Fast and Reliable Contact Computations for Grasp Planning

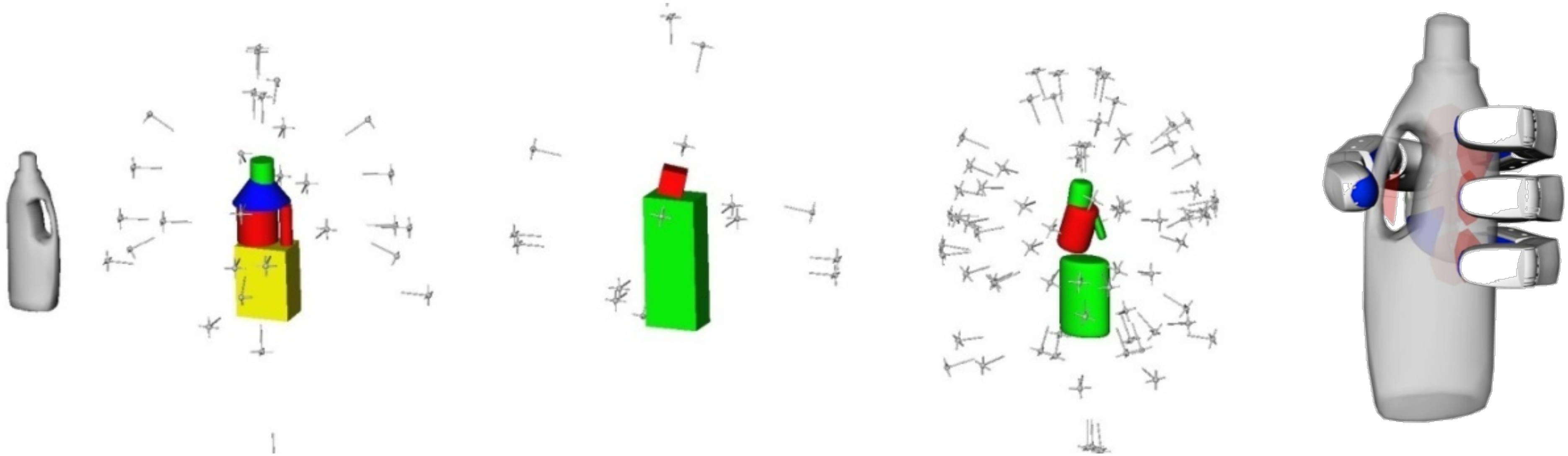
Young J. Kim, Min Tang
Ewha Womans Univ
Seoul, Korea♪

Zhixing Xue
KIT
Karlsruhe, Germany♪

Dinesh Manocha
UNC
Chapel Hill, U.S.A.♪

Forward Grasp Planning

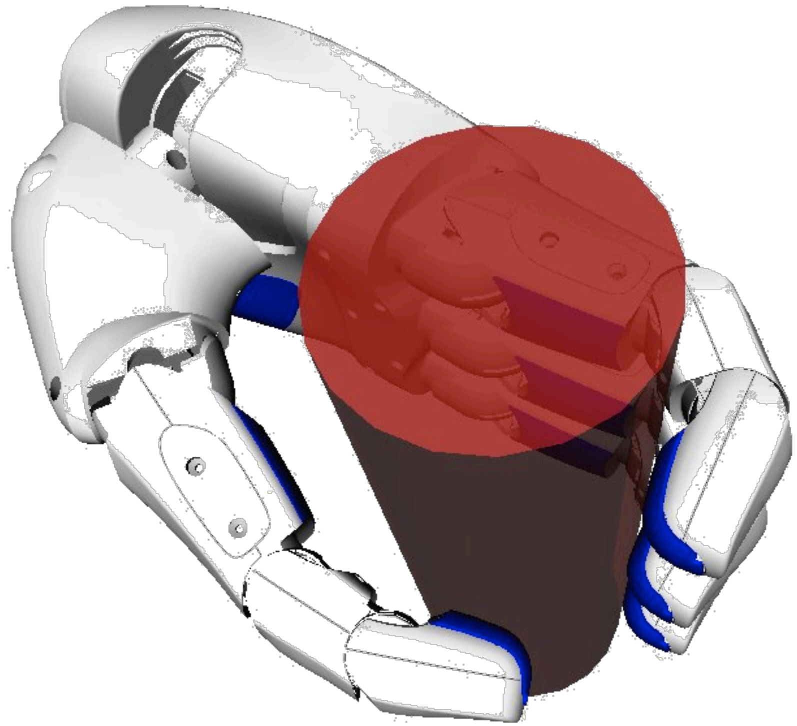
1. Generate an approach direction
2. Find contact points
3. Measure the grasp quality
4. Repeat this process



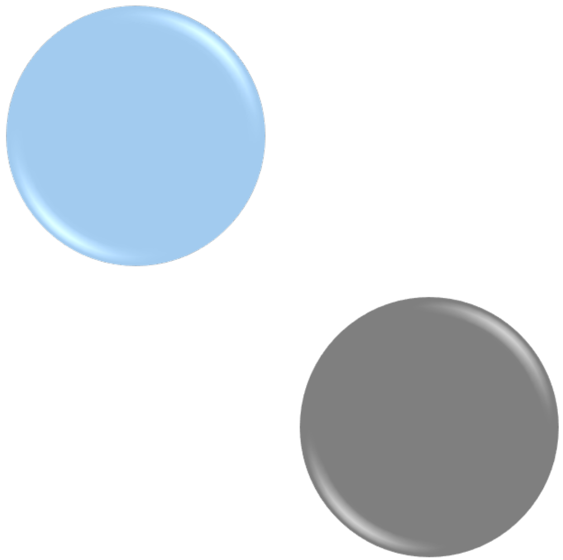
Challenge

- Given grasp approach ing directions

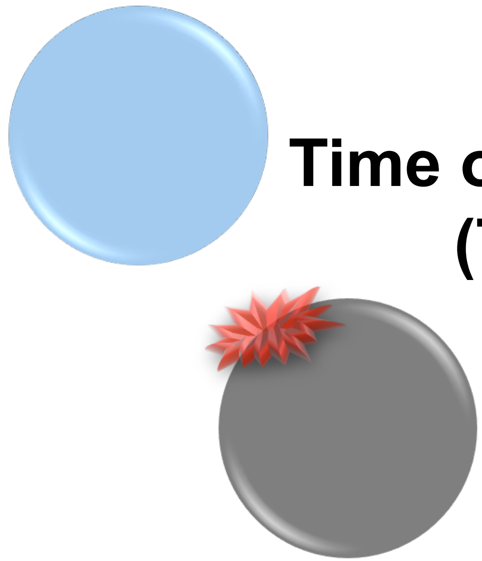
- Find all the contact po ints fast and reliably



