Eurographics 2012
Cagliari, Italy
May 13-18
33rd ANNUAL CONFERENCE OF THE EUROPEAN ASSOCIATION FOR COMPUTER GRAPHICS

Real-time Continuous Collision Detection and Penetration Depth Computation

Young J. Kim

http://graphics.ewha.ac.kr
Ewha Womans University

CRS4
Eurographics
EUROPEAN ASSOCIATION FOR COMPUTER GRAPHICS

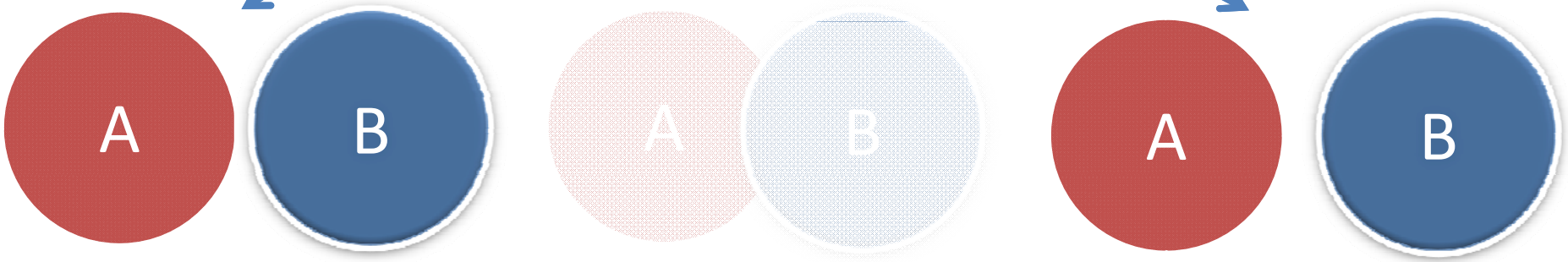
Autonomous Ice Serving Robot

(with Zhixing Xue at FZI)

Non-penetration Constraint

- **No overlap between geometric objects**

$$\text{interior}(A) \cap B \neq \phi$$



- **Crucial for mimicking the physical presence**

Earlier Research

- **Focused on checking for whether there is any overlap between A and B, *fixed in space***
  - Tons of papers published in the area of *collision detection*
  - Well-studied and matured technology
- **Not clear how to resolve such overlap**

Recent Research Trends

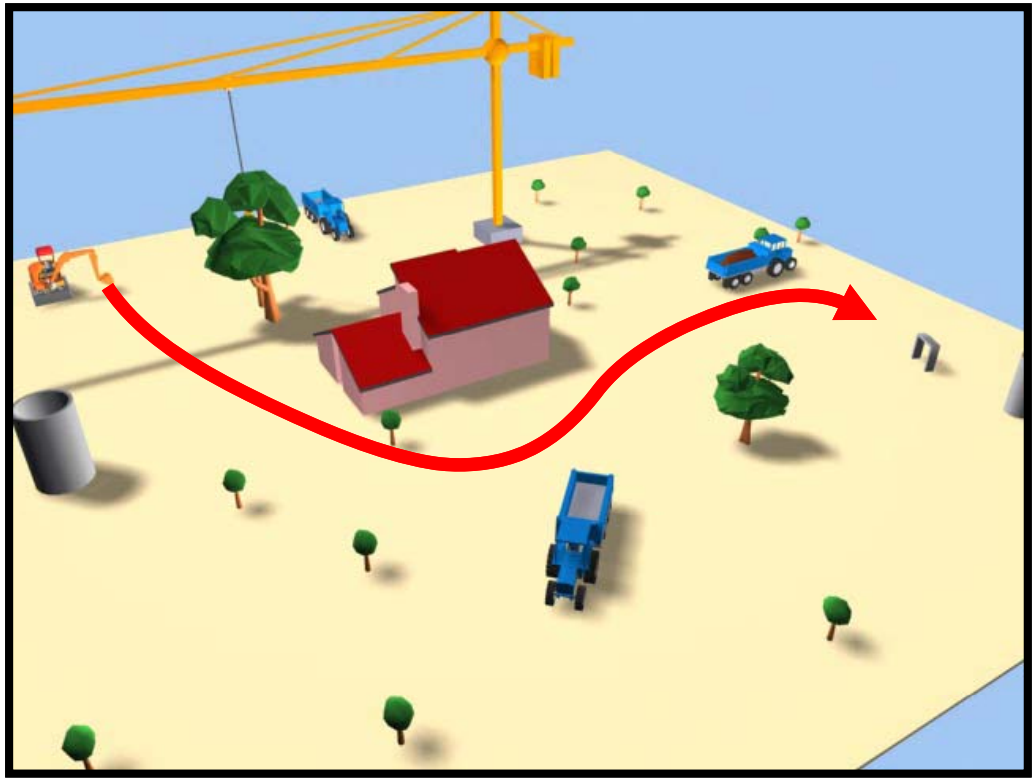
- **Avoid inter-penetration**
  - Continuous collision detection (CCD)
- **Allow inter-penetration but backtrack**
  - Penetration depth (PD)

Applications of Continuous CD

- **Rigid body dynamics**
  - Find the time of contact (ToC) to apply forces

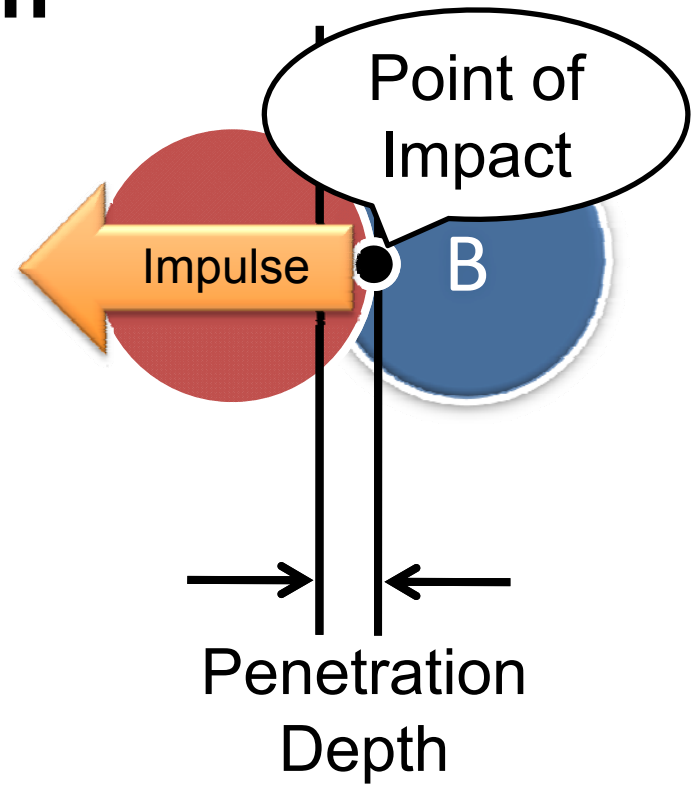
Applications of Continuous CD

- **Motion planning**
  - Check whether a path is collision-free



Applications of Penetration Depth

- **Dynamics simulation**
  - Penalty-based
  - Impulse-based



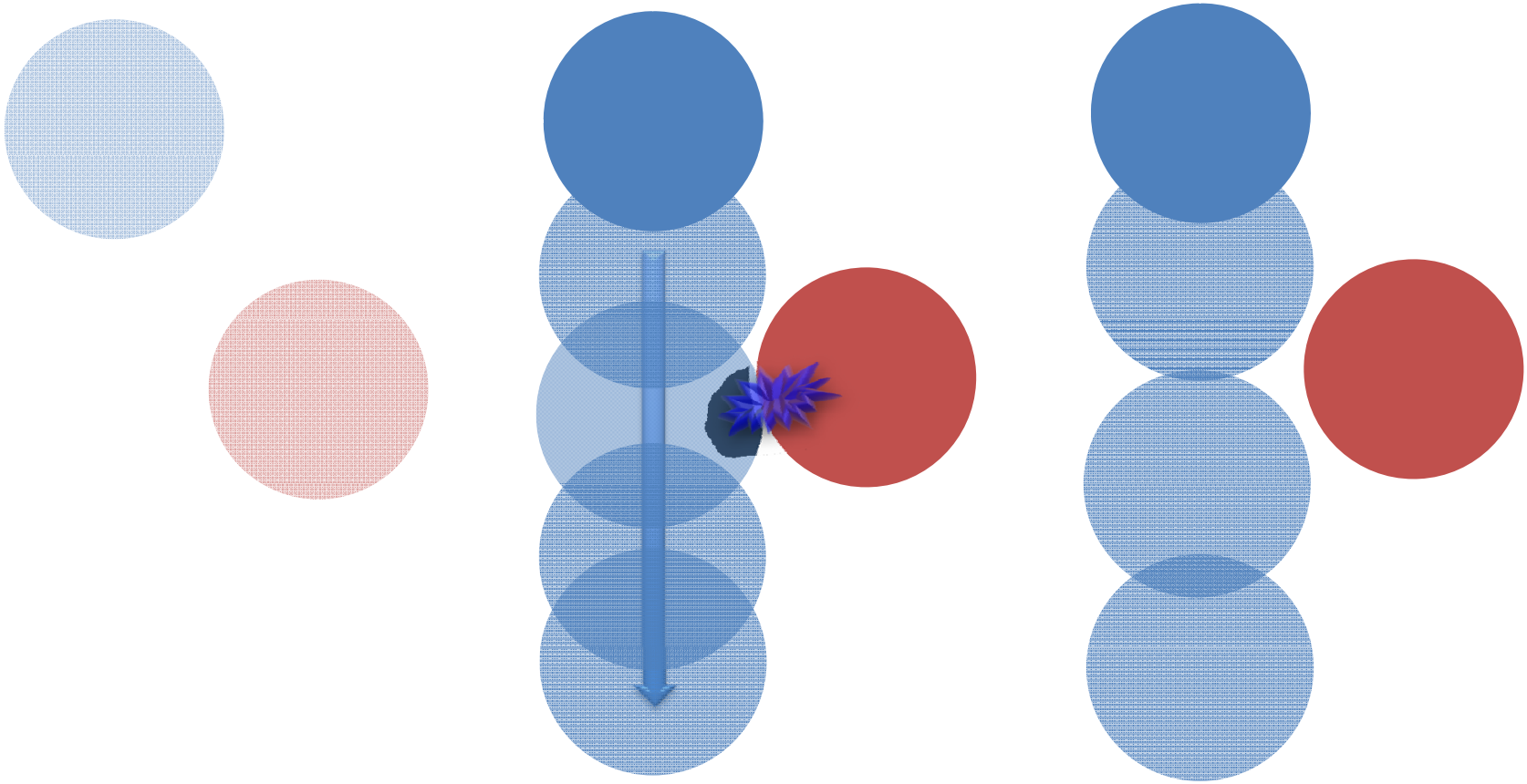
Goal

- **Recent research results**
  - CCD (rigid, polygon-soups, articulated)
  - PD (pointwise, translational)

- **Applications**
  - Real-time rigid body dynamics
  - Robotic grasping
  - CAD disassembly

CONTINUOUS COLLISION DETECTION

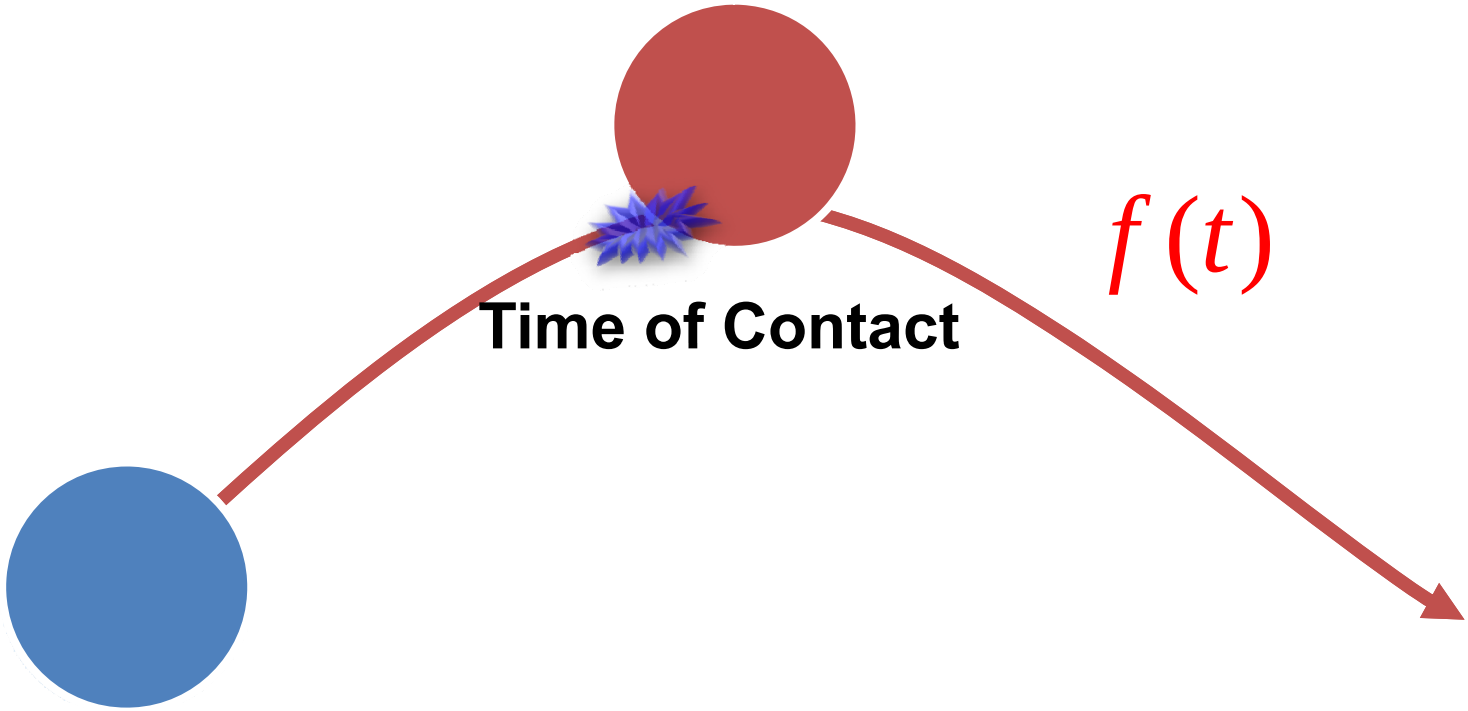
Discrete Collision Detection



Collision Missing

Continuous Collision Detection

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

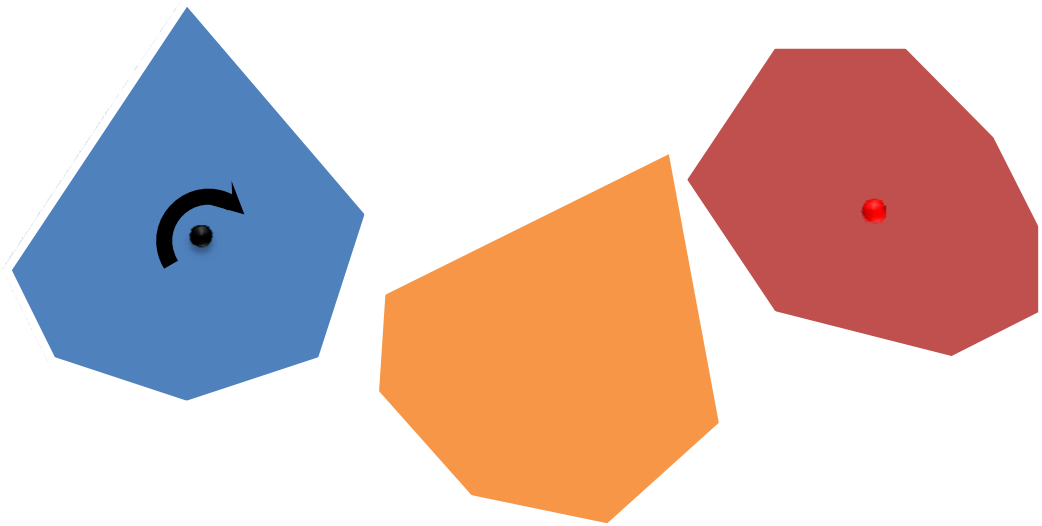


Previous Work on CCD

- Algebraic solution -[Canny86], [Redon00], [Kim03], [Choi06]
- Swept volume -[Abdel-Malek02],[Hubbard93], [Redon04a,b]
- Bisection -[Redon02], [Schwarzer02]
- Kinetic data structures -[Kim98], [Kirkpatrick00], [Agarwal01]
- Minkowski sum -[Bergen04]
- Conservative advancement
  - [Mirtich96], [Mirtich00],[Coumans 2006],[Zhang06], [Tang09]

