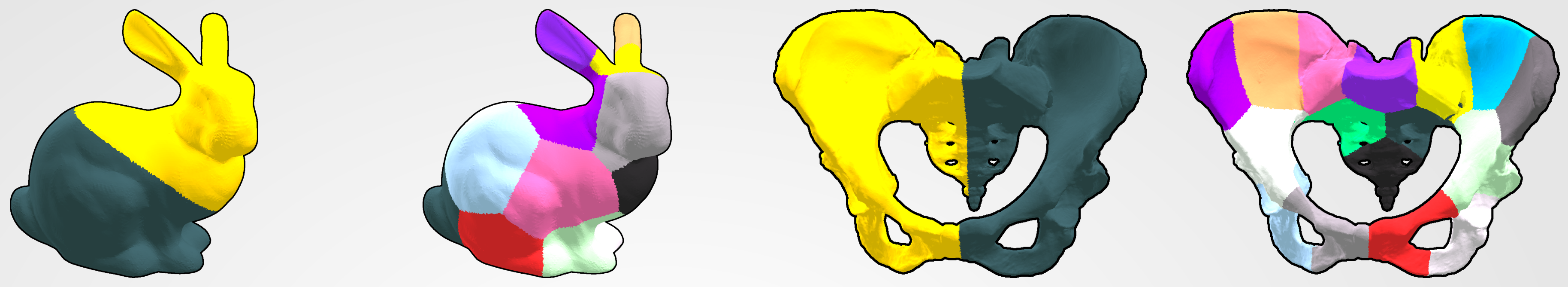


- Left: Initial scene and result of PCA
- Right: Scene in the local coordinate system of the PCA