Goal of Contact Computations♪

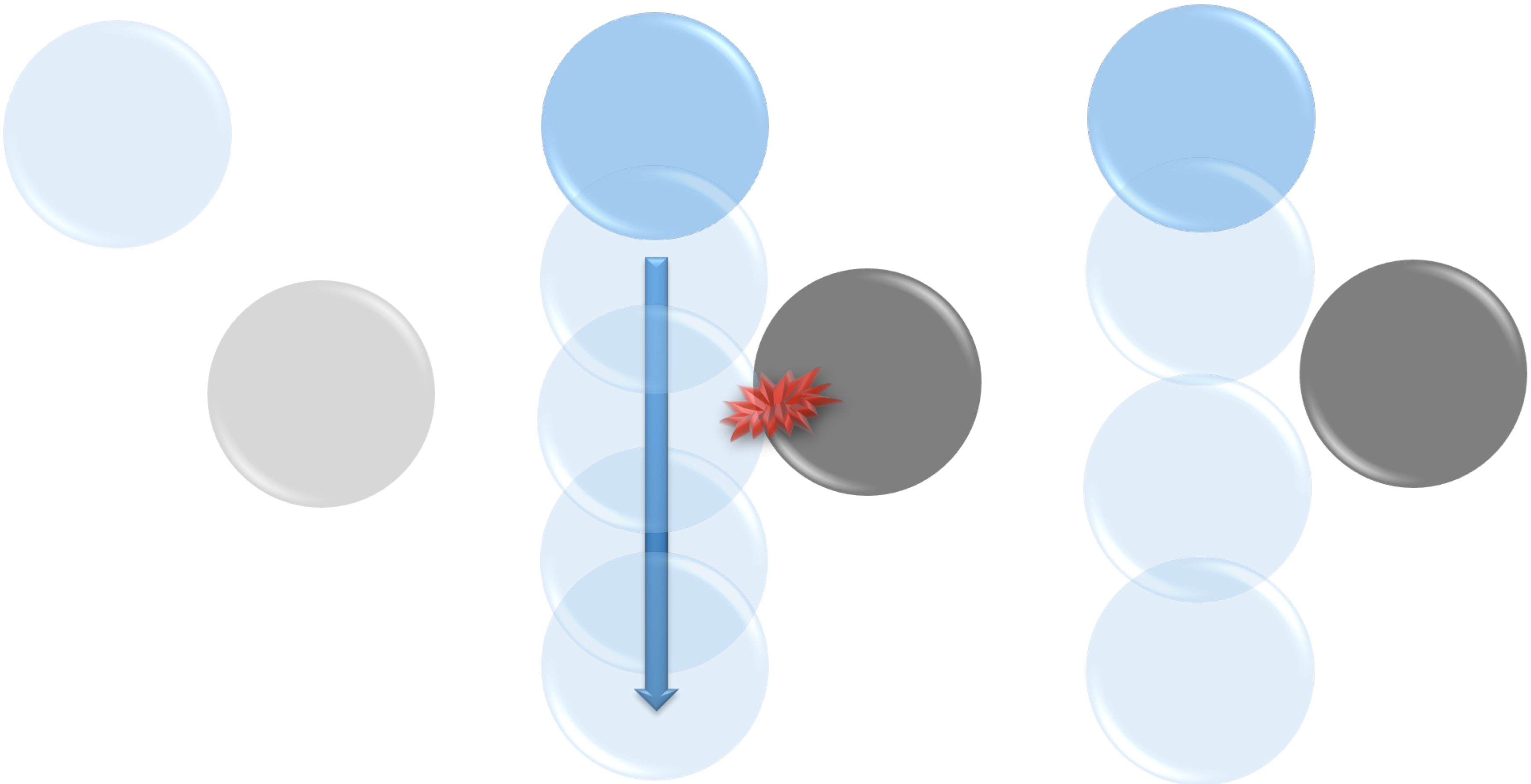


No Collision♪



Collision♪

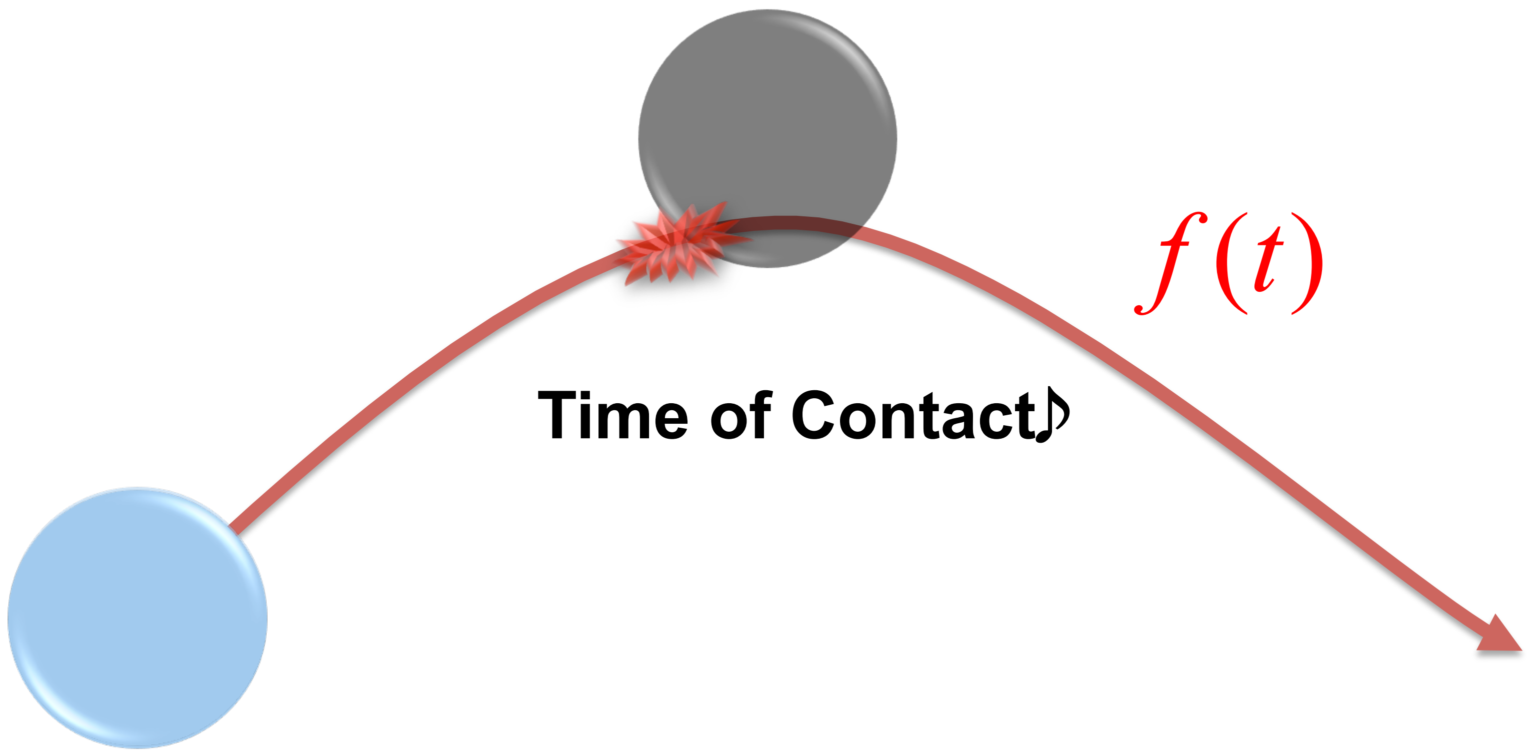
Discrete Collision Detection♪



Collision Missing♪

Continuous Collision Detection (CCD)♪

- Motion trajectory ***f(t)*** is known in advance♪

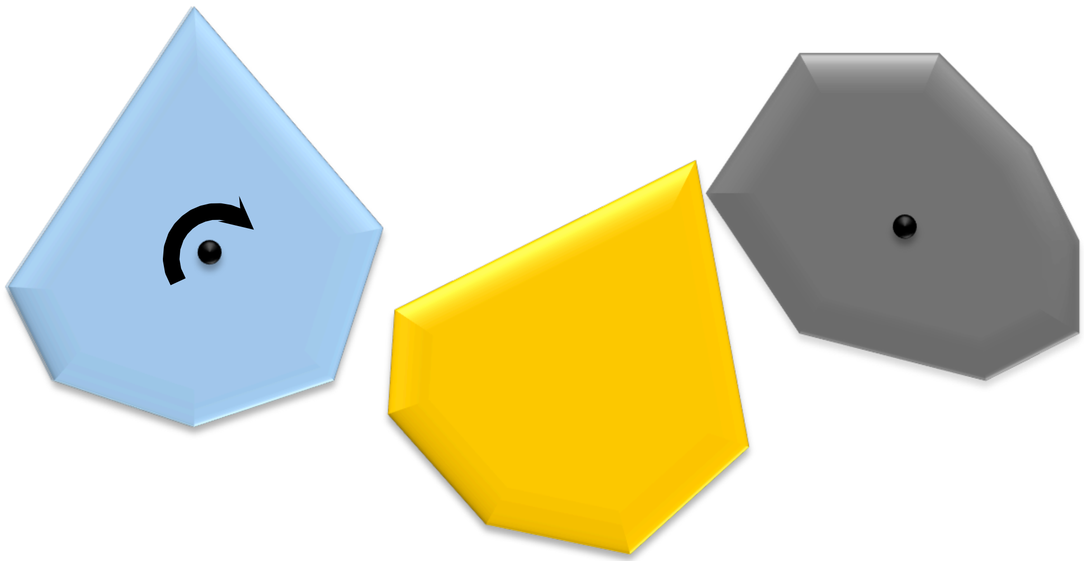


Outline

- Continuous collision detection (CCD)
  - Convex
  - Non-convex
  - Polygon-soup
  - Articulated model
  - Deformable model
- Evaluation of CCD in grasp planning

Conservative Advancement (CA)♪

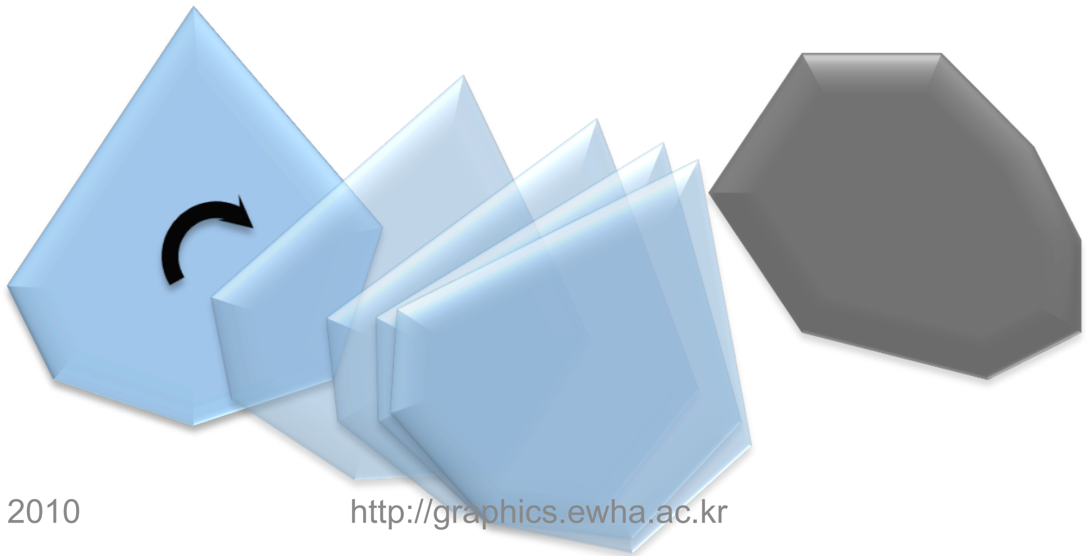
- Assume objects are *convex*
- Find the 1st time of contact (ToC) of a moving object



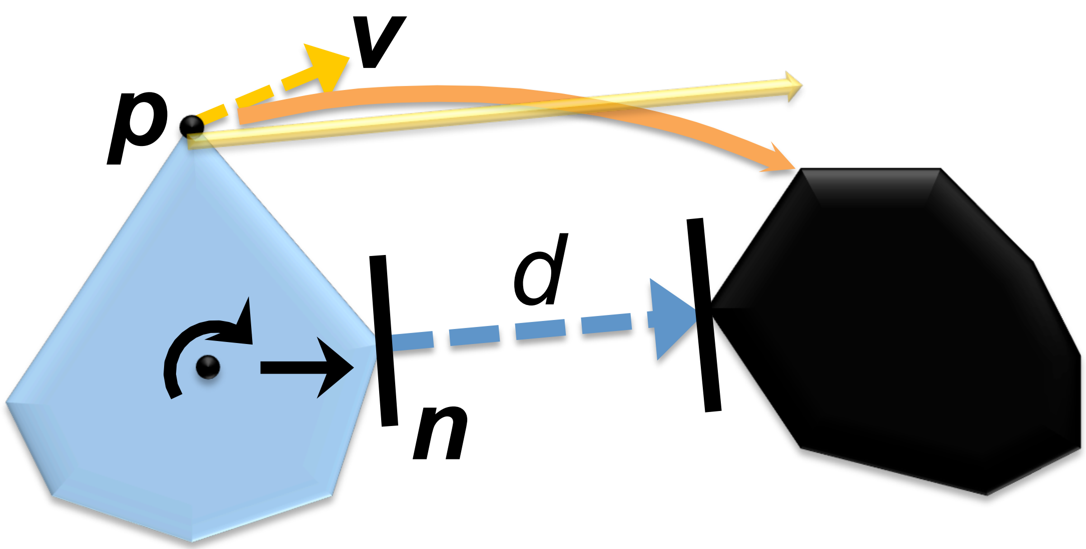
Conservative Advancement (CA)♪

1. Find a step size Δt_i to conservatively advance the object without collision
2. Repeat until inter-distance < ε

$$ToC = \Delta t_1 + \Delta t_2 + \Delta t_3 + \Delta t_4$$



Calculating Δt in CA



d, ***n***: closest distance, direction vector

v: velocity

$$\int_0^{\Delta t} |v(t) \bullet n(t)|\,dt \le \int_0^{\Delta t} \max(|v(t) \bullet n(t)|)\,dt \le d$$

$$= \mu$$

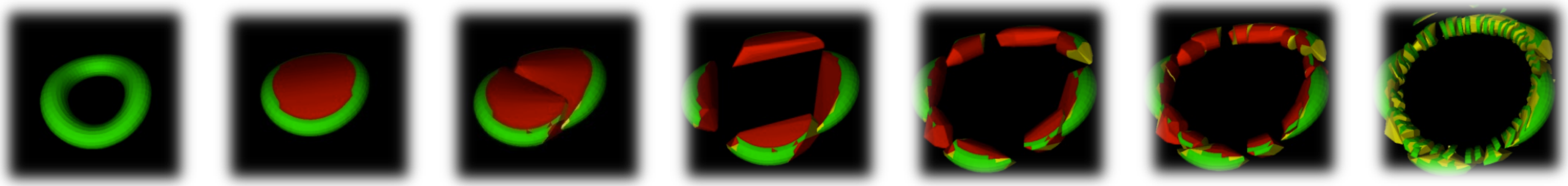
Calculating Δt in CA

$$\mu \Delta t \le d \qquad \therefore \Delta t \le \frac{d}{\mu}$$

$$\int_0^{\Delta t} | v(t) \bullet n(t) | \, dt \le \int_0^{\Delta t} \underbrace{\max(| v(t) \bullet n(t) |)}_{= \mu} dt \le d$$