Conservative Advancement (CA)

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

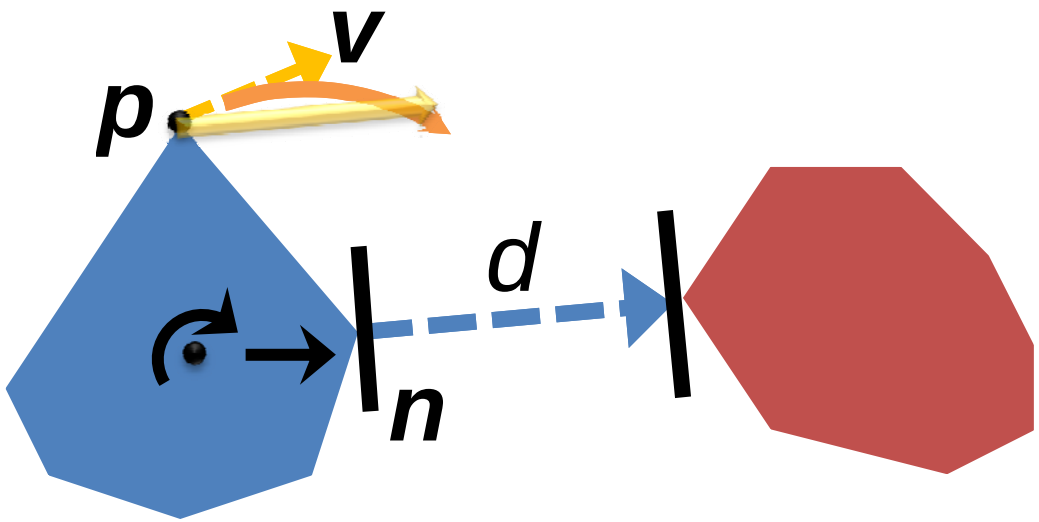


Conservative Advancement (CA)

1. Find a step size Δt_i to conservatively advance the object until collision occurs
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

$$\int_0^{\Delta t} \left| \mathbf{v}(t) \cdot \mathbf{n}(t) \right| dt \le \int_0^{\Delta t} \max(|\mathbf{v}(t)|, |\mathbf{n}(t)|) \, dt = \mu$$

Calculating Δt in CA

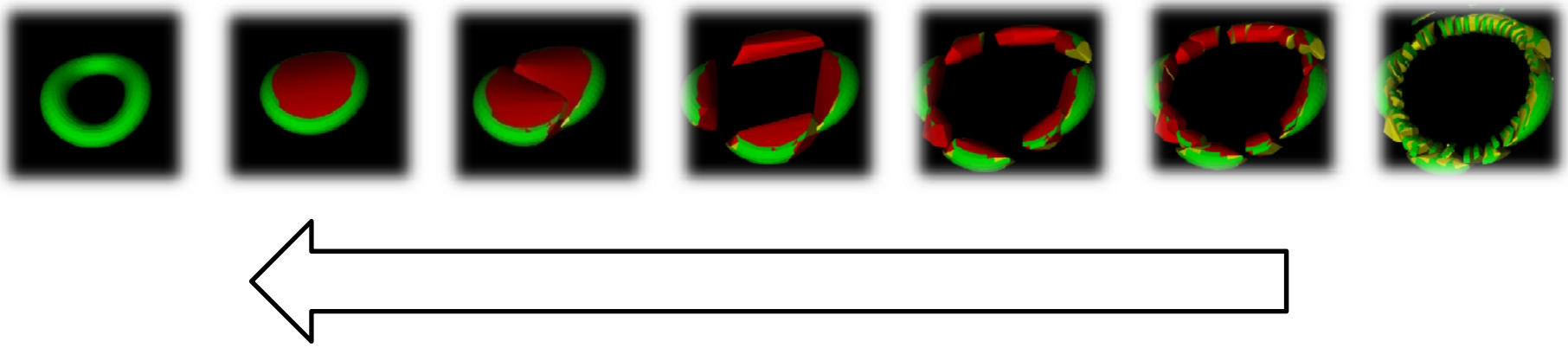
$$\mu \Delta t \leq d \qquad \therefore \Delta t \leq \frac{d}{\mu} \quad = \mu$$

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

Extension to Non-convex

[Zhang et. al PG 06]

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



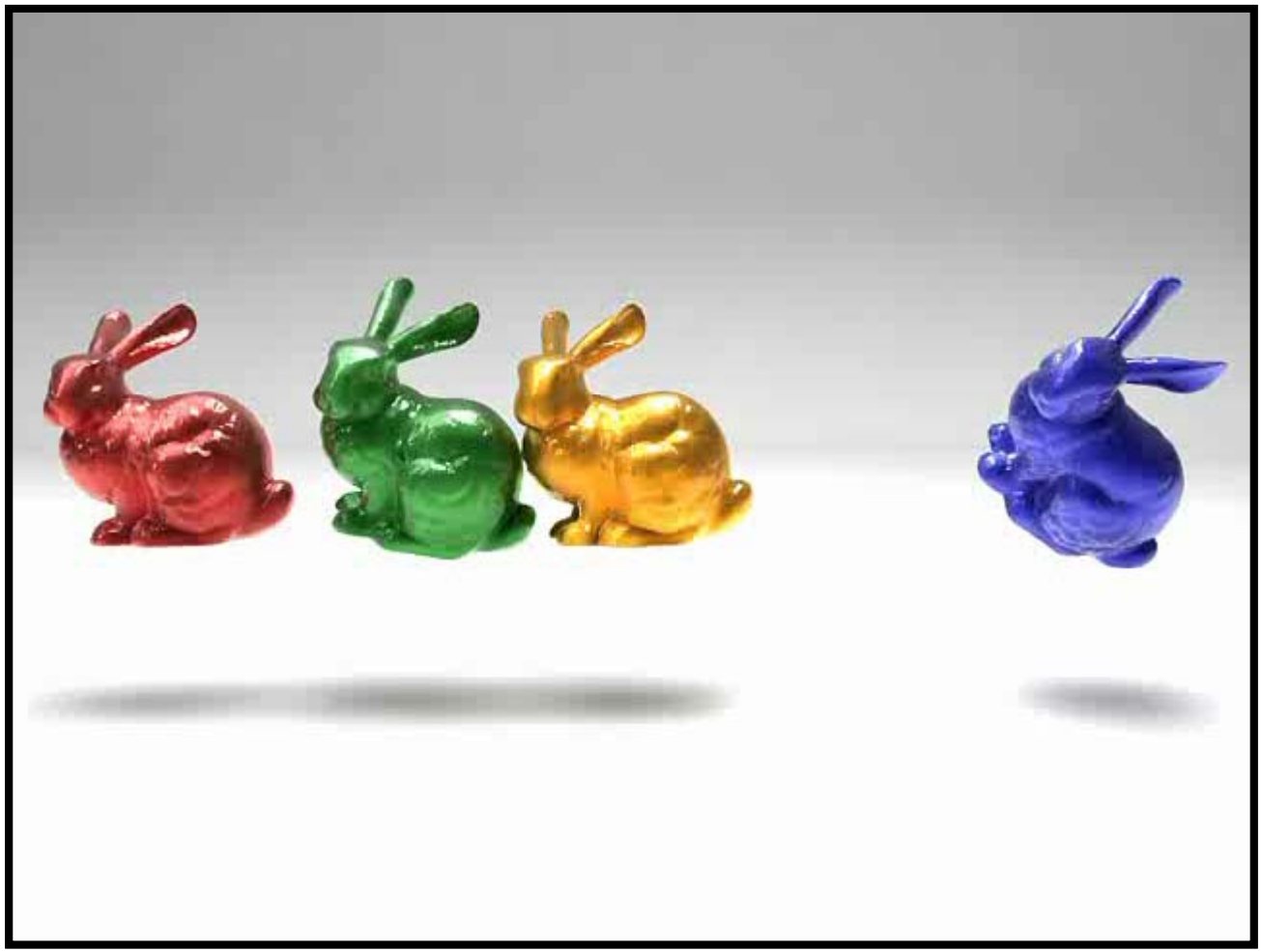
Santa vs. Thin Board

37K triangles

51,546 FPS

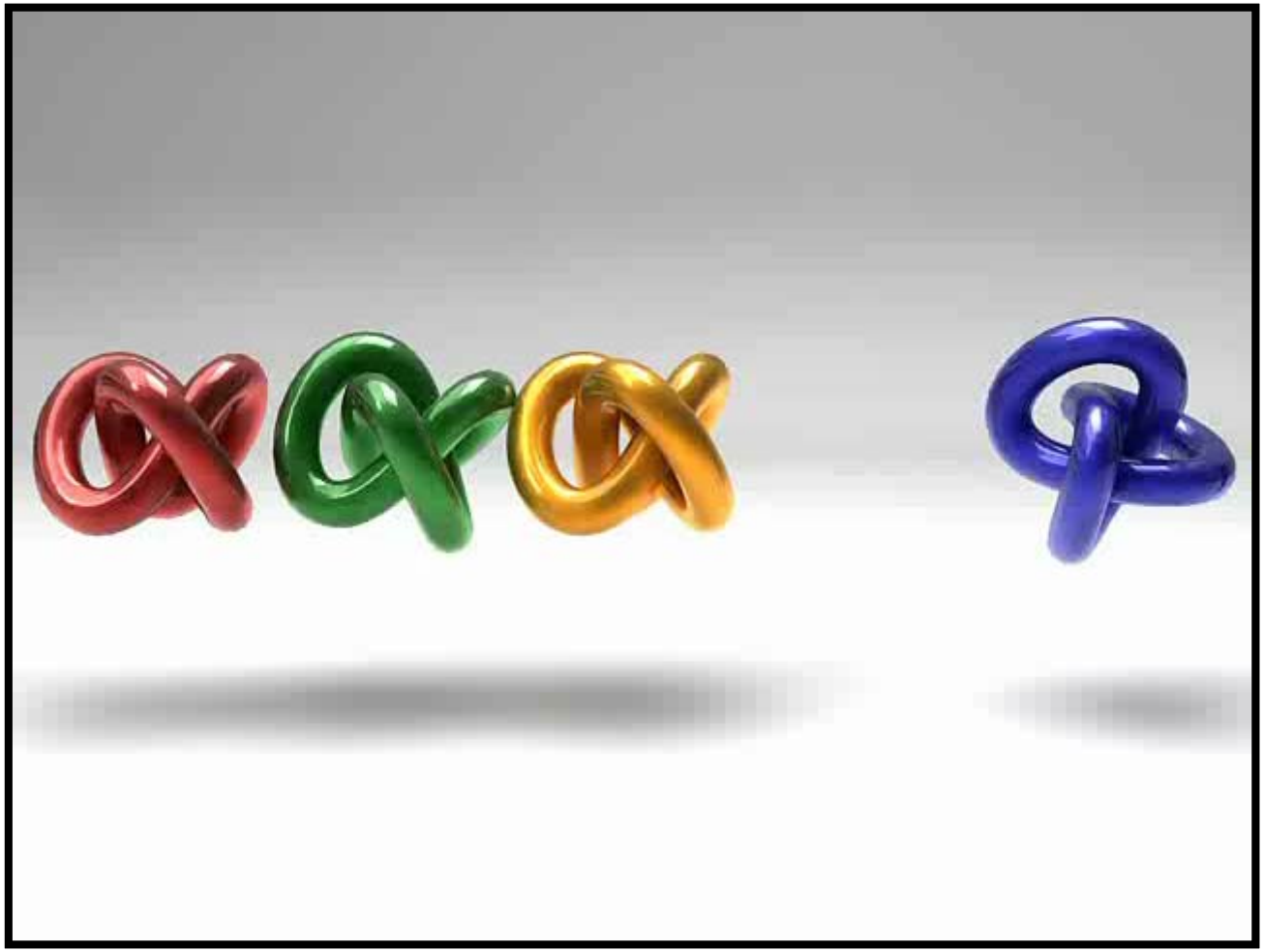
# of iterations
3.68

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

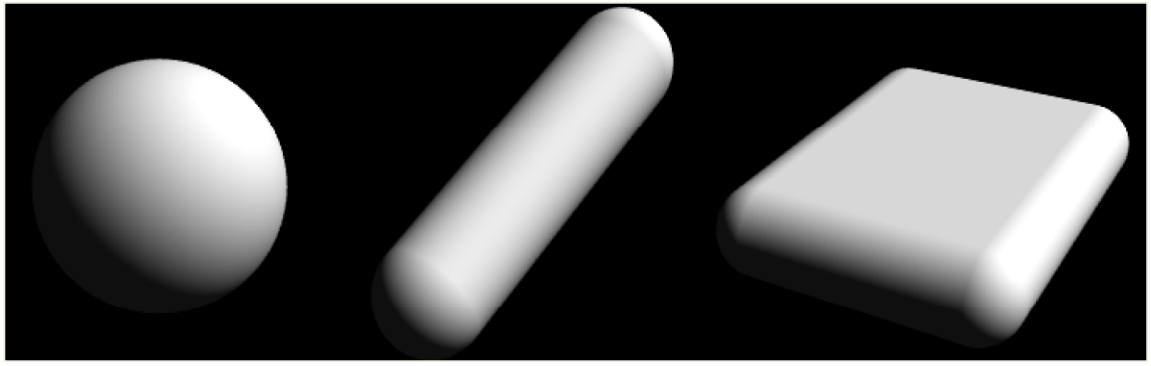
# of iterations 4.46

Extension to Polygon-Soups
[Tang et. al IEEE ICRA 09]