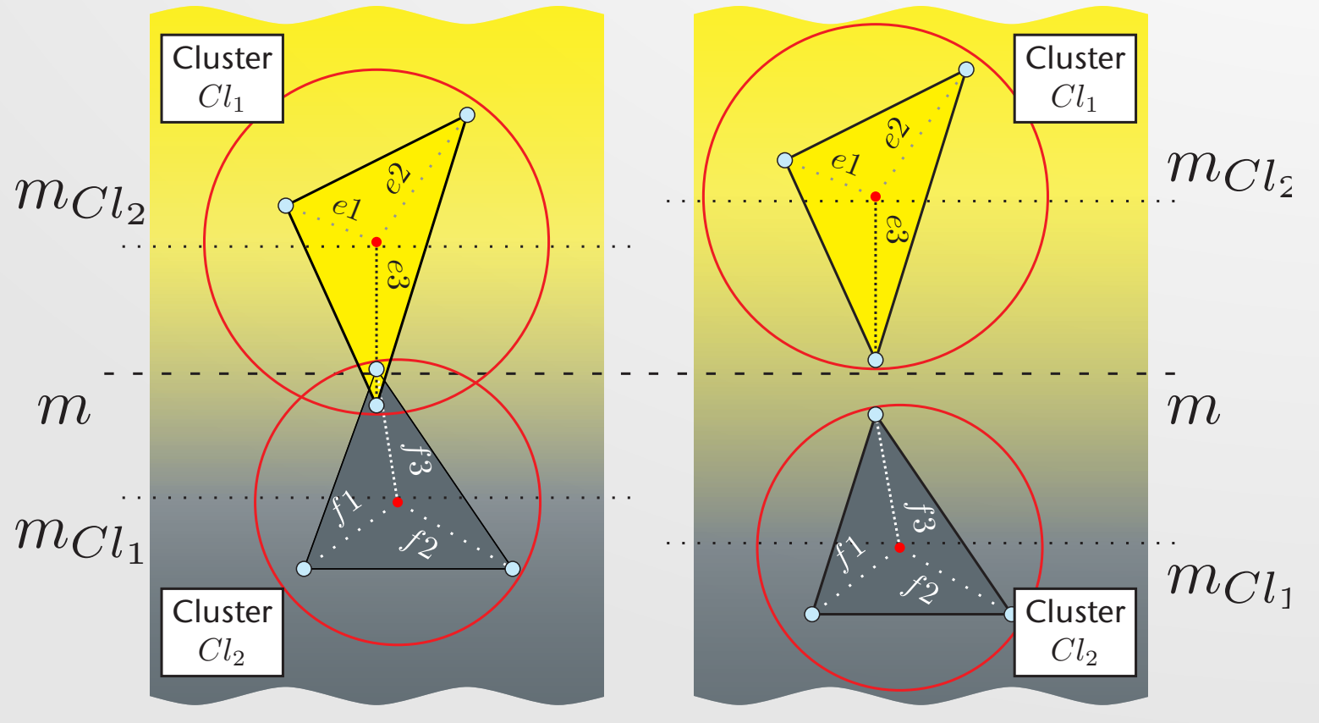


Scene/Object Subdivision using Fuzzy C-Means

Examples of some high-detailed objects, partitioned by fuzzy C-means

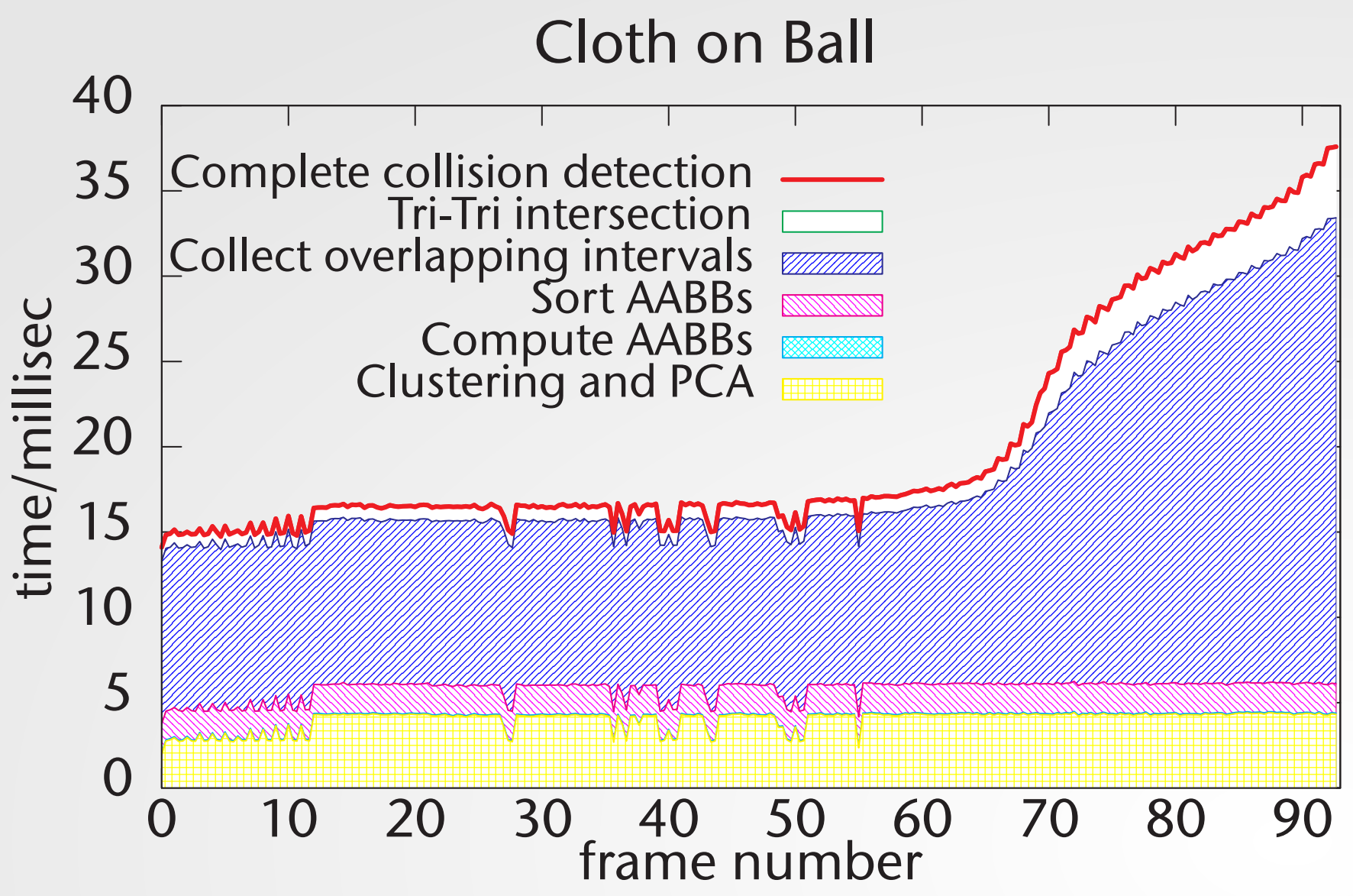


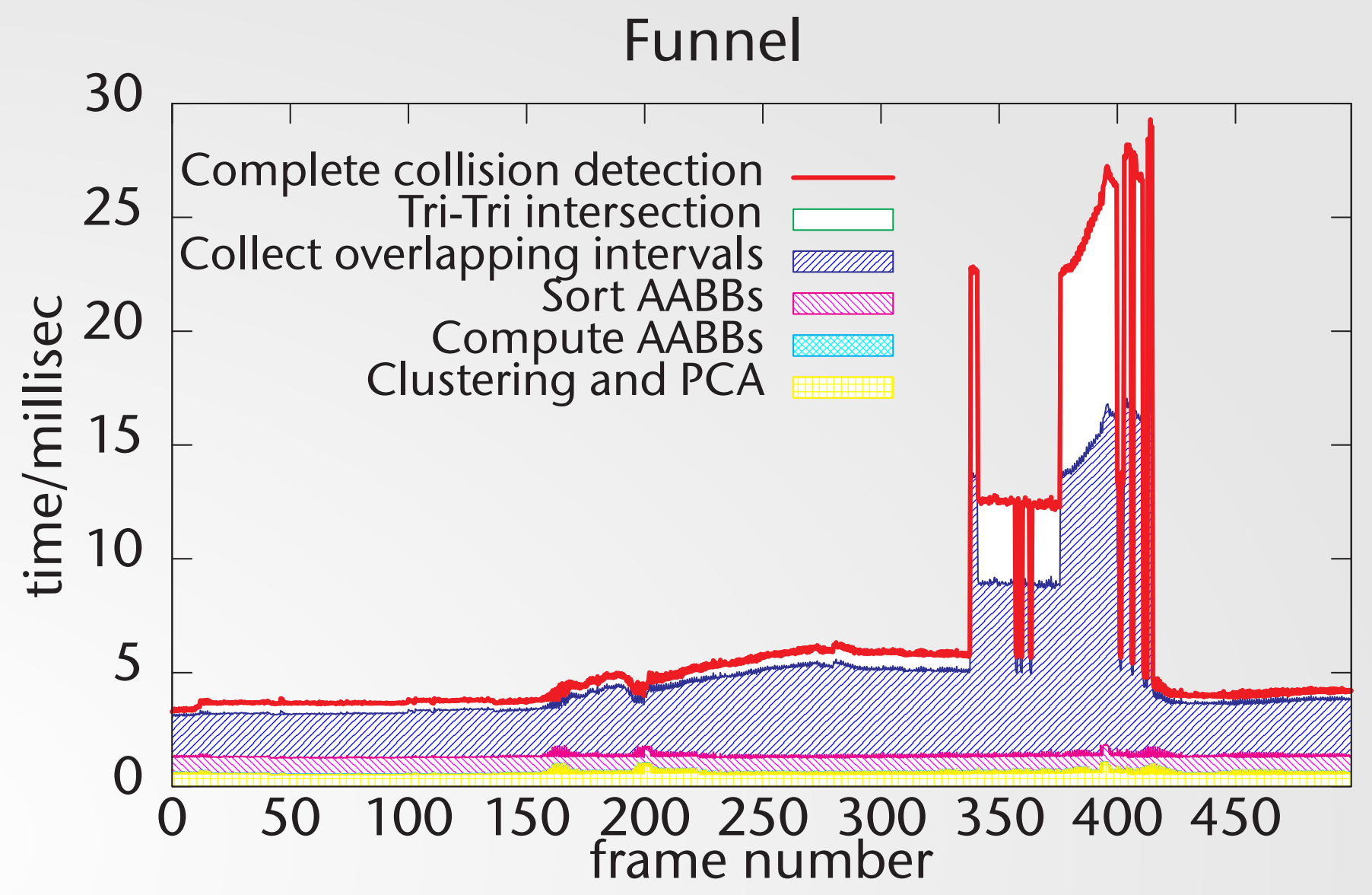
- Overlap between clusters in **close proximity** $= \|m - m_{Cl_i}\|_2 \geqq \max_{i=1,2,\dots,n}(\max_{k=0,1,2}(\|C_i - vertex_{i,k}\|_2))$ where $m =$ hard clustering border, $m_{Cl_i} =$ fuzzy clustering border, $C_i =$ centroid of primitive i, $vertex_{i,k} =$ vertex k of primitive i