Extension to Non-convex [PG06]

- Use of convex decomposition
- Build a hierarchy of decomposed convex pieces and perform CA *hierarchically*



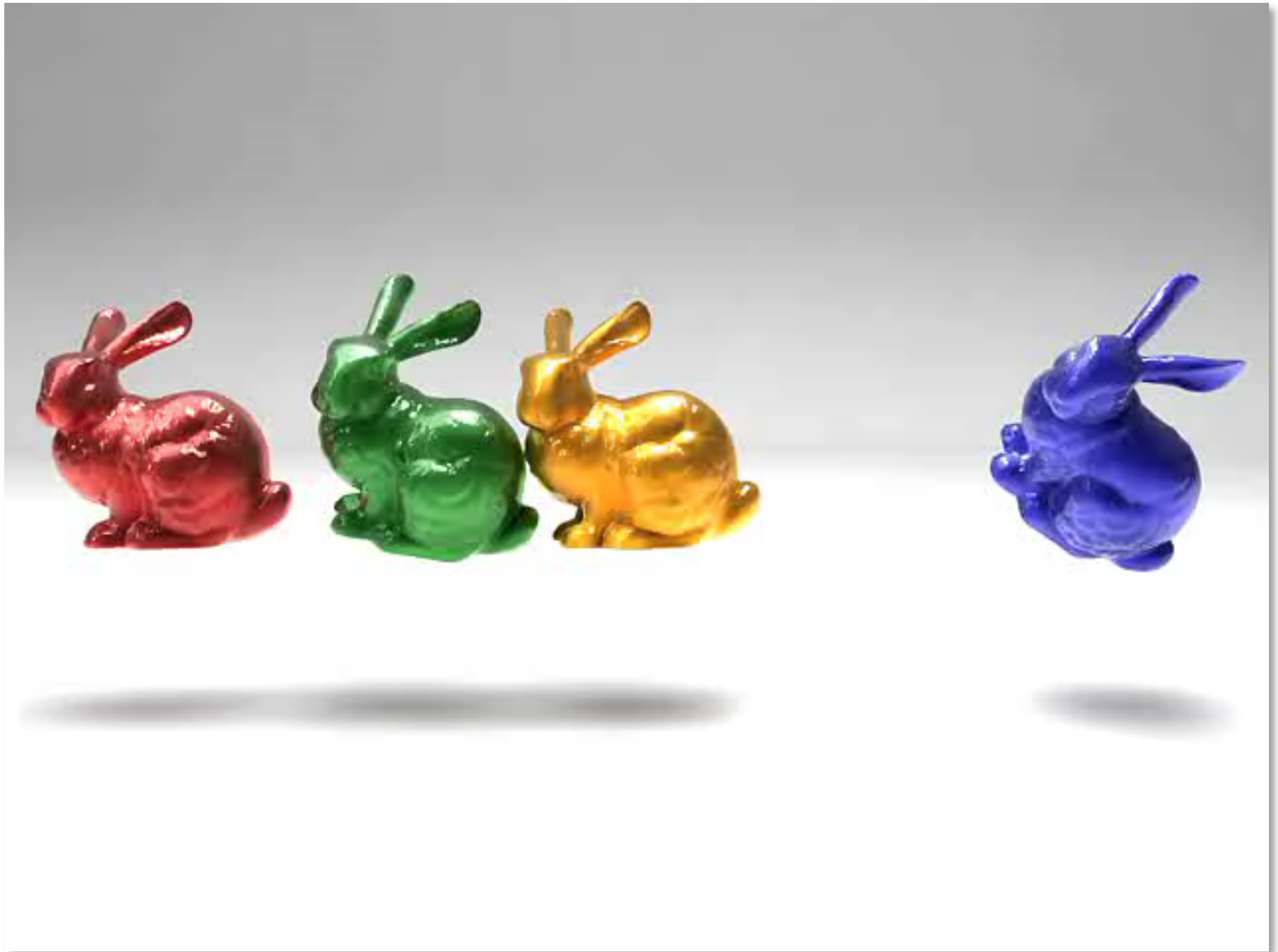
Santa vs. Thin Board

37K triangles

51,546 FPS

# of iterations
3.68

Bunny vs. Bunny

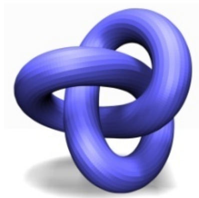


70K triangles/bunny 110 FPS # of iterations 4.7

Torusknot vs. Torusknot

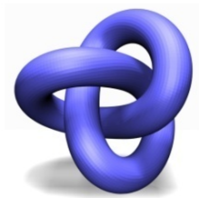
2.8K
1067 FPS

# of iterations 4.49



11K
400 FPS

# of iterations 4.49

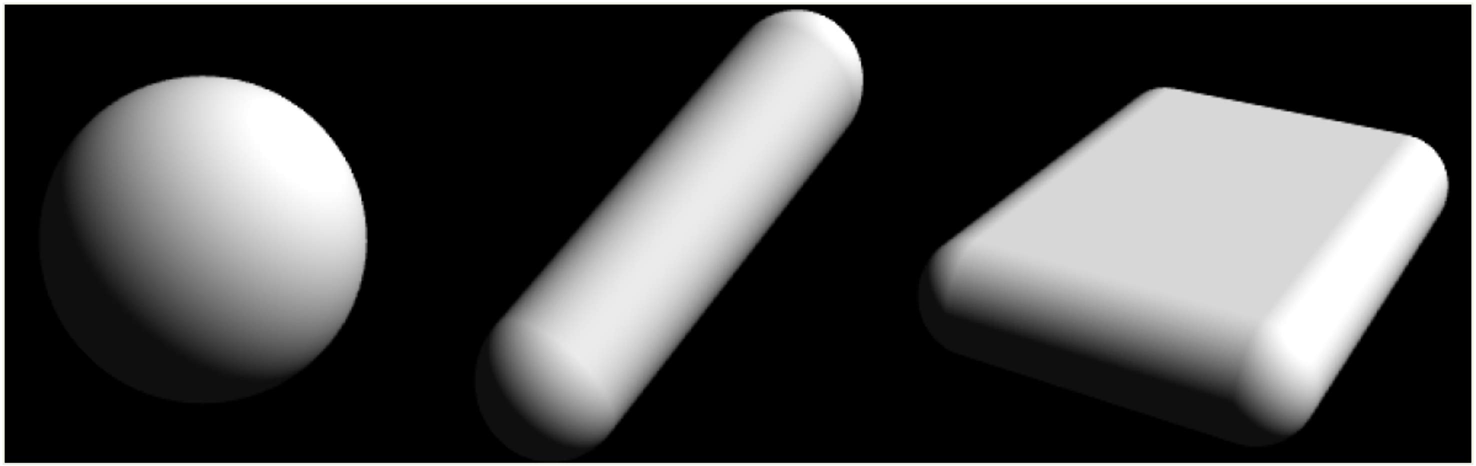


34K
186 FPS

# of iterations 4.46

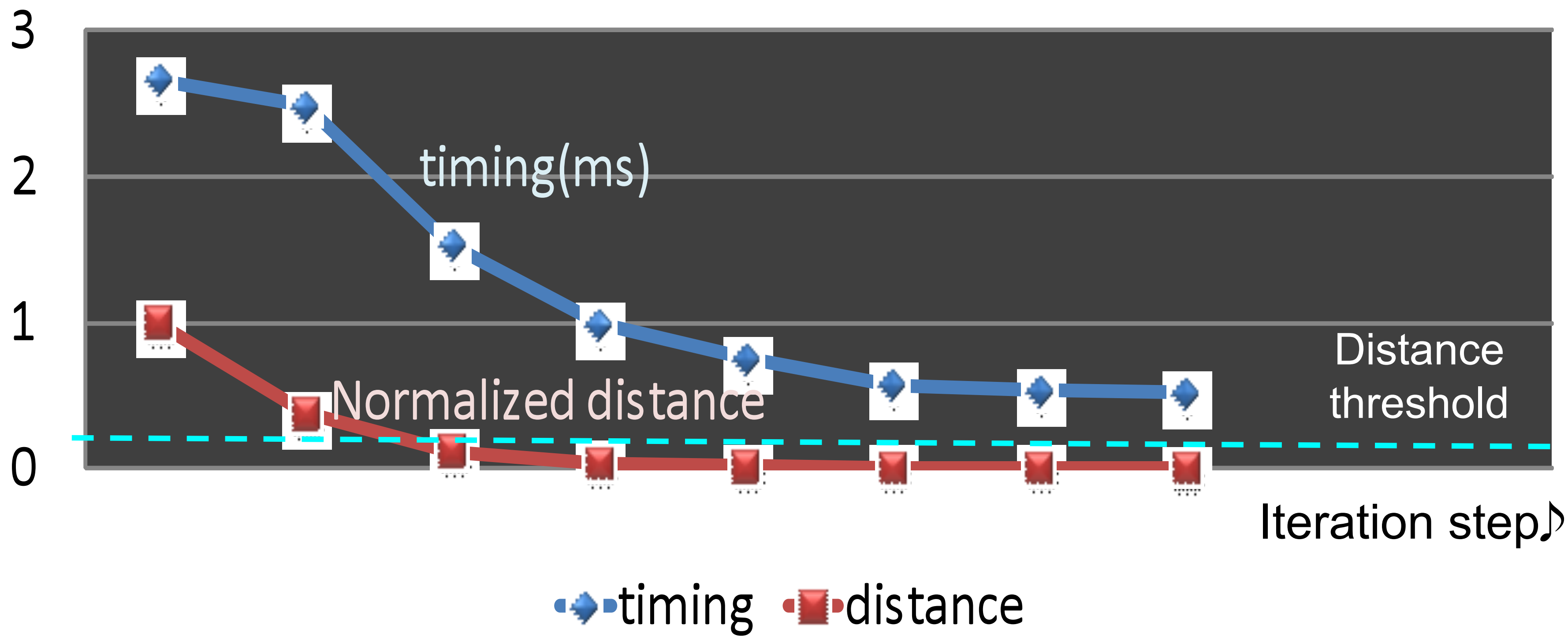
Extension to Polygon-Soups [ICRA09]