- **Construct the bounding volume hierarchy of polygons**
- **Motion bound calculation** $\Delta t \leq \frac{d}{\mu}$
  - Bounding volume
  - Triangles

Extension to Polygon-Soups
[Tang et. al IEEE ICRA 09]

- **We use swept sphere volumes**



[Larsen et. al IEEE ICRA 1999]

Motion Bound Calculation

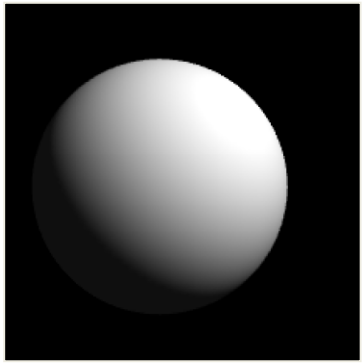
- **Motion bound of SSV (e.g. PSS)**

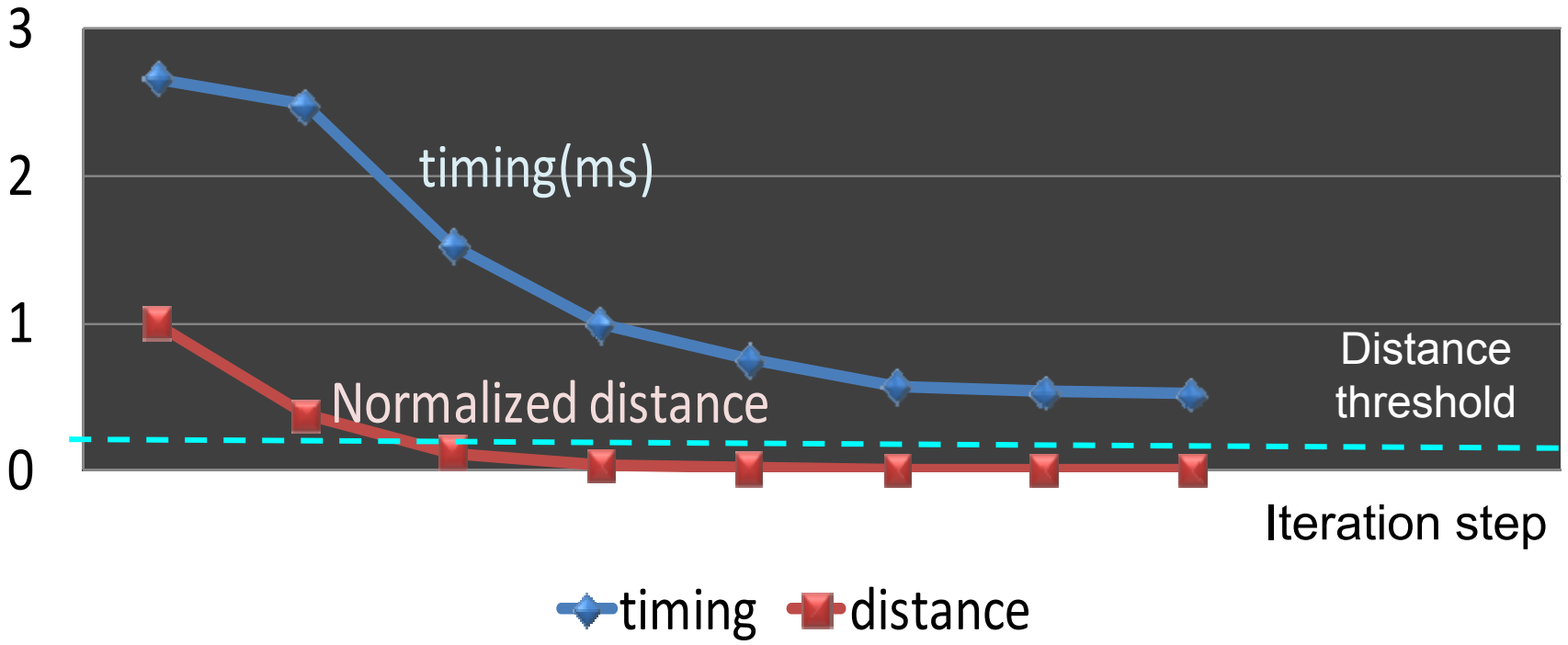
$$\mu \leq \|\boldsymbol{\omega} \times \mathbf{n}\| \left(\left\| \mathbf{c}_1^{\perp} \right\| + \mathbf{r} \right)$$

$\boldsymbol{\omega}$: rotational velocity

$\mathbf{n}$: closest direction

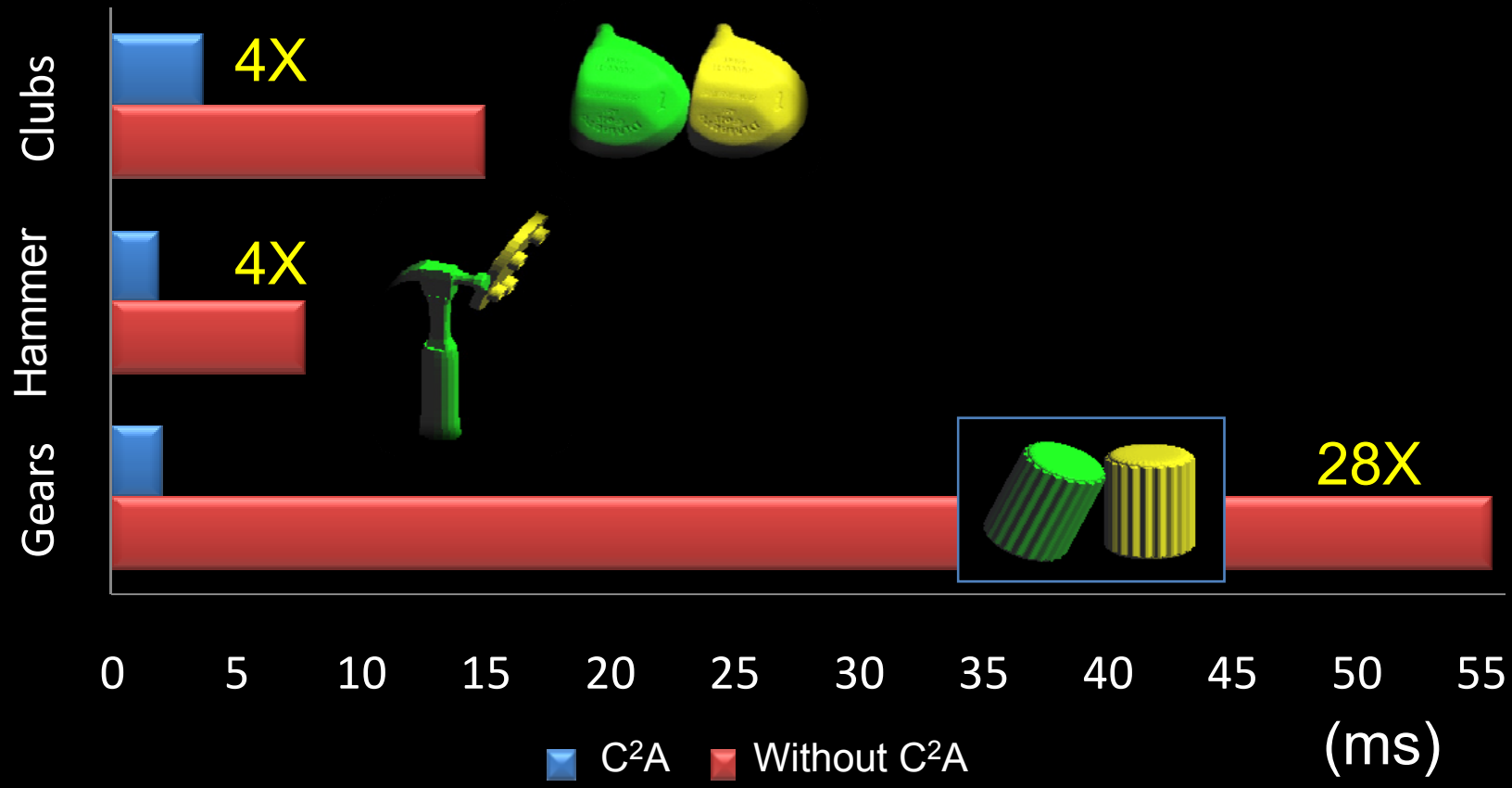
$\mathbf{r}$: radius of PSS



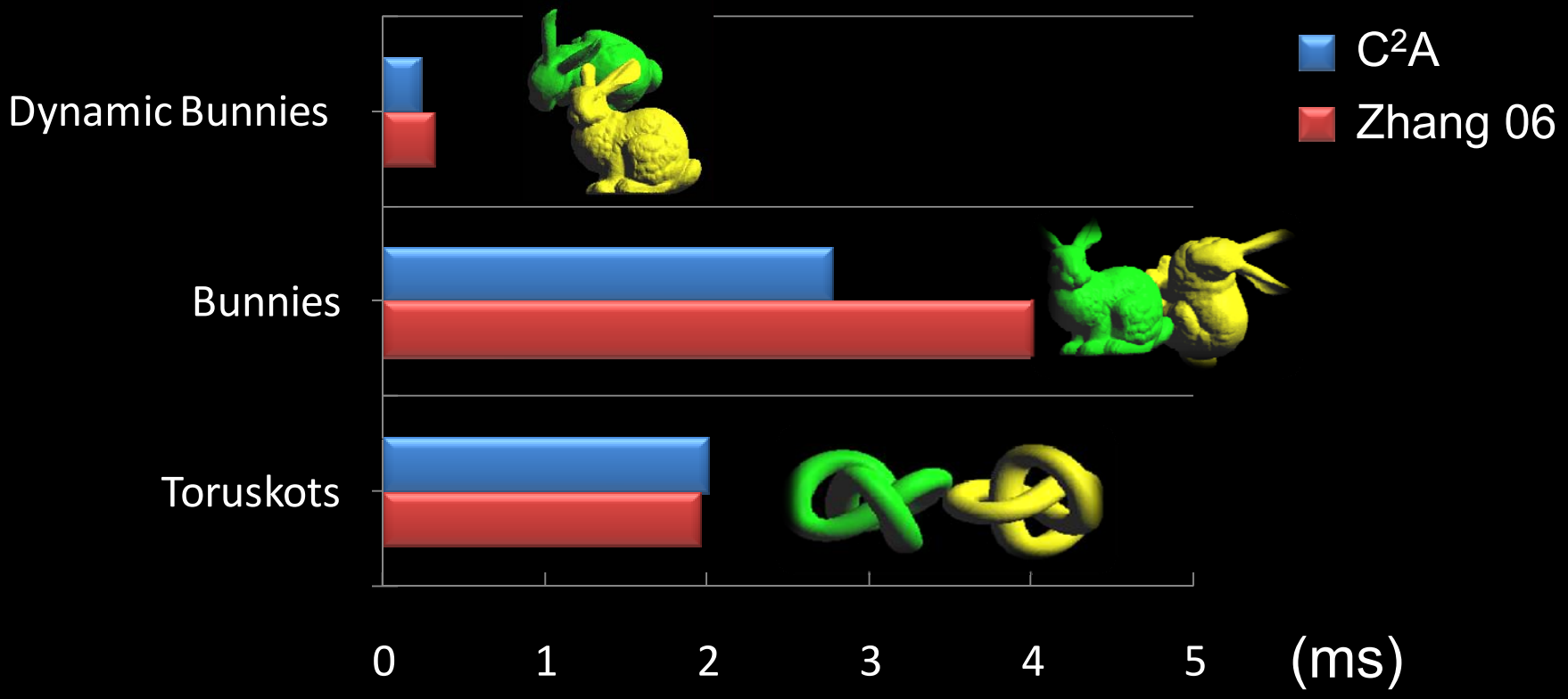


1. Compute approximate distance in the beginning
2. Compute exact distance toward the end

Results - Timings



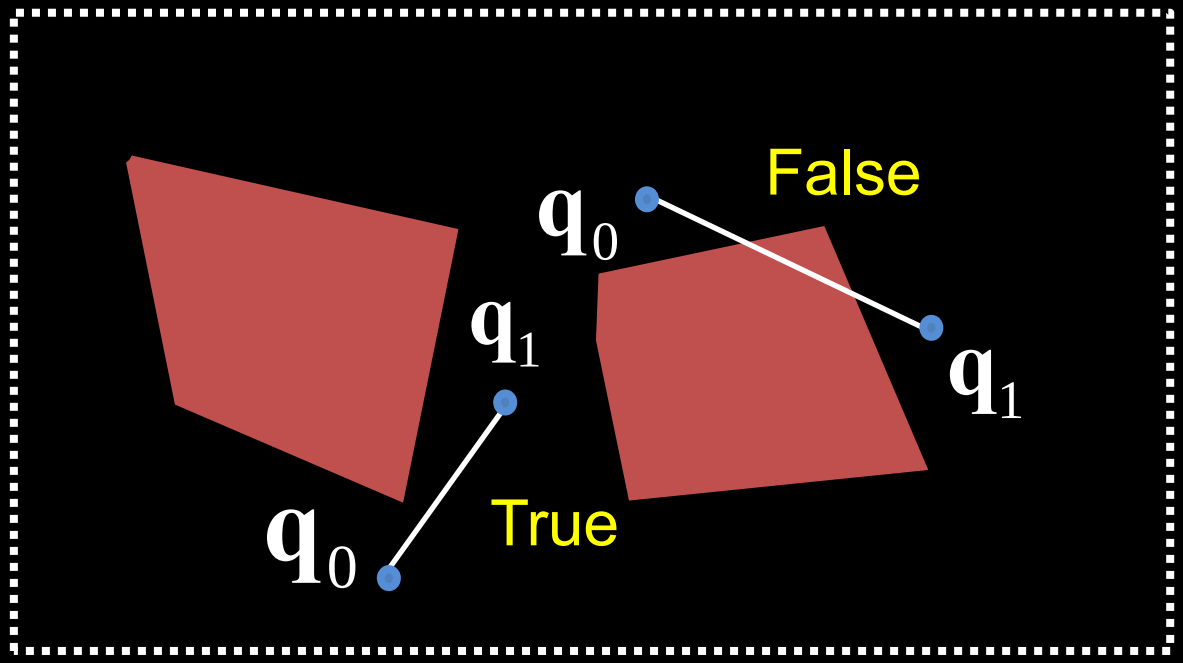
Comparisons against [Zhang 06]