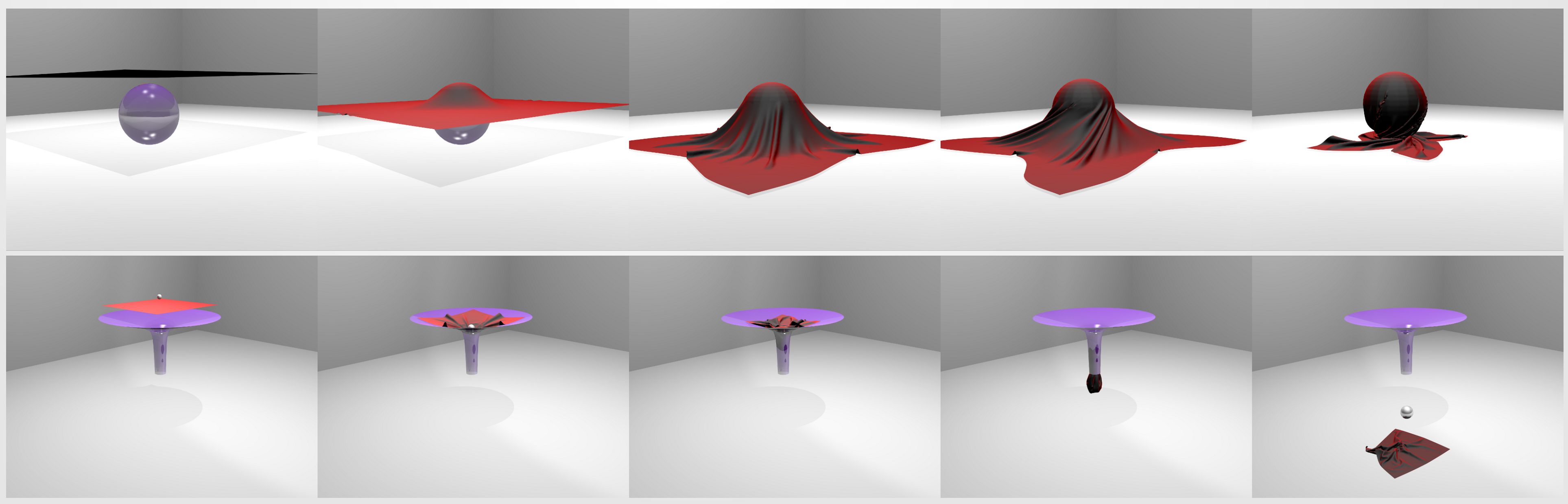


- Membership value $\in [0,1]$ in the clustering step controls size of cluster overlap → trade-off between cluster overlap and performance

Benchmark Timings







References

[1] D. Kim, J.P. Heo, J. Huh, J. Kim, and S. Yoon. Hpccd: Hybrid parallel continuous collision detection using cpus and gpus. In *Computer Graphics Forum*, volume 28, pages 1791-1800. Wiley Online Library, 2009.

[2] S. Pabst, A. Koch, W. Straßer. Fast and scalable cpu/gpu collision detection for rigid and deformable surfaces. In *Computer Graphics Forum*. volume 29, pages 1605-1612. Wiley Online Library, 2010.

[3] M. Tang, D. Manocha, J. Lin, and R. Tong. Collision-streams: fast gpu-based collision detection for deformable models. In *Symposium on Interactive 3D Graphics and Games*, pages 63-70. ACM, 2011.

[4] M. Tang, D. Manocha, and R. Tong. Mccd: Multi-core collision detection between deformable models using front-based decomposition. *Graphical Models*, 72(2):7-23, 2010.

WSCG 2013, Plzeň, Czech Republic