- Swept Sphere Volume (SSV) [Larsen 1999]:
  - Point Swept Sphere (PSS)
  - Line Swept Sphere (LSS)
  - Rectangle Swept Sphere(RSS)

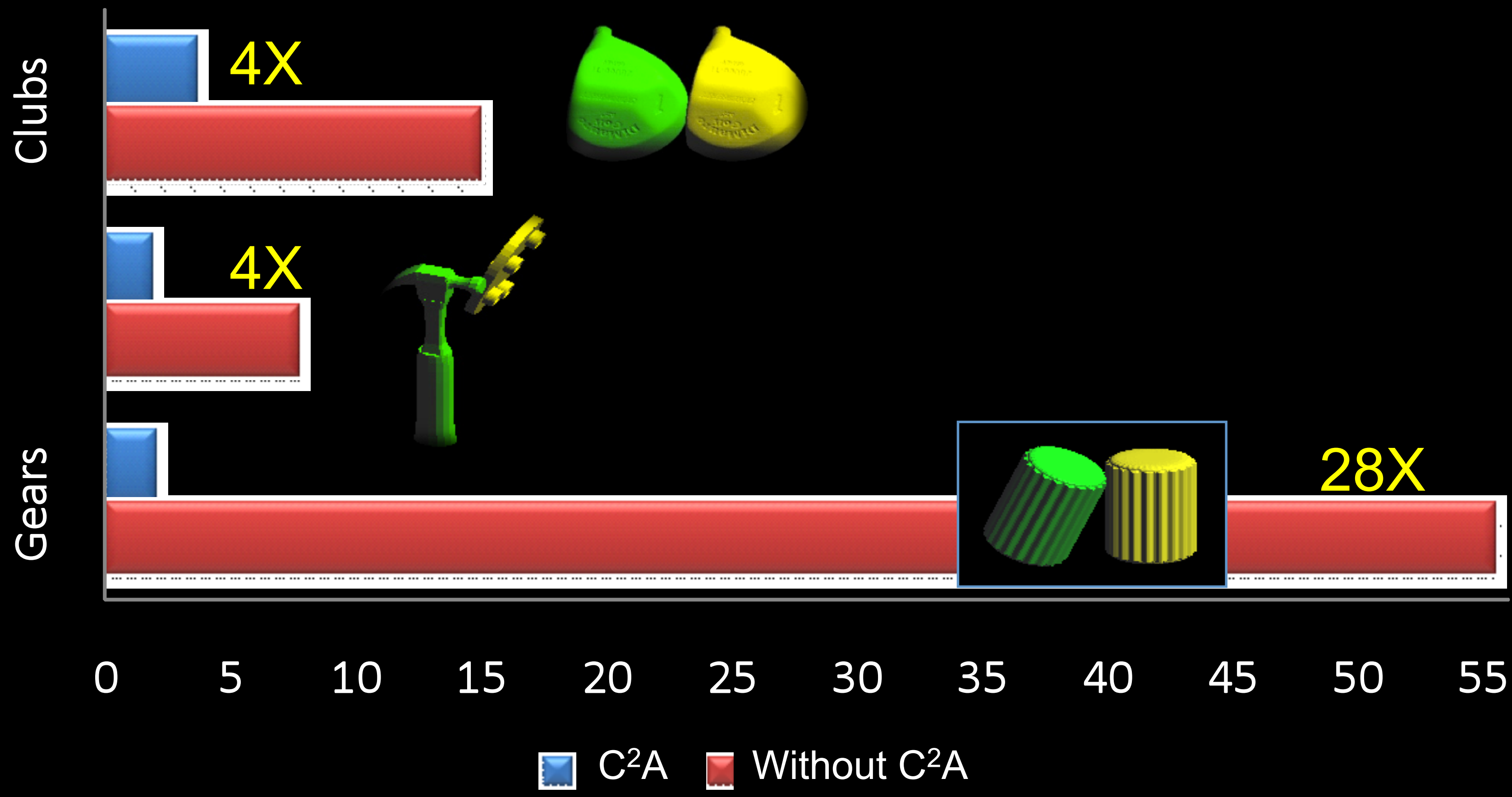


PSS♪ LSS♪ RSS♪

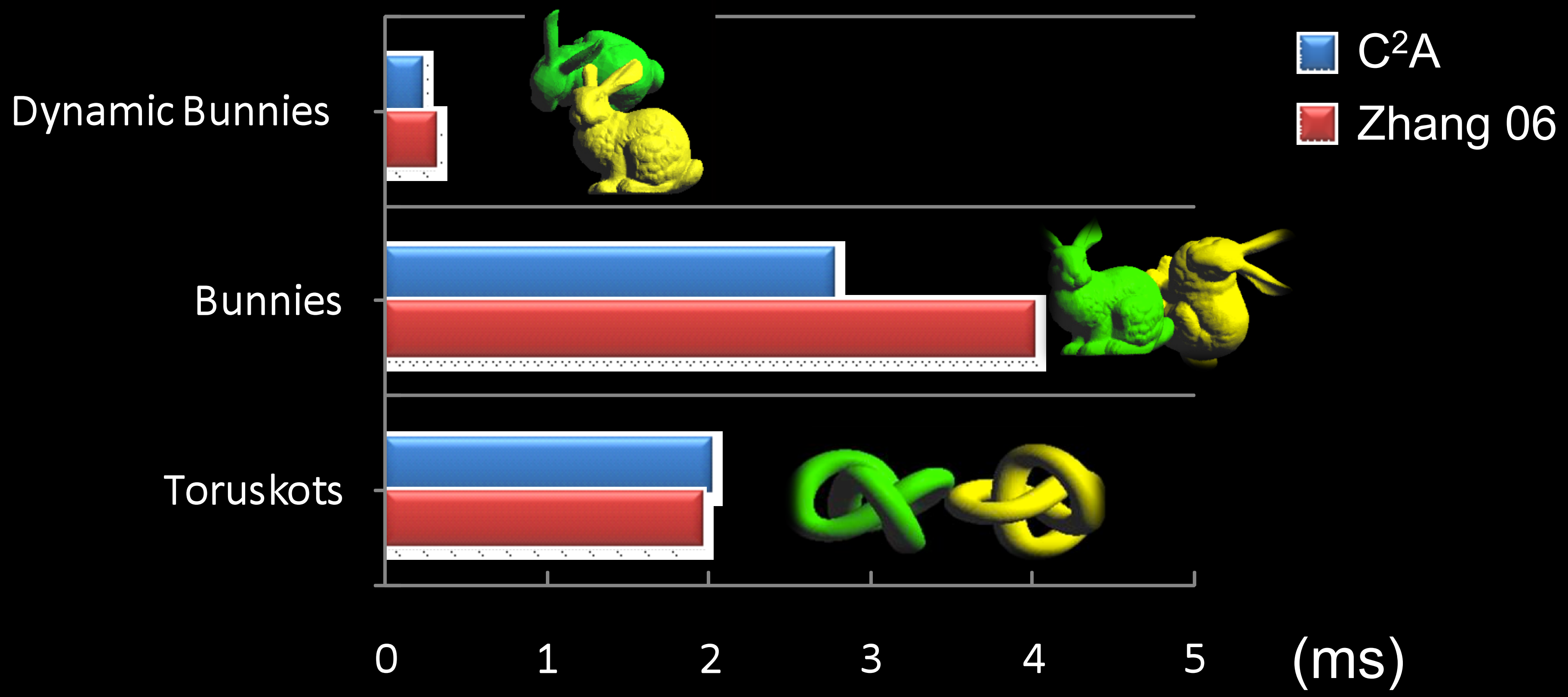
Controlling the Depth of BVH Traversal



Results - Timings



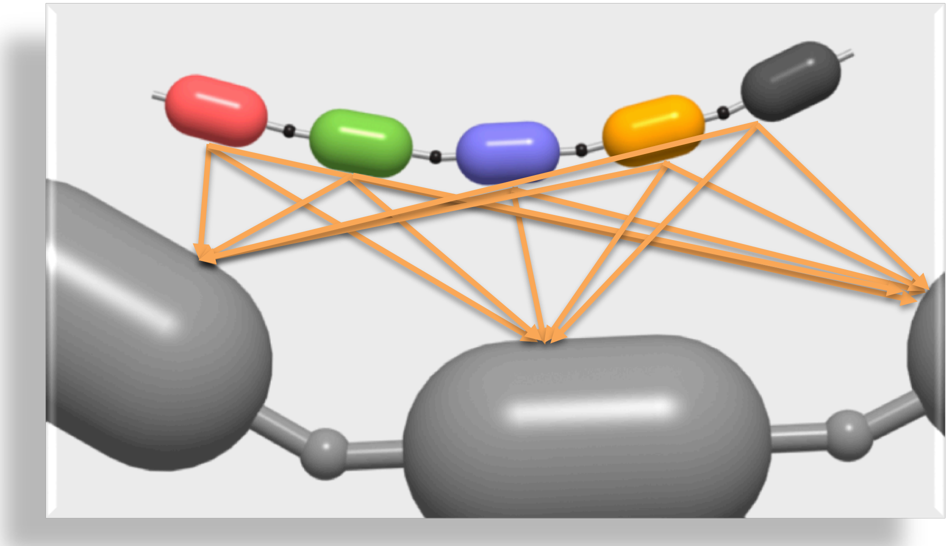
Comparisons against [Zhang 06]