- [Zhang 06] can handle only manifold surfaces

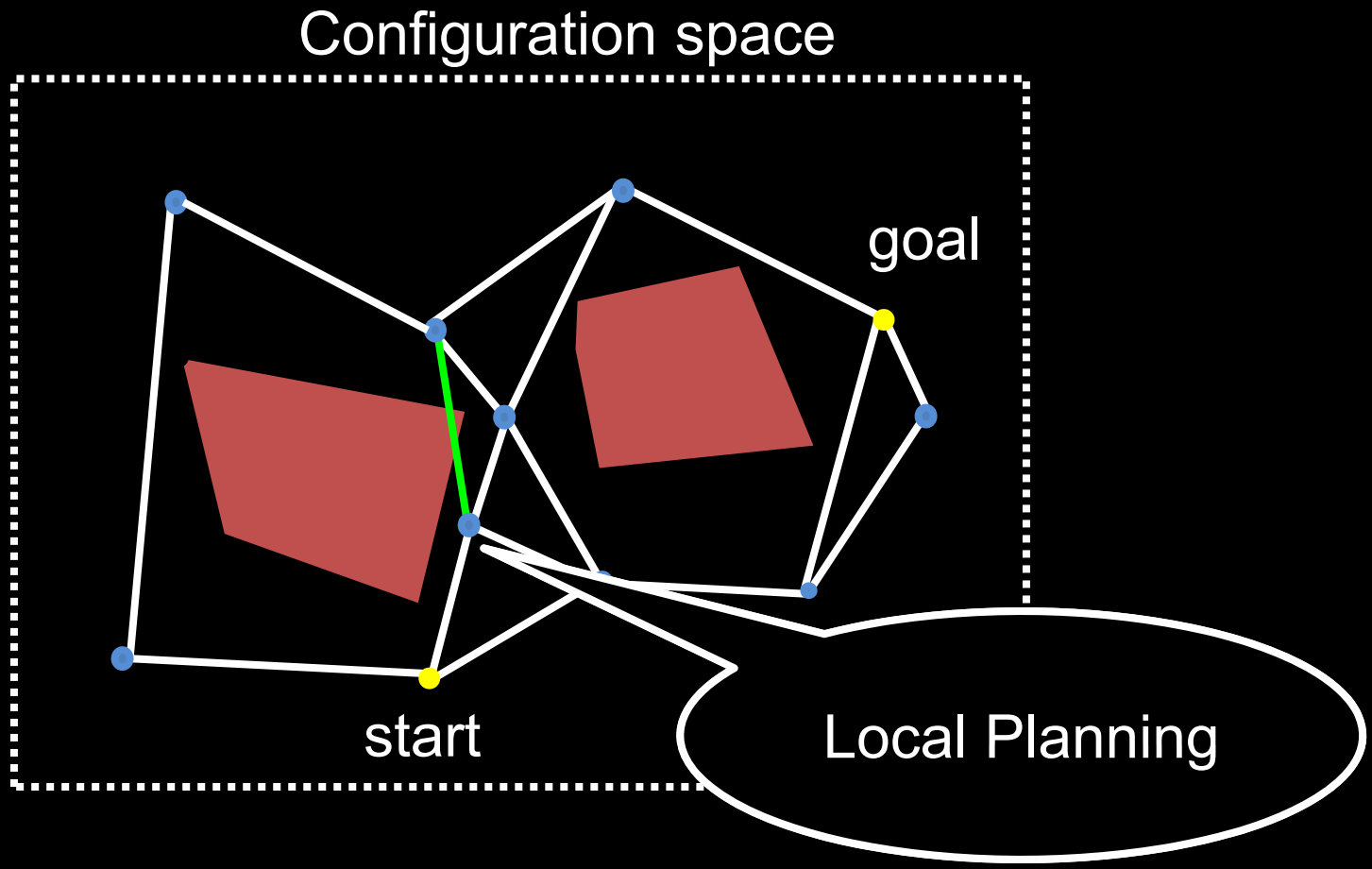
Boolean CCD [Tang et al. 2010]

- Check whether $\forall q(t) \in \text{Free Space}$, $t \in [0,1]$ or not, where $q(t)$ is the configuration of a robot at t



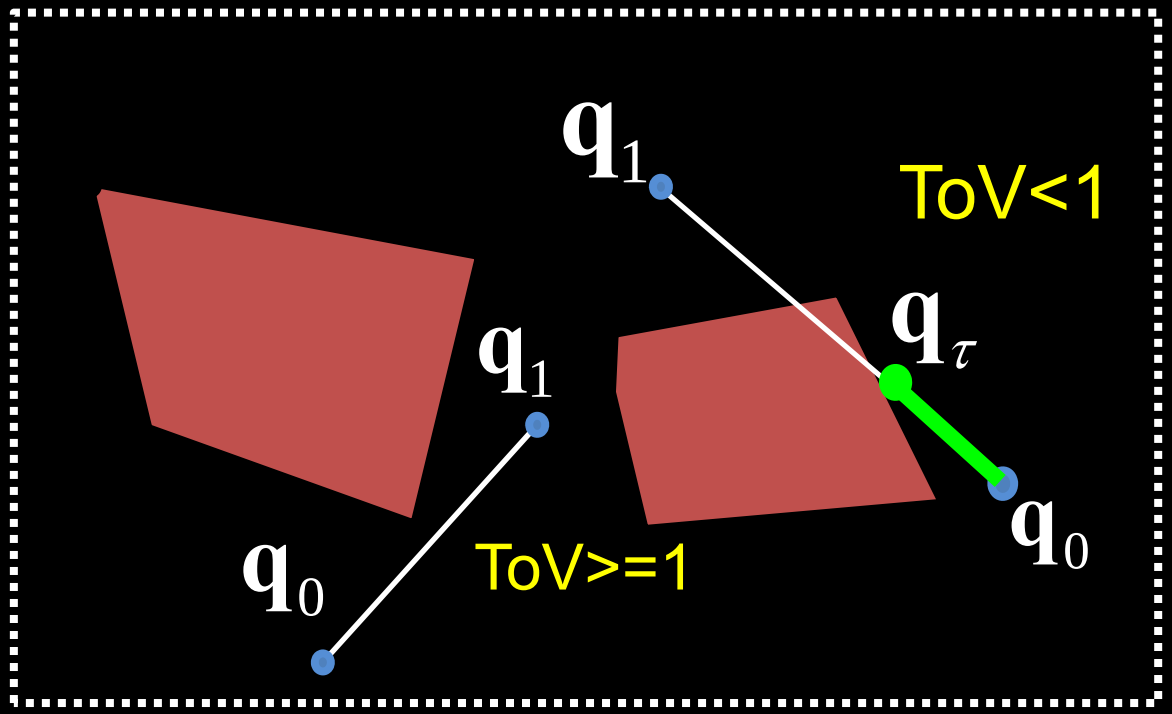
Probabilistic Roadmaps

Construction phase and query phase



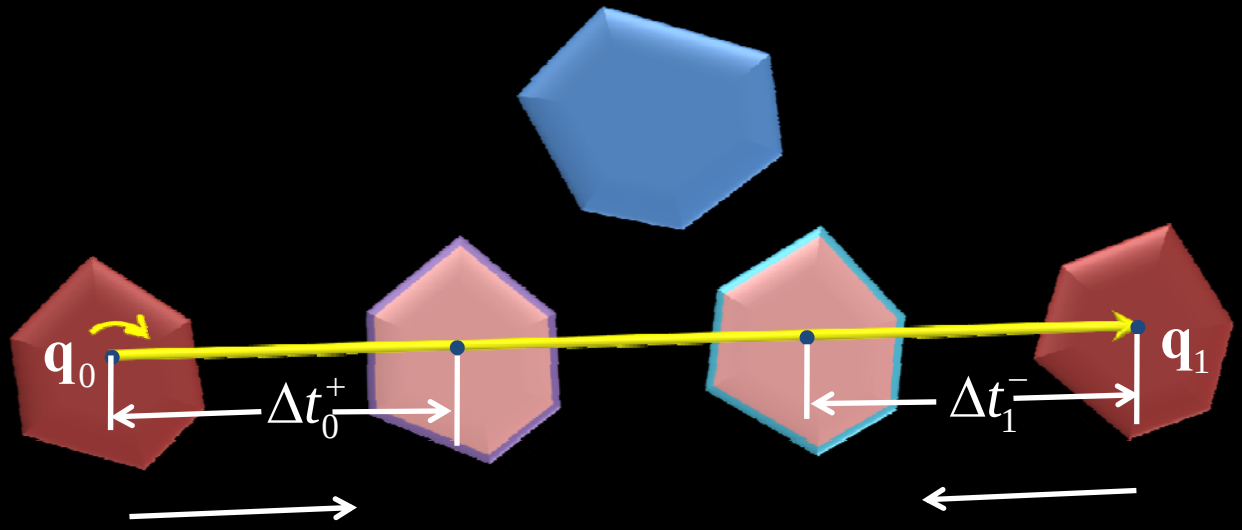
Boolean CCD

- Find the time of violation (ToV)
- If ToV>=1, valid path; otherwise, not valid



Boolean CCD

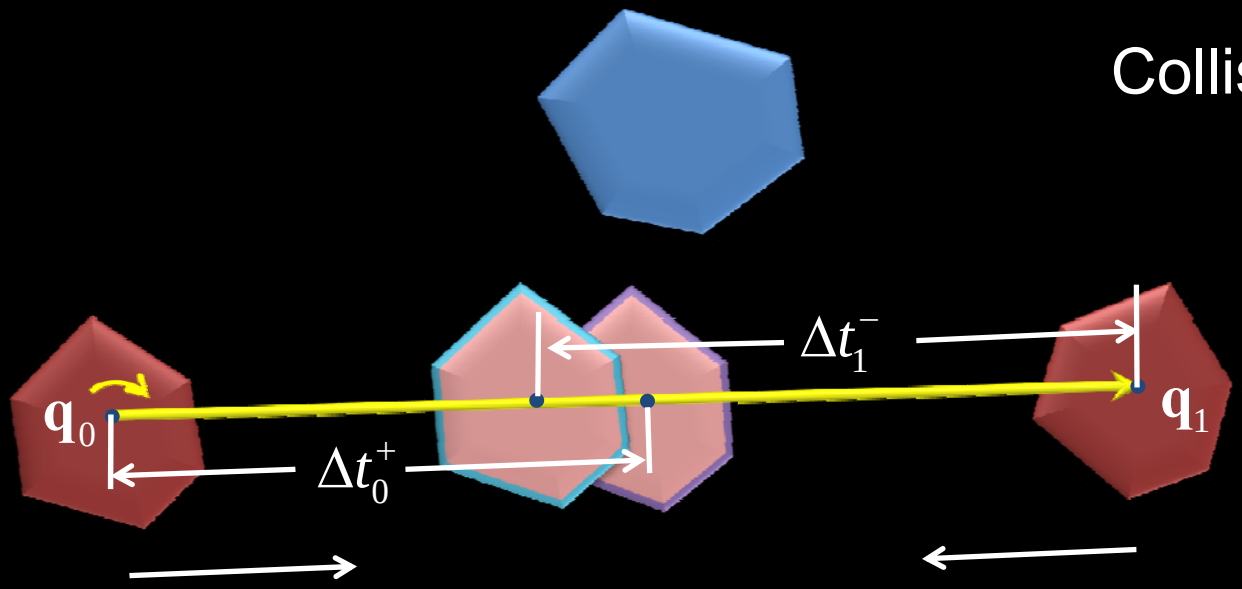
- Dual advancements from the both end-configurations



Boolean CCD

$$\Delta t_0^{+} + \Delta t_1^{-} \geq 1$$

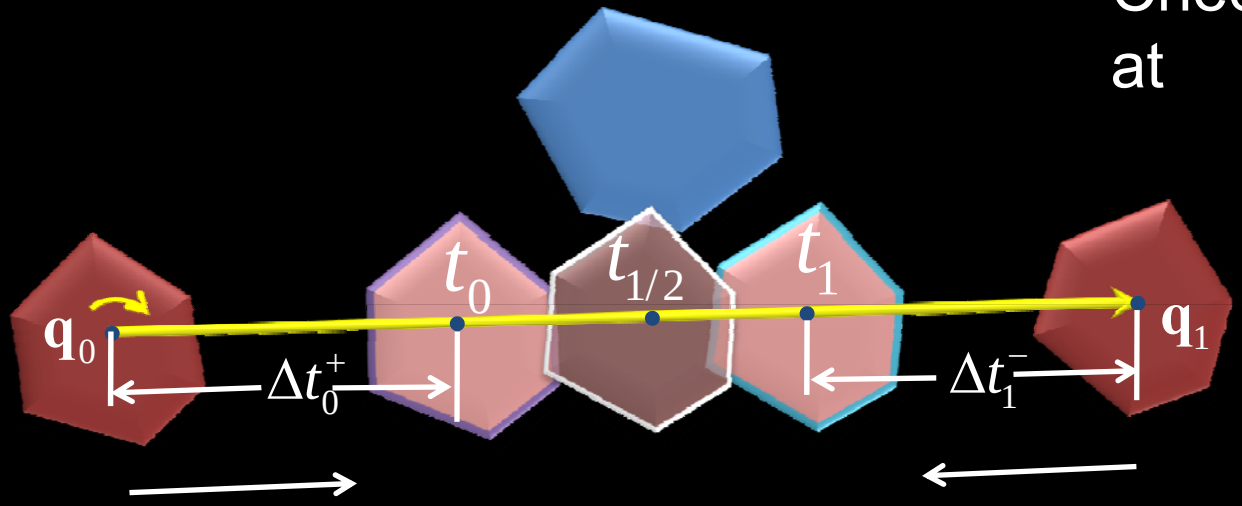
Collision-free



Boolean CCD

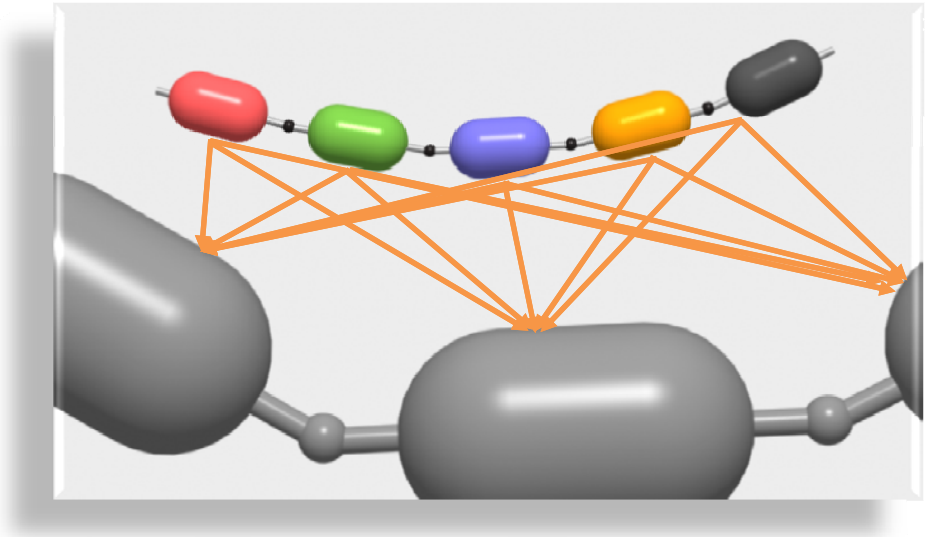
$$\Delta t_0^{+} + \Delta t_1^{-} < 1$$

Check collision at $t_{1/2} = \frac{t_0 + t_1}{2}$



Extension to Articulated Models
[Zhang et. al SIGGRAPH 07]