- [Zhang 06] can handle only manifold surfaces

Extension to Articulated Models [SIGGRAPH07]♪

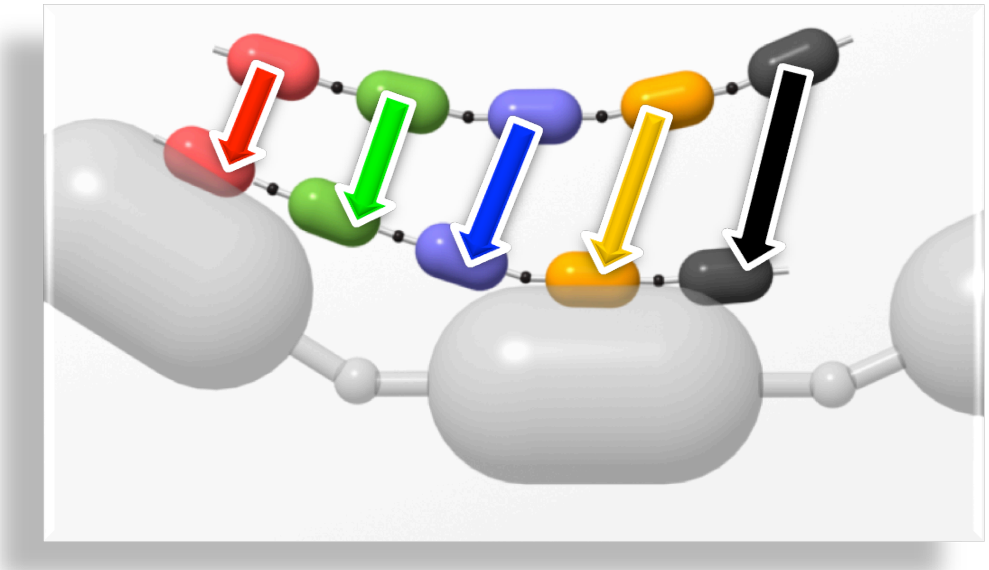
- Treat each link as a rigid body
- Apply CA to each link independently
- Taking the minimum of CA results



Link Culling♪

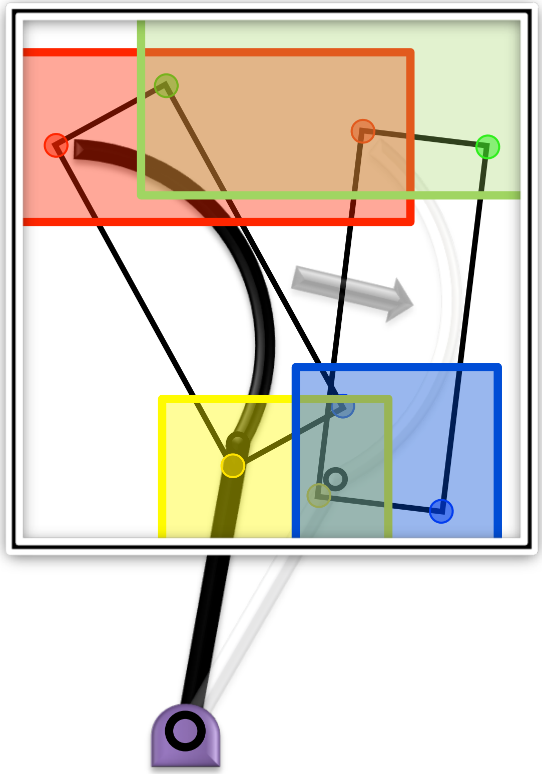
1. Some links are collid ed later than others

2. Some links are not e ven collided during t he entire time interva l♪



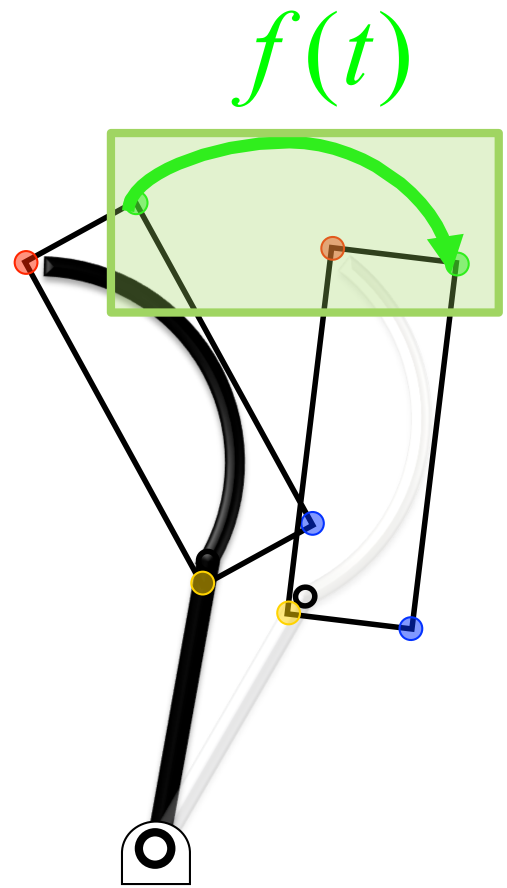
Dynamic AABB Construction

- Idea
  1. Pre-compute AABBs for links
  2. Compute the AABBs of vertic es of moving AABBs
  3. Compute the enclosing AAB B♪



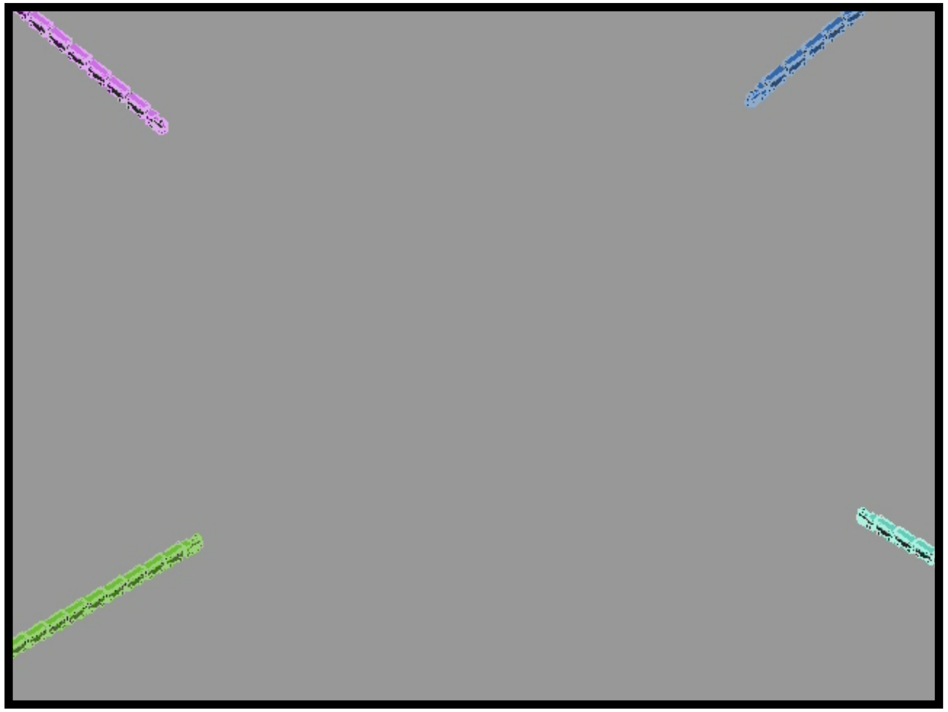
Dynamic AABB Construction

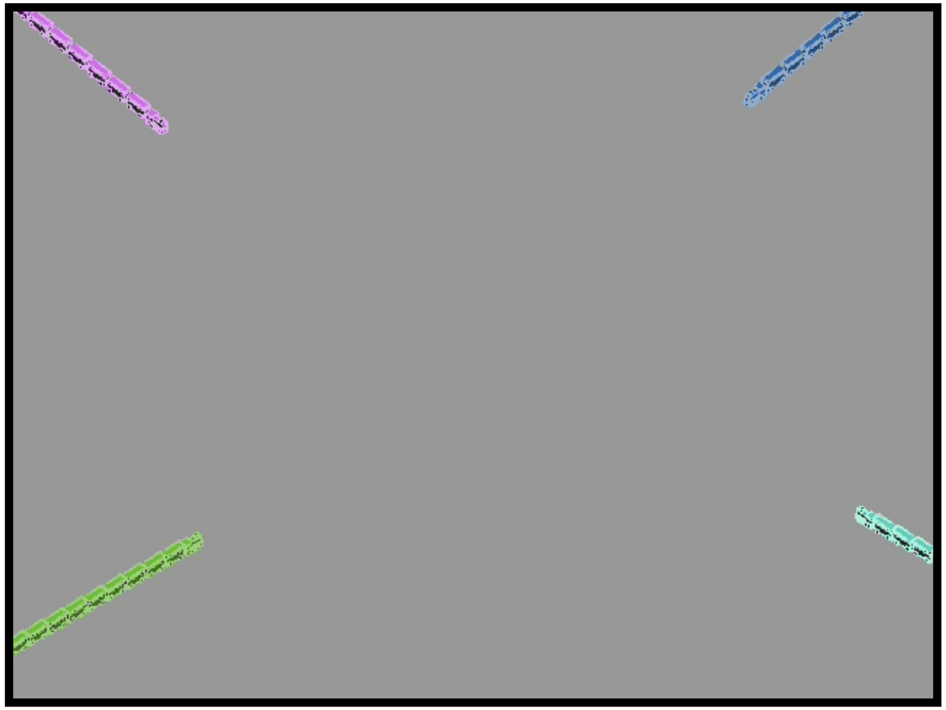
- Given a function *f(t)*, compute its *tight-fitting* AABB



- Easy way
  - Use interval arithmetic [Redon et al. 2004a,b]
  - Overshooting problem♪

Bounding Volume Culling♪





Interval Arithmetic♪

Taylor Models♪

Locomotion Benchmark♪

- **CCD performance**
  - **1.22 msec**
- Mannequin
  - 15 links, 20K tri
- Obstacles
  - 101K tri
- Locomotion SW
  - Footstep™