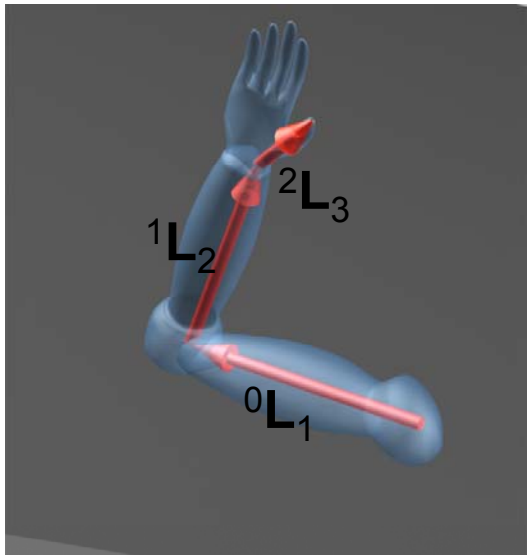
- Treat each link as a rigid body
- Apply CA to each link independently
- Taking the minimum of CA results

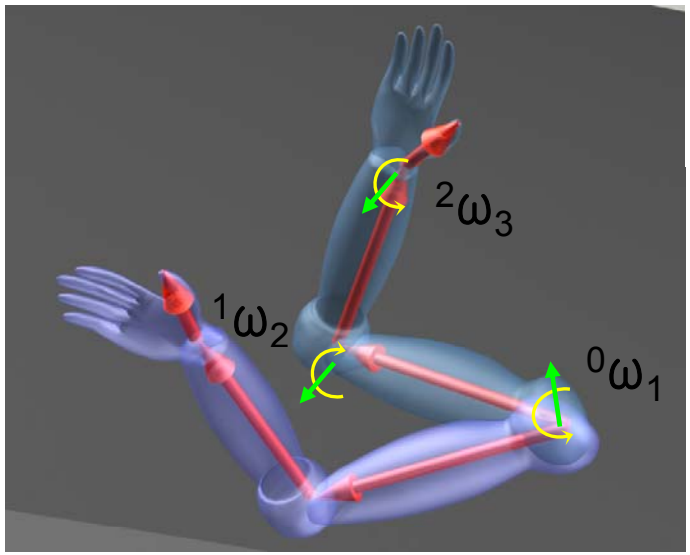


Motion Bound Calculation $\Delta t \leq \frac{d}{\mu}$

μ

$$\leq \left| {}^{0}\boldsymbol{\upsilon}_1 \cdot \mathbf{n} \right| + \left| \mathbf{n} \times {}^{0}\boldsymbol{\omega}_1 \right| \left| {}^{0}\mathbf{L}_1 \right|_{\mu} + \sum_{j=2}^{i-1} \left(\left| {}^{j-1}\boldsymbol{\upsilon}_j \right| + \left(\left| \mathbf{n} \times {}^{0}\boldsymbol{\omega}_1 \right| + \sum_{k=2}^{j} \left| {}^{k-1}\boldsymbol{\omega}_k \right| \right) \left| {}^{j-1}\mathbf{L}_j \right|_{\mu} \right)$$





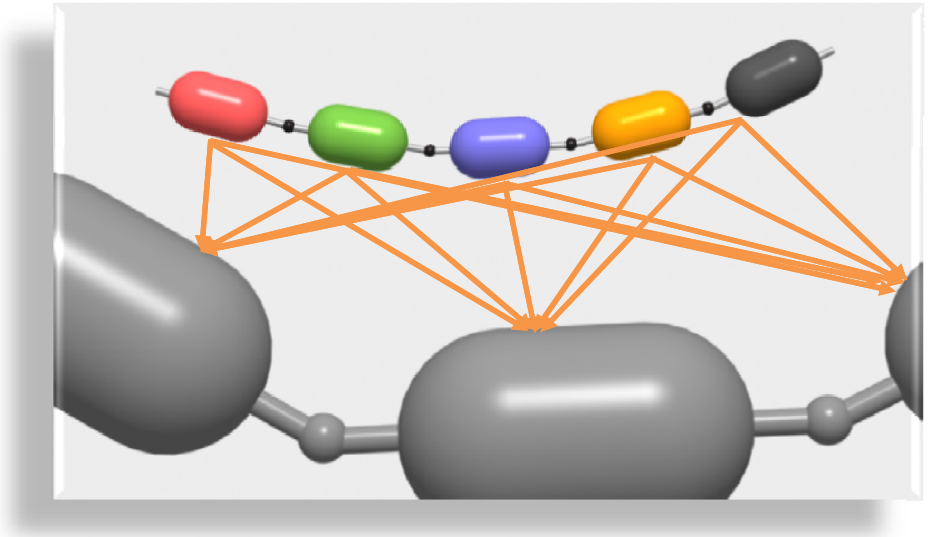
$^{0}v_i$: velocity of link i

$^{i-1}\omega_i$: rotational velocity of link i w.r.t. link i-1

$^{i-1}L_i$: difference vector btwn the links

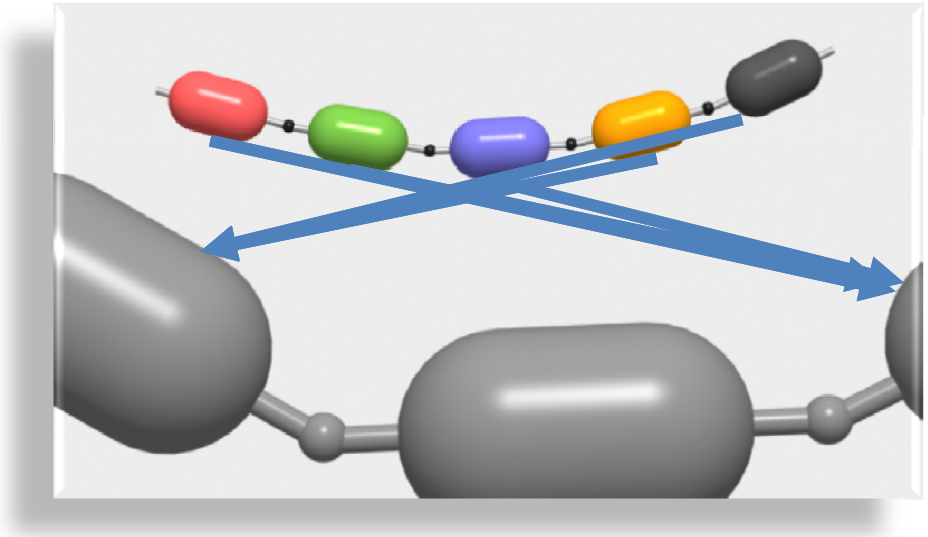
Problems in Straightforward CA

- Problems
  - $O(n^2)$ checking between individual links



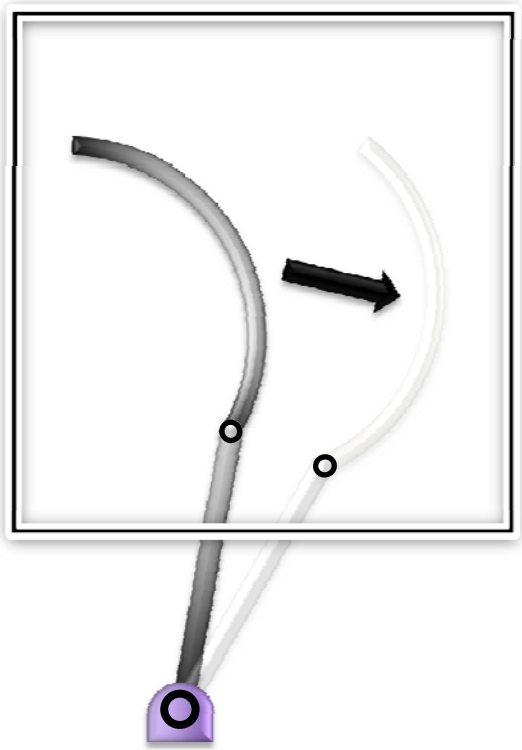
Spatial Culling

- Cull the link pairs that are far apart
- Use bounding volume-based collision-culling

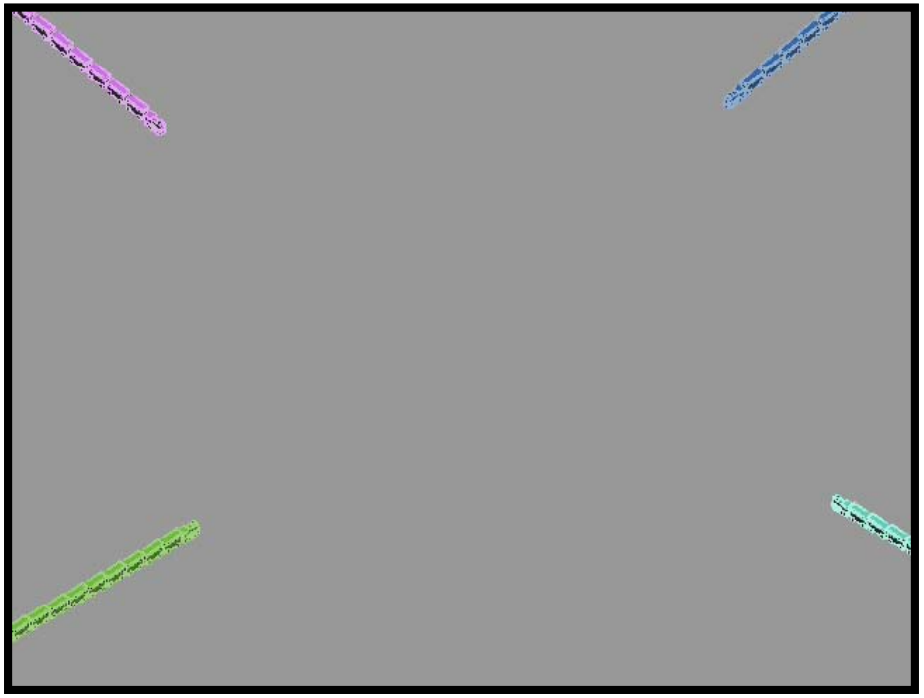


Spatial Culling using Dynamic AABB

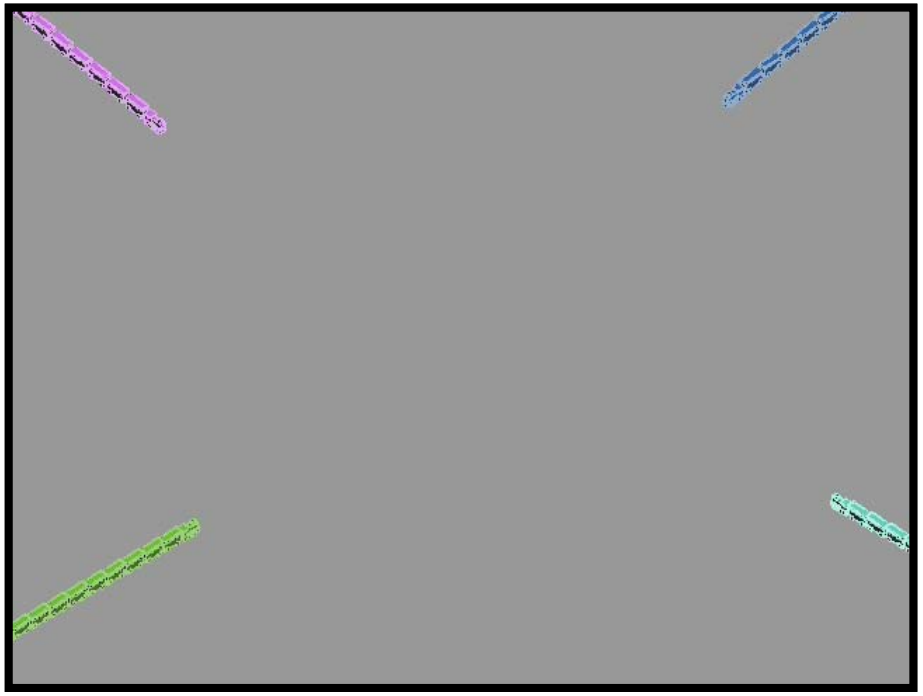
- Goal
  - Compute an axis-aligned bounding box (AABB) that bounds the motion of a moving link



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 triangles
- **Self-CCD performance**
  - **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**

Articulated Body Dynamics Benchmark

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

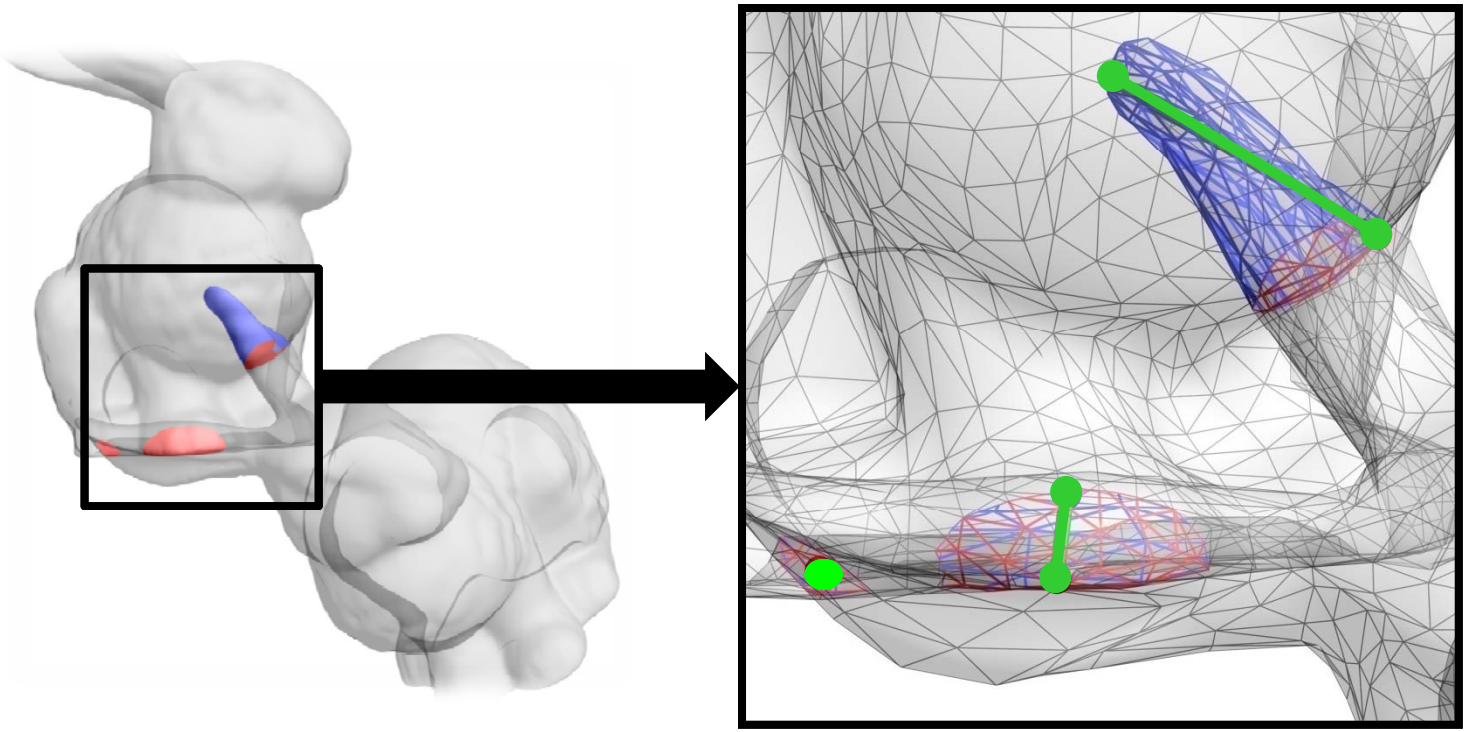
Software Implementations

- **Source codes are available**
  - http://graphics.ewha.ac.kr/FAST (2-manifold)
  - http://graphics.ewha.ac.kr/C2A (polygon-soups)
  - http://graphics.ewha.ac.kr/CATCH (articulated)
  - http://graphics.ewha.ac.kr/CCQ (for motion planning)