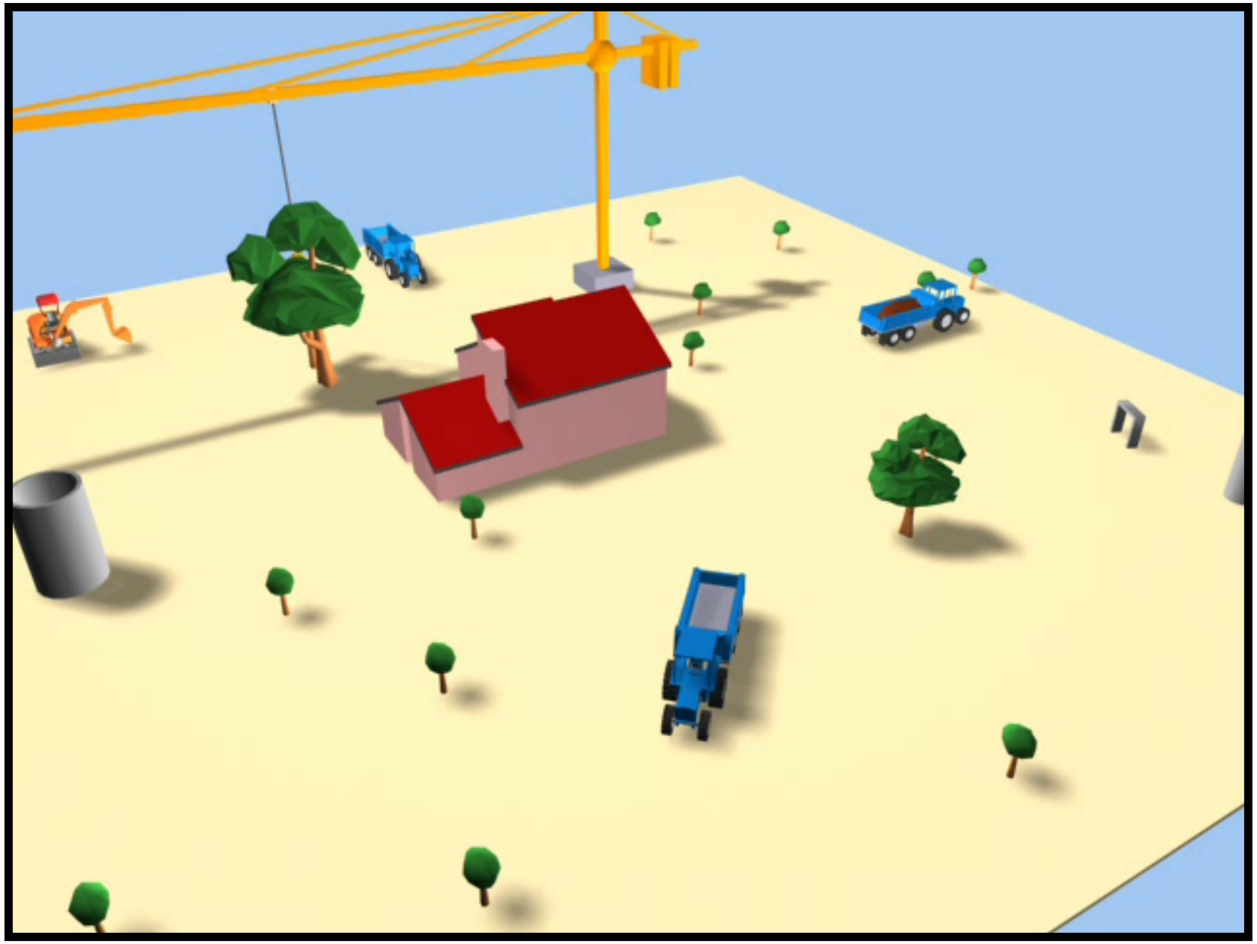
Exercise Benchmark♪

- Mannequin
  - 15 links, 20K tri angles
- **Self-CCD perfo rmance**
  - **0.38 msec♪**

Motion Planning Benchmark 1♪

- Excavator
  - 52 links, 19K tri
- Obstacles
  - 0.4M tri

- **CCD performance**
  - **100~700 msec**

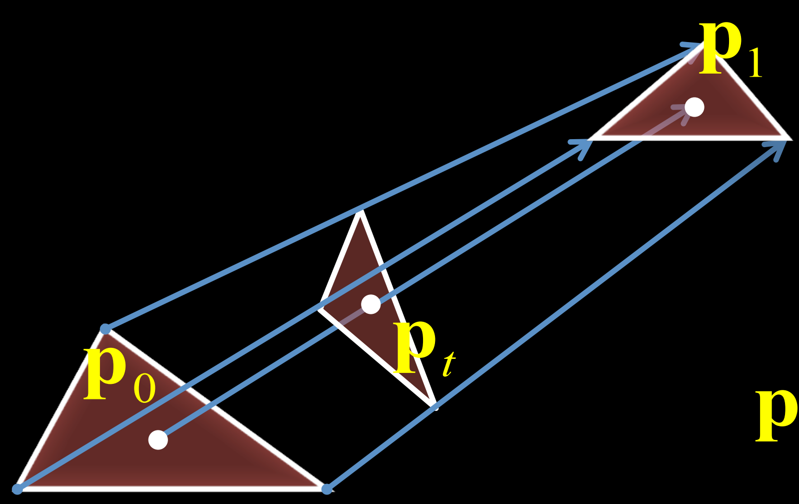


Motion Planning Benchmark 2♪

- Tower crane
  - 14 links, 1288 tri
- **CCD performance**
  - **5.66~15.1 msec**

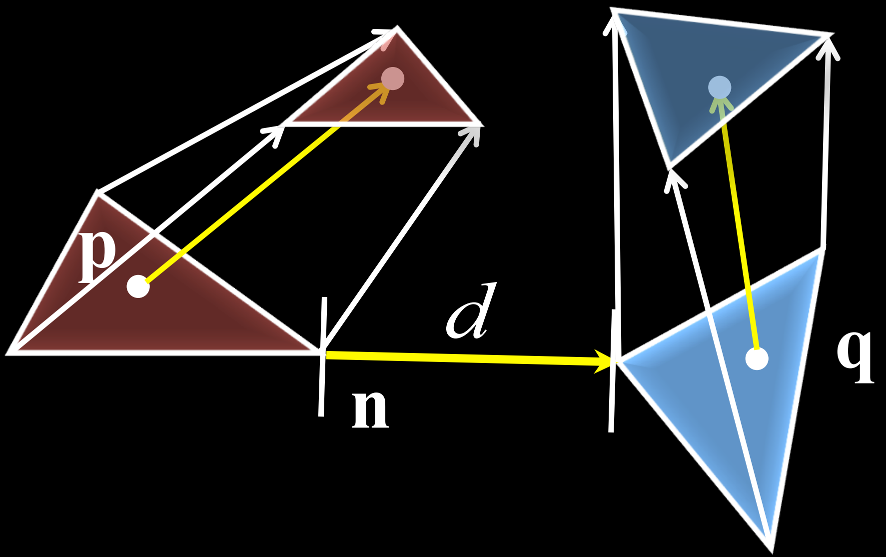
Extension to Deformable
[ICRA 2010]

- Point-wise linear interpolation



$$\mathbf{p}_t = (1-t)\mathbf{p}_0 + t\mathbf{p}_1$$

Extension to Deformable

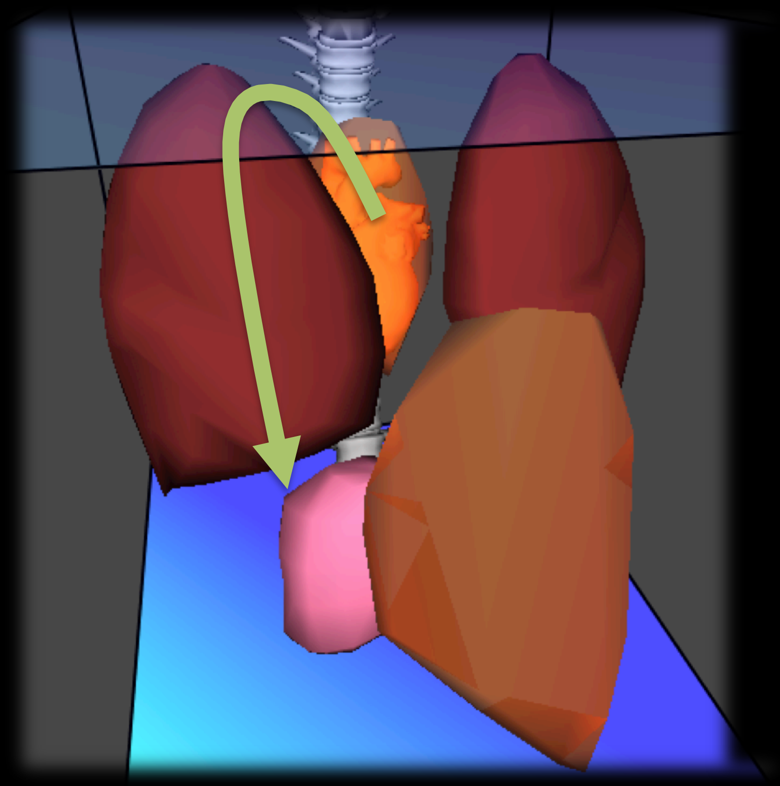


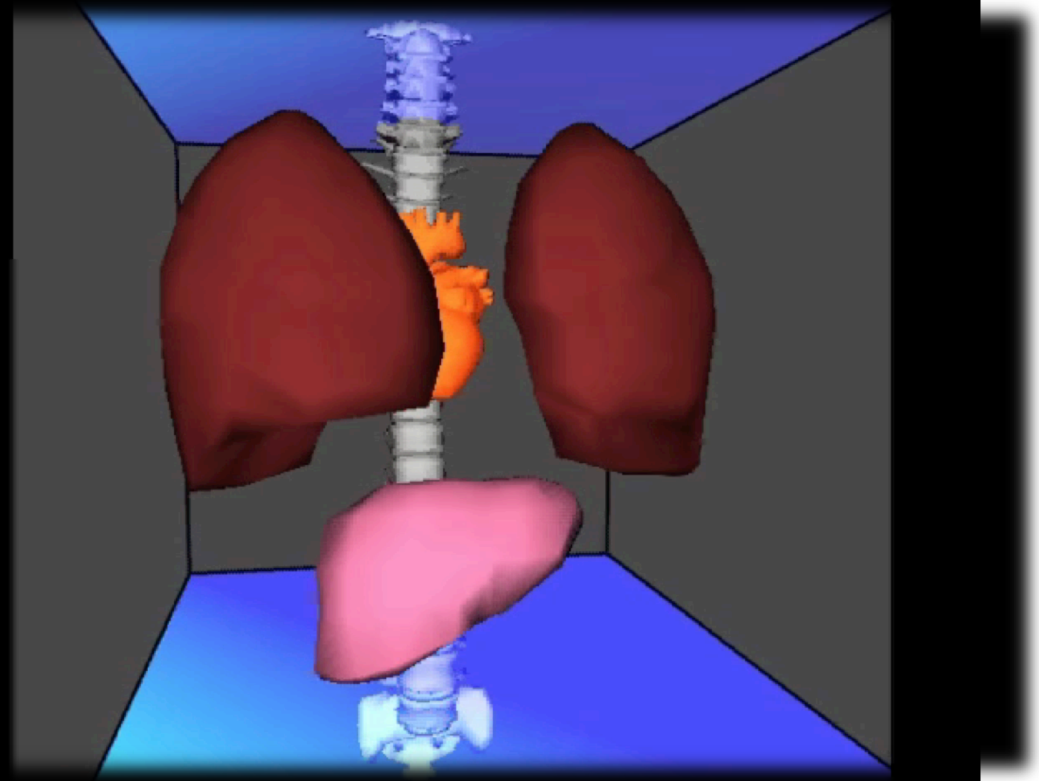
$$= \mu$$

$$\left(\max\left(\mu_{\mathbf{p}}\right)+\max\left(\mu_{\mathbf{q}}\right)\right)\cdot dt \le d \;\Longrightarrow\; dt \le \frac{d}{\mu}$$

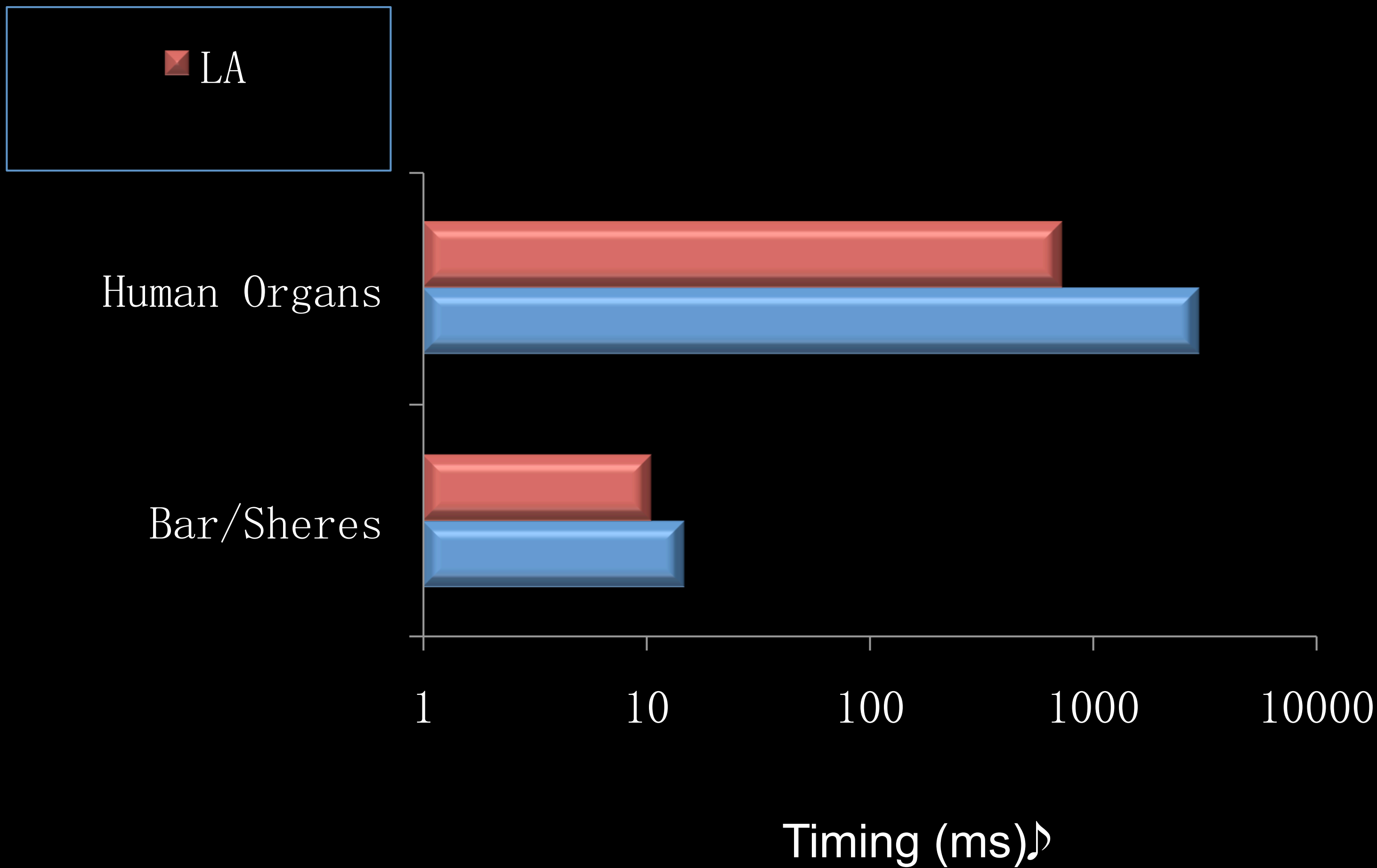
Deformable Motion Planning

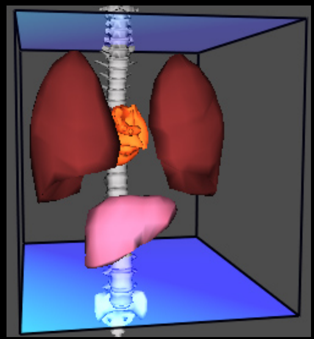
Human Organs (14K triangles) [Rodriguez 06]

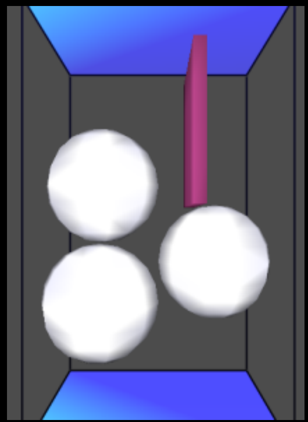




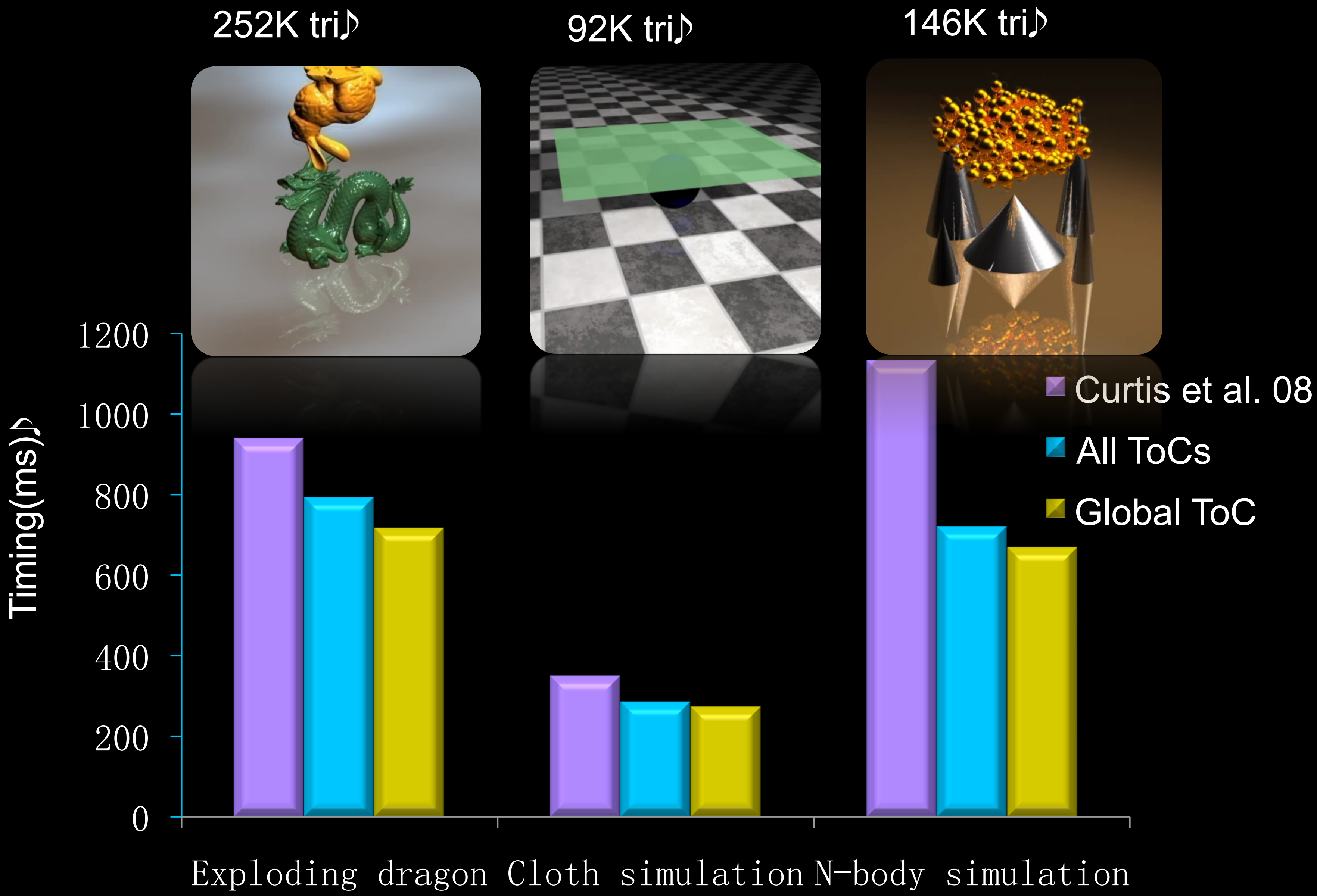
Deformable Motion Planning Benchmarks







UNC Dynamic Scene Benchmark



More on Deformable CCD

- Collision avoidance session
- Thursday, May 6th, 10:20 AM

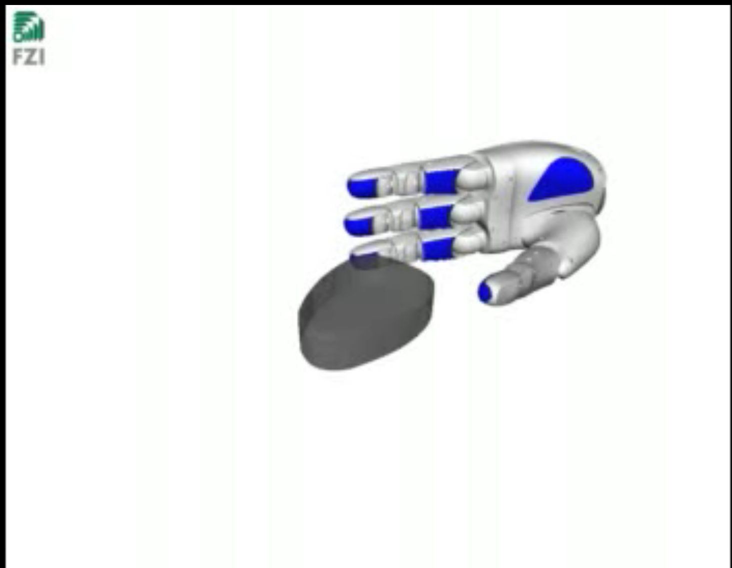
- Continuous collision detection for non-rigid contact computations using local advancement