PENETRATION DEPTH COMPUTATION

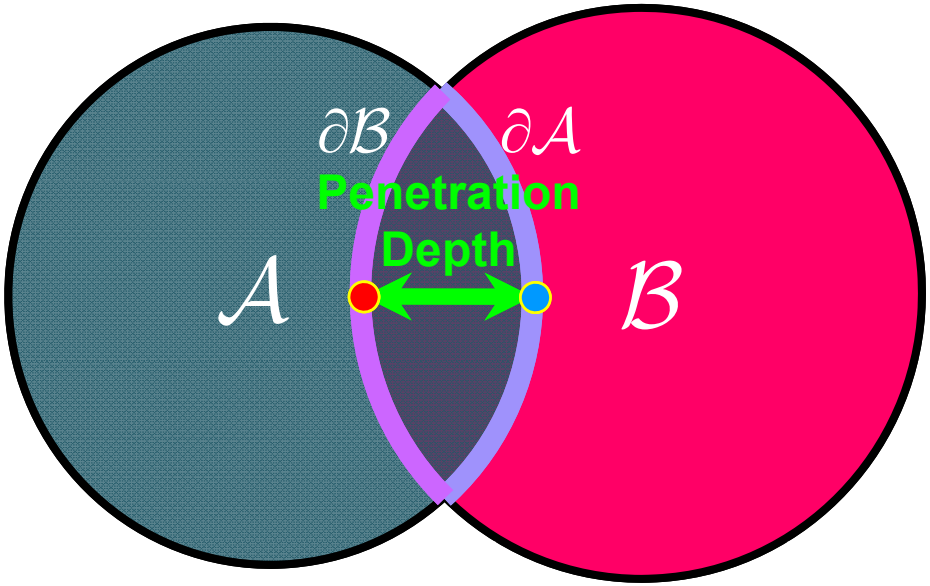
Pointwise Penetration Depth
[Tang et. al SIGGRAPH 09]

- **Defined as deepest interpenetrating points**

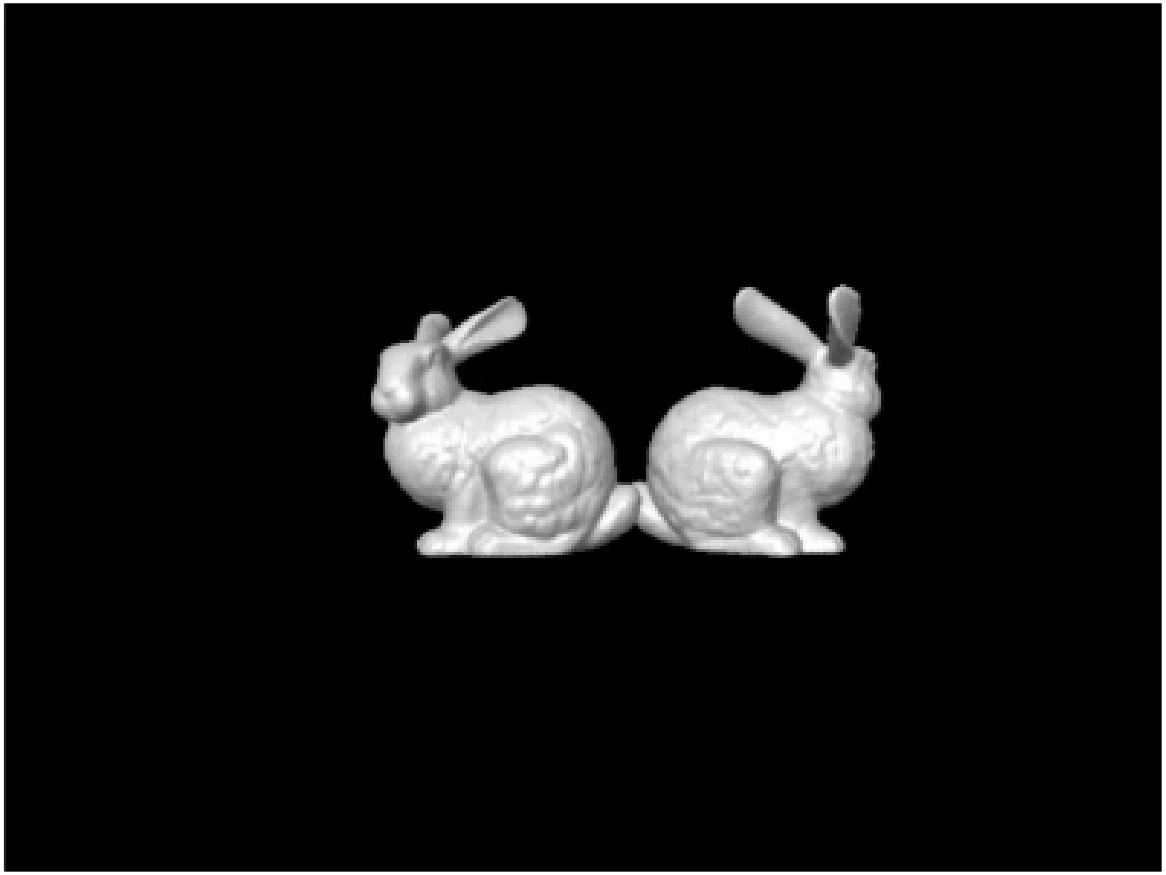


Pointwise Penetration Depth

1. Find intersection surfaces $\partial\mathcal{A}$ and $\partial\mathcal{B}$
2. Penetration depth = $H(\partial\mathcal{A}, \partial\mathcal{B})$

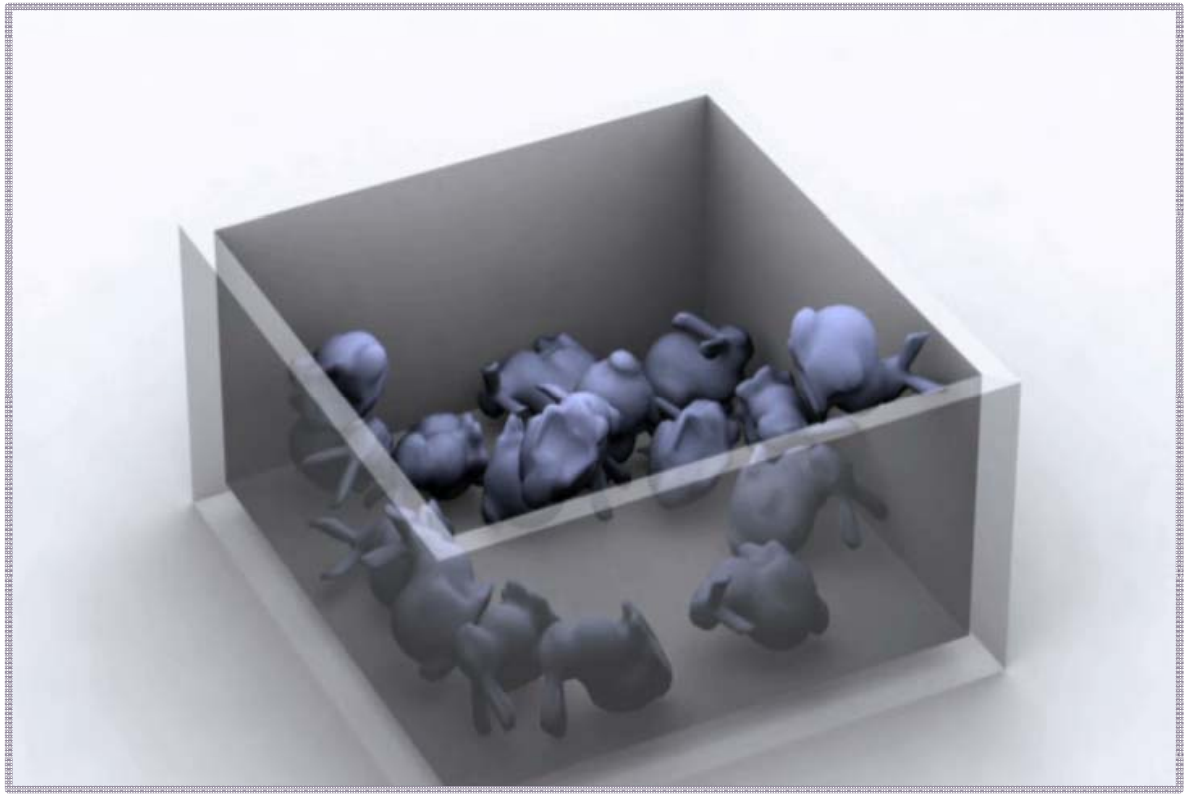


Pointwise Penetration Depth



Demo (40K Bunny vs 40K Bunny)

Benchmark: Pointwise PD



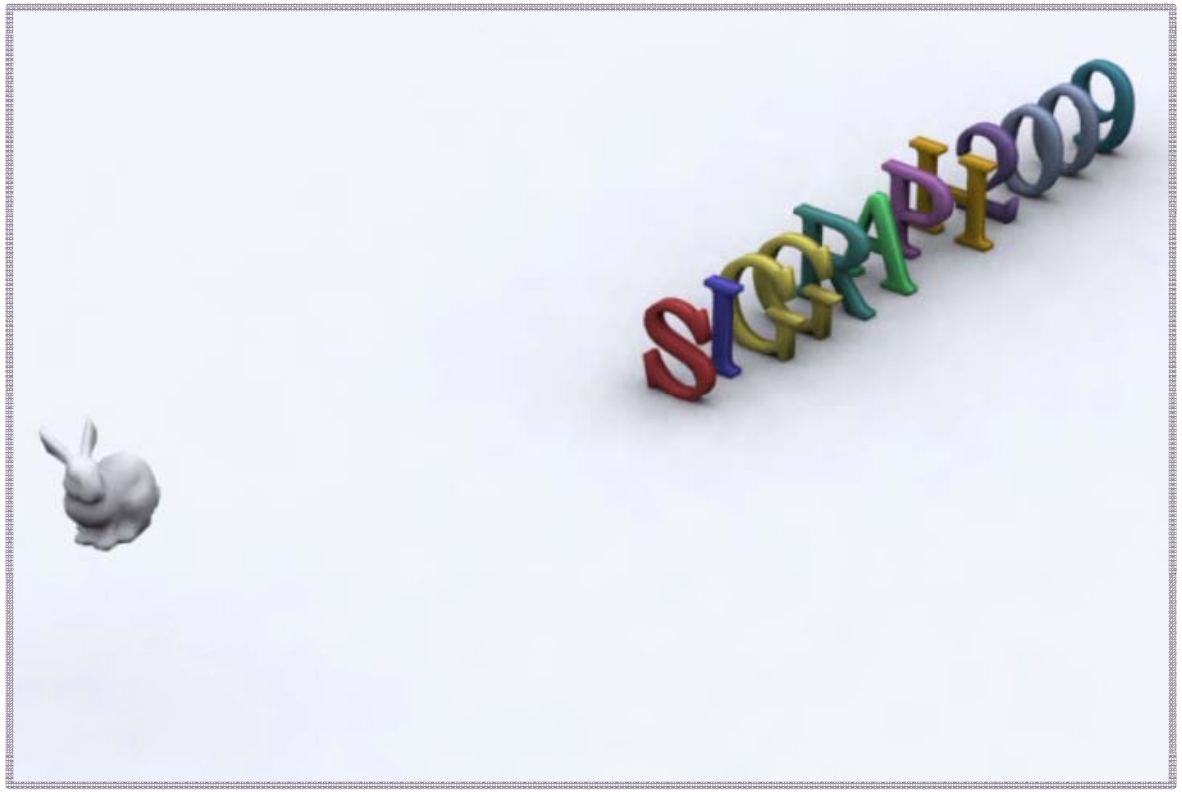
Model complexity

- 50K tri

Avg. Performance

- 3.88ms/pair

Benchmark: Pointwise PD



Model complexity

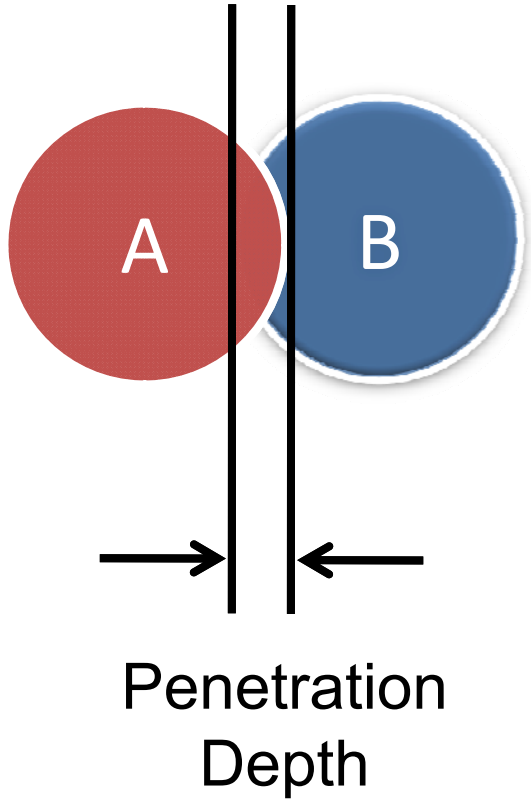
- 3.5K tri

Avg. performance

- 0.95ms/pair

Penetration Depth
[Dobkin 93]

- Minimum translational distance to separate overlapping objects



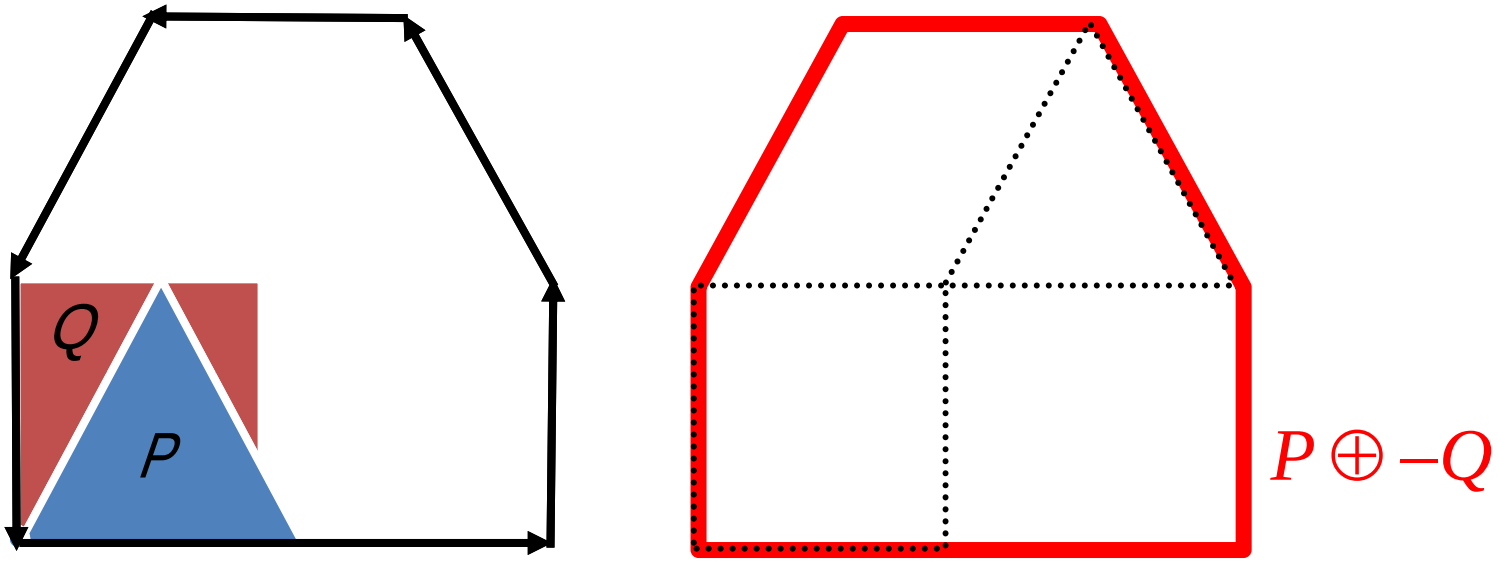
Previous Work on PD

- Convex polytopes -[Cameron and Culley86], [Dobkin93], [Agarwal00], [Bergen01], [Kim04]
- Non-convex polyhedra -[Kim02],[Redon and Lin06], [Lien08a,b], [Hachenberger09]
- Distance fields -[Fisher and Lin01], [Hoff02], [Sud06]
- Pointwise PD -[Tang09]
- Generalized PD – [Ong and Gilbert96], [Ong96], [Zhang07]
- Volumetric PD - [Wellner and Zachmann09]

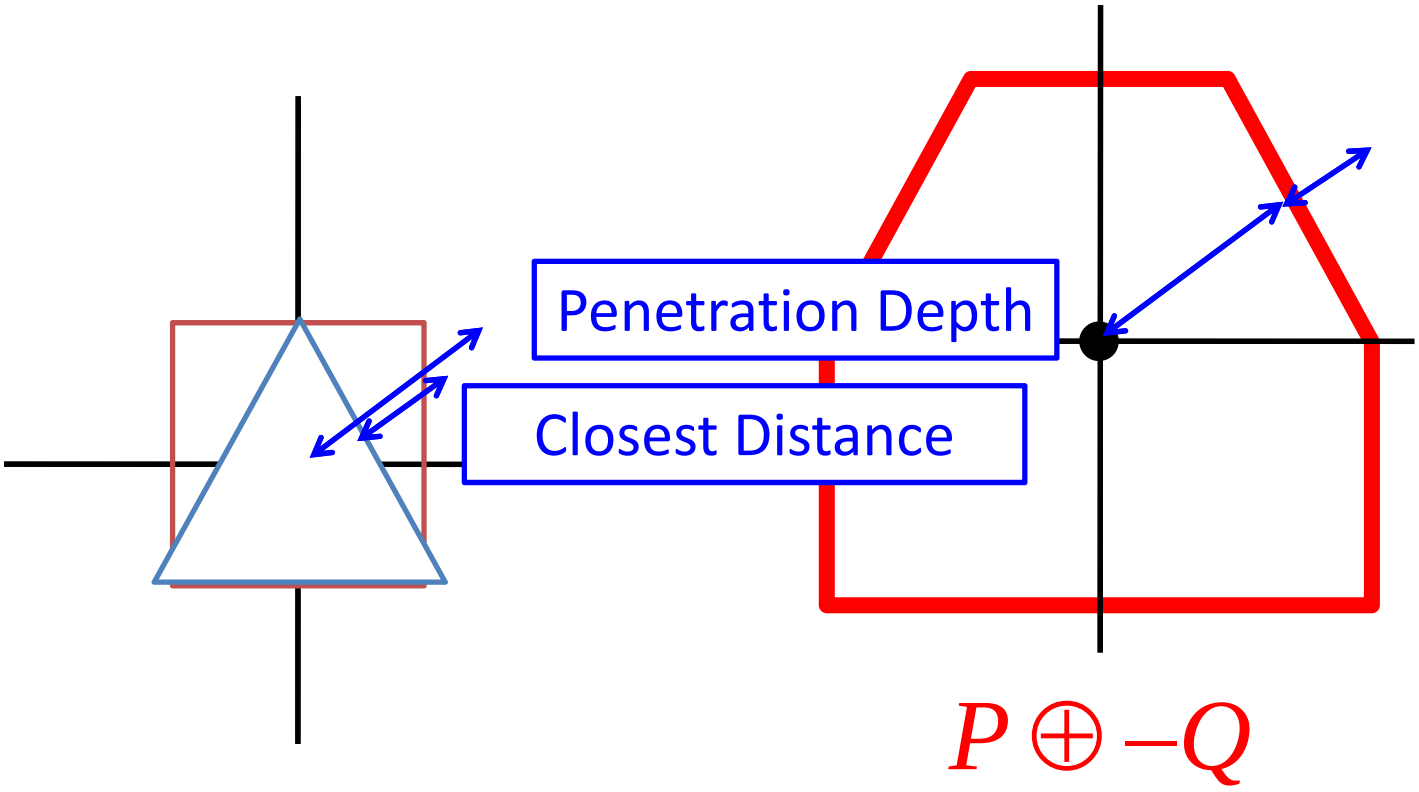
Minkowski Sum

$$P \oplus Q = \{\mathbf{p} + \mathbf{q} \,|\, \mathbf{p} \in P, \mathbf{q} \in Q\}$$

$$P \oplus -Q = \{\mathbf{p} - \mathbf{q} \,|\, \mathbf{p} \in P, \mathbf{q} \in Q\}$$

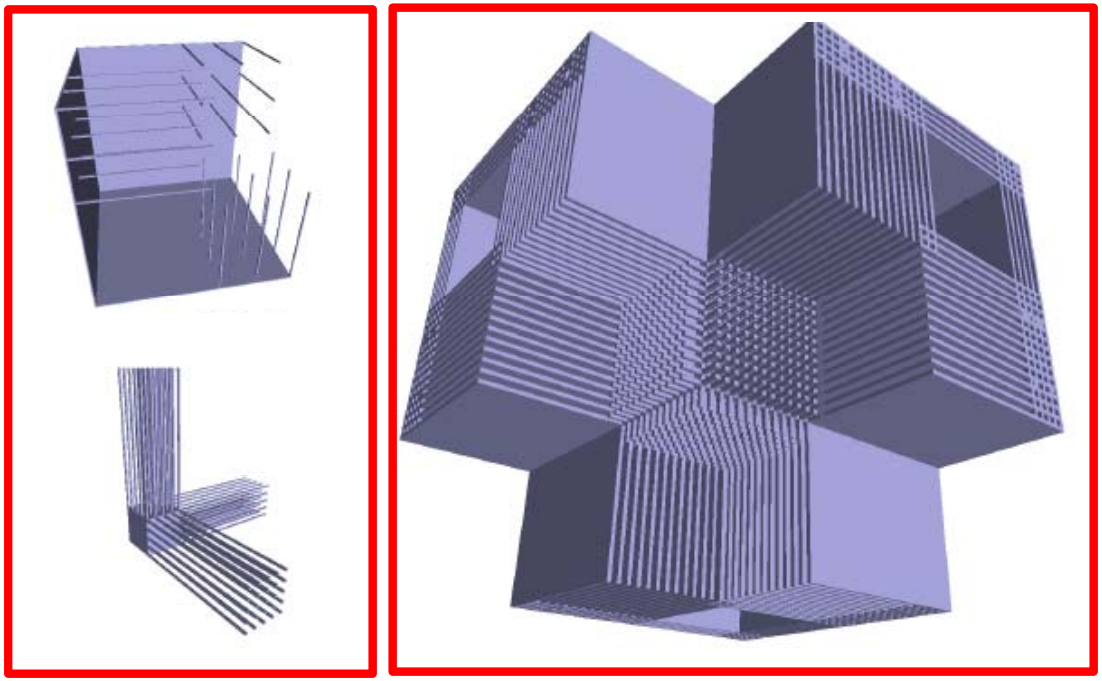


Proximity VS Minkowski Sum

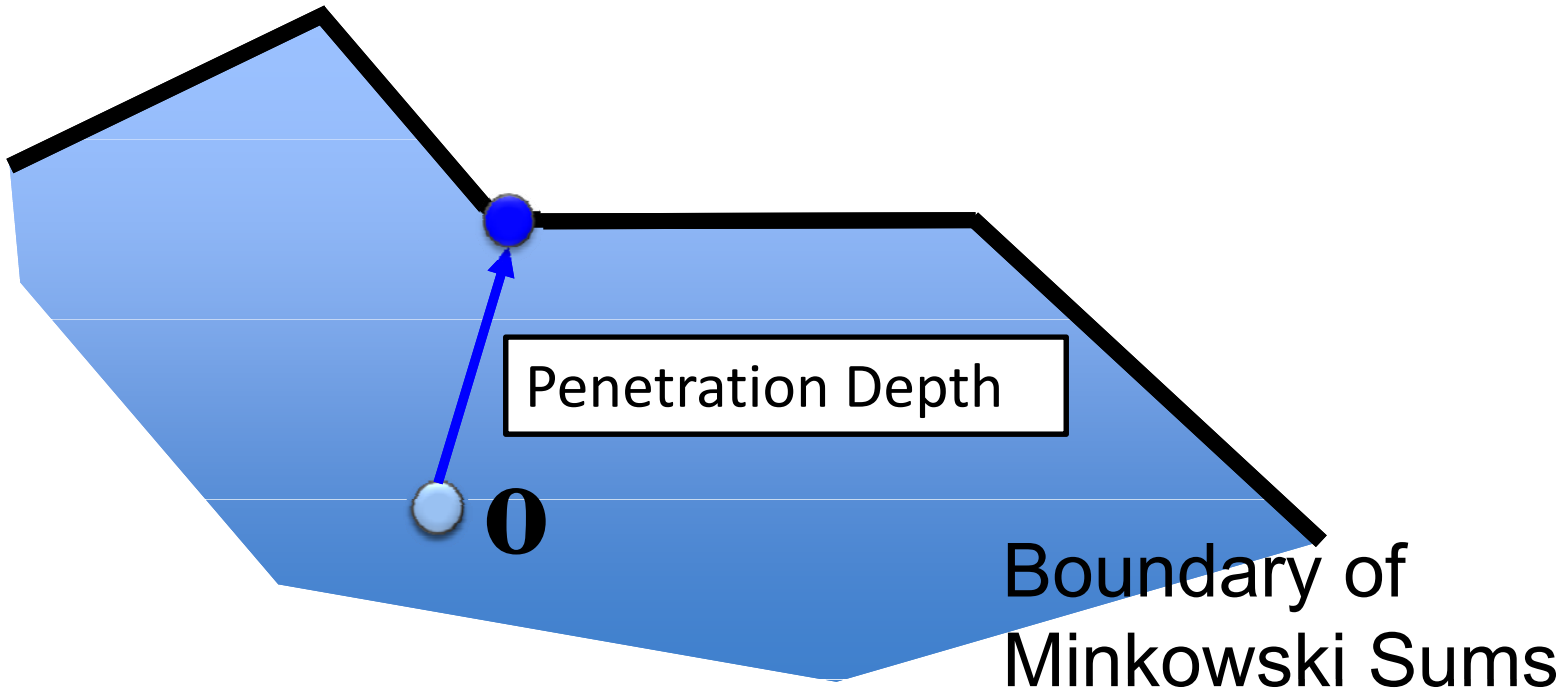


Combinatorial Explosion

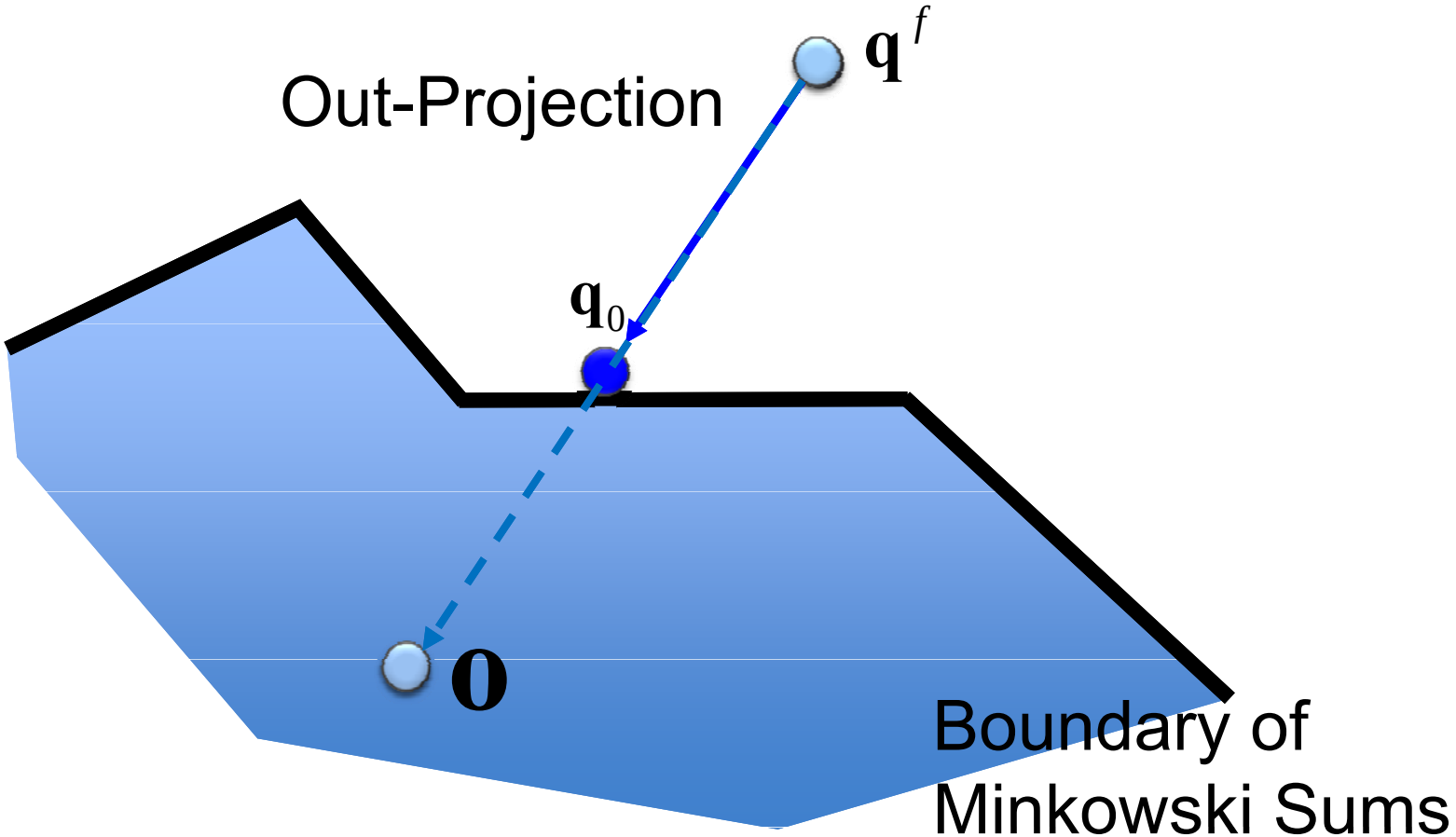
- **Complexity of Minkowski Sum**
  - $O(m^3n^3)$ with m and n triangles