Software Implementations

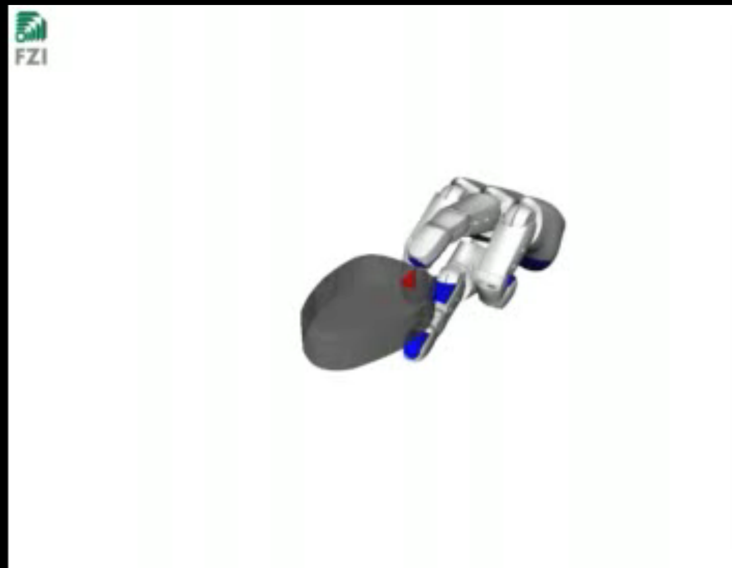
- Source codes are available
  - http://graphics.ewha.ac.kr/FAST
  - http://graphics.ewha.ac.kr/C2A
  - http://graphics.ewha.ac.kr/CATCH

Grasp Planning

- Decomposed into hand movement part and finger closing part



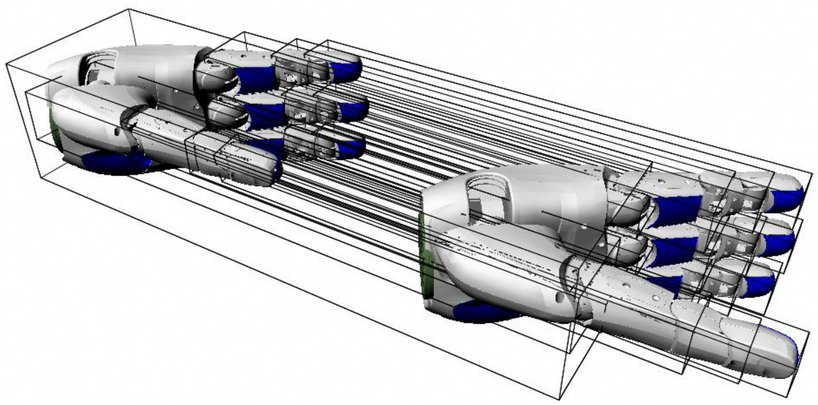
Hand Motion♪



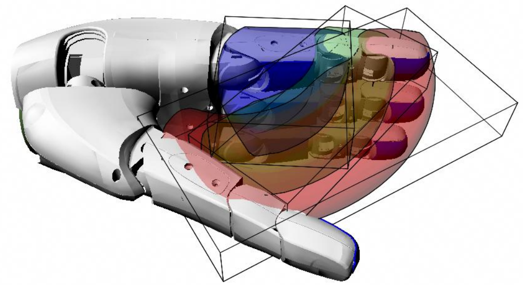
Finger Closing♪

Grasp Planning

- Extent of the finger links' motion are pre-computed and used to cull the objects that are far from the robotic hand



Hand Moving

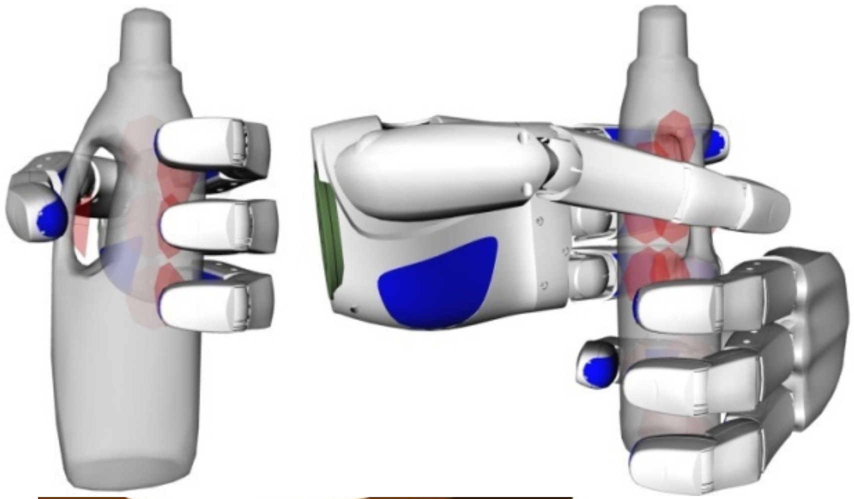


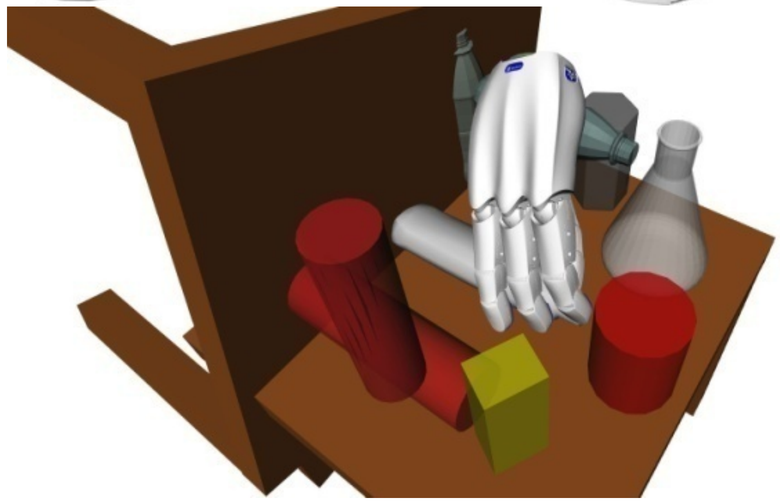
Finger Closing

Experimental Results

- Three settings
  - Standalone
  - Bimanual grasping
  - Grasping in complex e nvironment

- Based on GraspIt!





Experimental Results

- CATCH is 4 ~ 10 times faster than the extended PQP in GraspIt!
- At least 10 grasp candidates can be tested within one second

Experiment	T_{moving} msec	$T_{closing}$ msec	T_{ray} msec	N_1	N_2	T_{total}	Q_{best}
Standalone CATCH	4	14	2	528	102	32	0,164
Standalone PQP	20	101	2	528	152	77	0,162
Bimanual CATCH	41	49	2	460	10	25	0,198
Bimanual PQP	130	289	2	460	87	110	0,323
Environment CATCH	22	50	3	70	3	14	0,058

Real Robot Execution

Conclusions

- Fast and reliable CCD methods
- Application to forward grasping

- Future work
  - Deformable grasping
  - Fast (real-time?) grasping using graphics hard ware

Acknowledgements

- Xinyu Zhang, Minkyoung Lee (Ewha)
- Stephane Redon (INRIA)

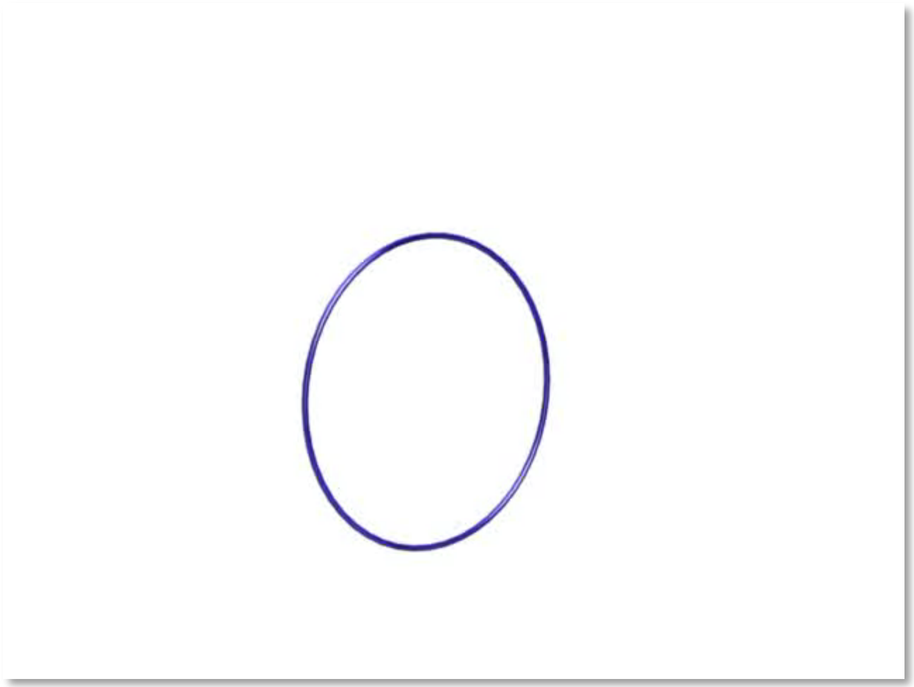
- MKE/KEIT in Korea (IT core research)

Thank you for listening!

http://graphics.ewha.ac.kr

Rigid Body Dynamics

1996 FPS



529 FPS

Articulated Body Dynamics Benchmark♪

- Four trains
  - 10 links, 23K tri (each)
- **CCD performance**
  - **535 msec♪**