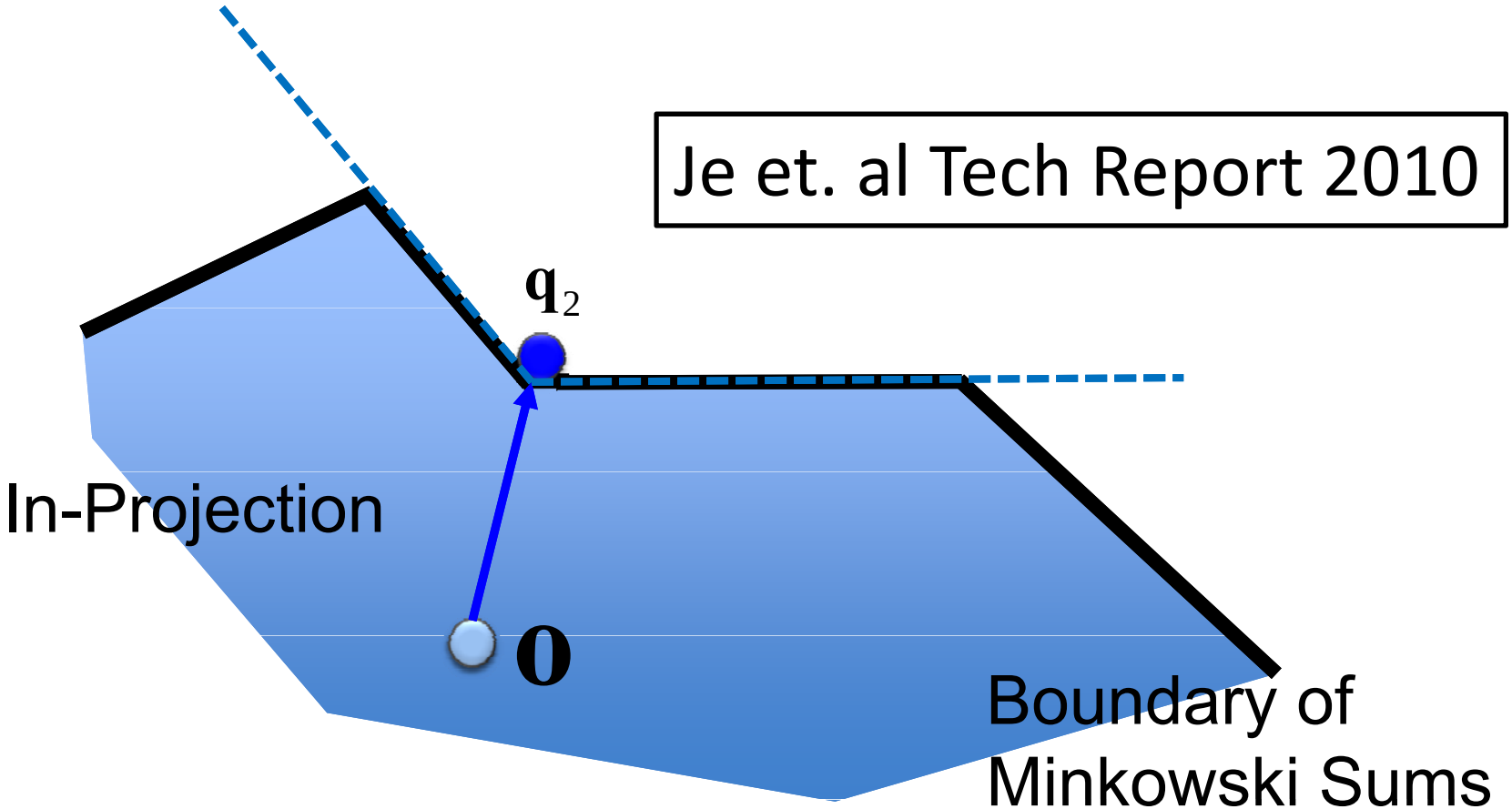
PD Estimation



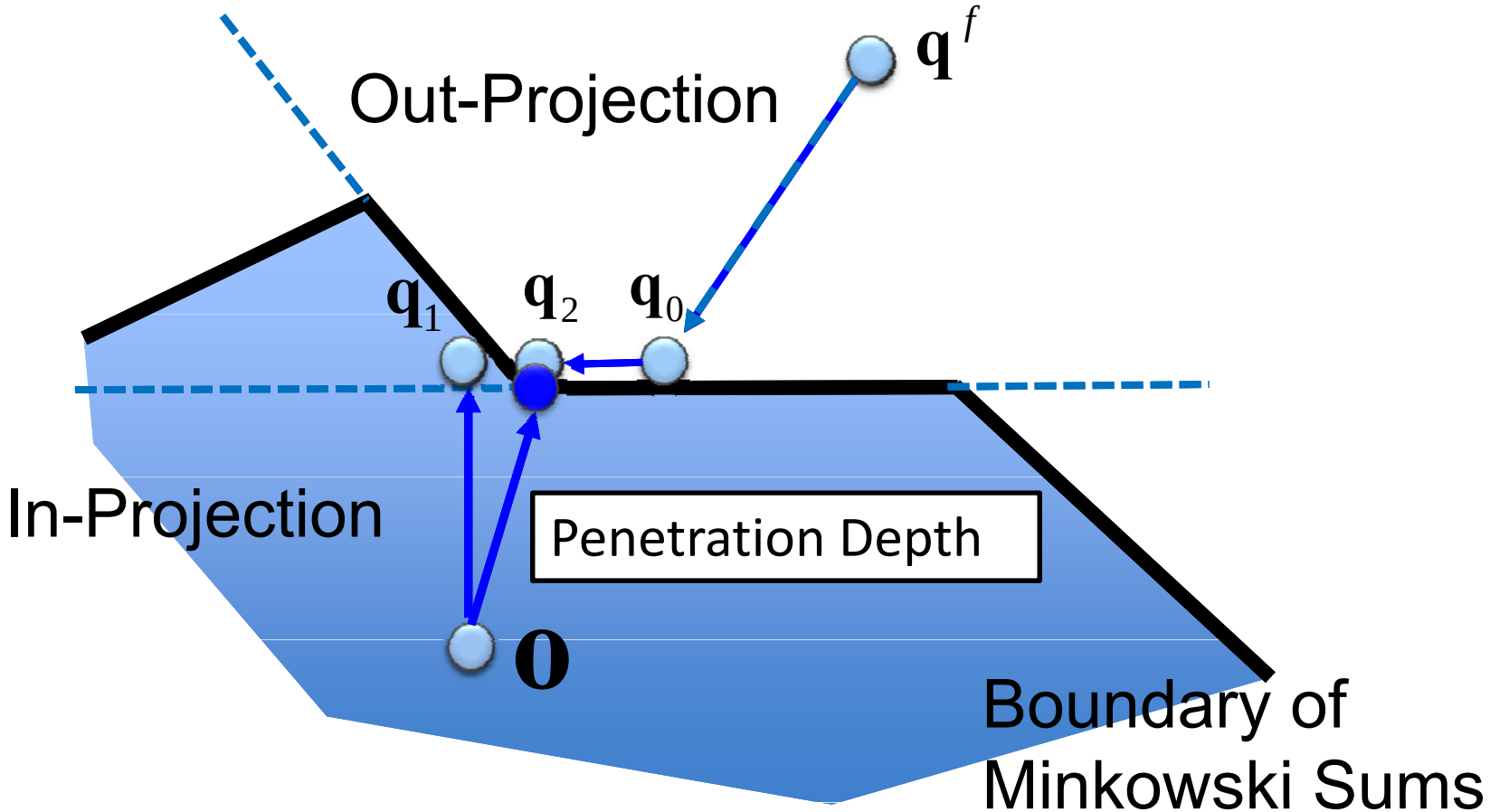
Out-Projection = CCD



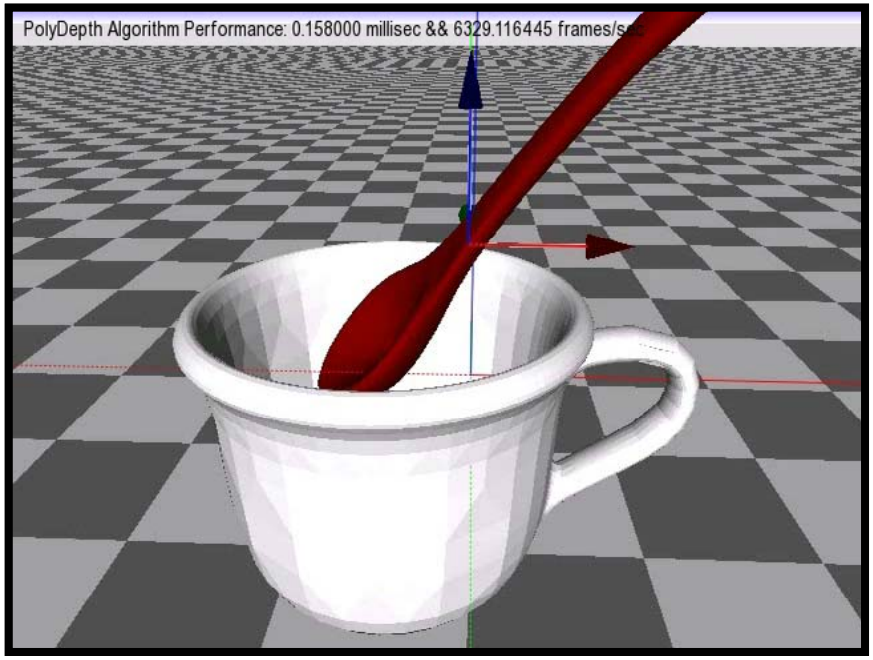
In-Projection = LCP



PolyDepth: Iterative Optimization



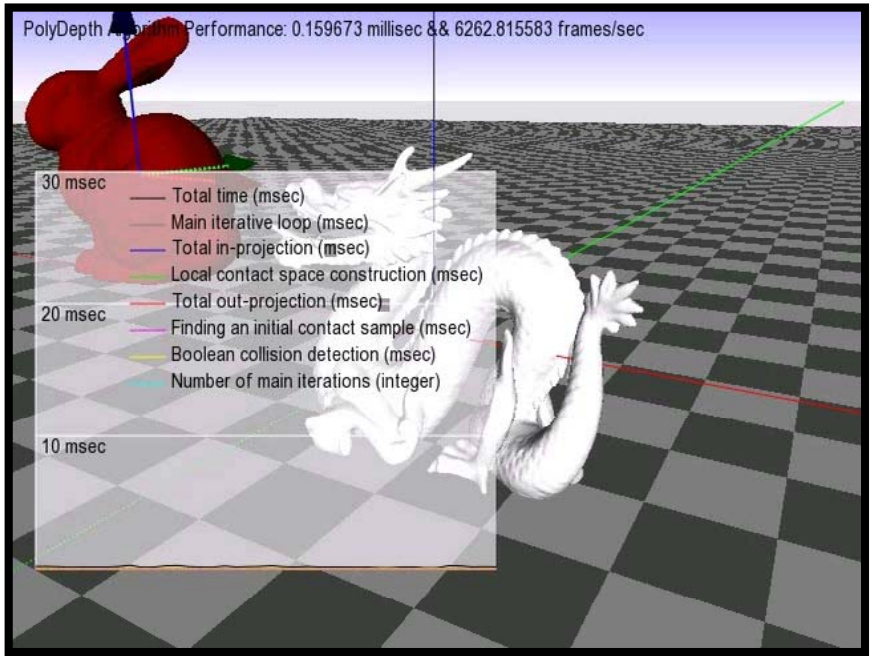
PolyDepth Performance



- Spoon: 1.3K triangles
- Cup: 8.4K triangles

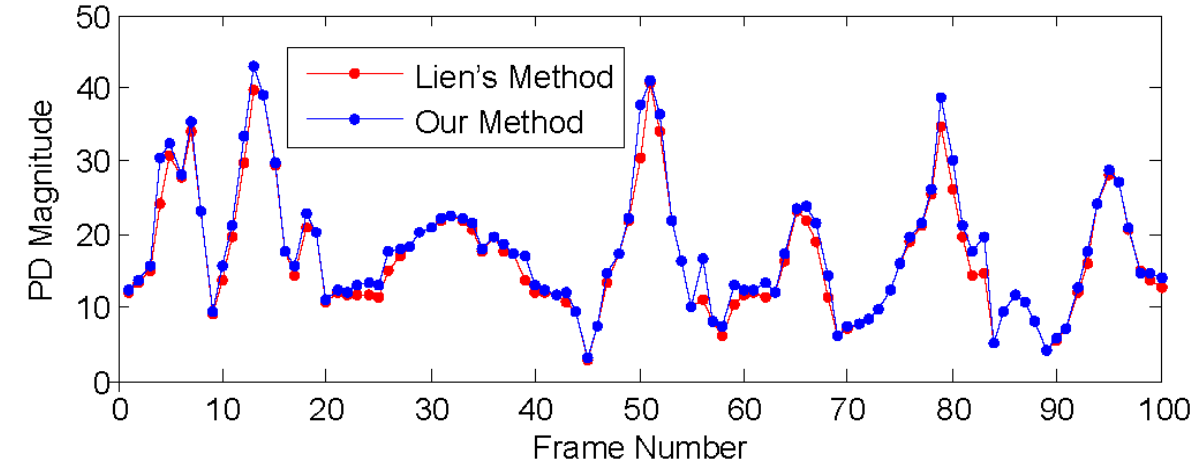
- Time: 1~7 msec

PolyDepth Performance

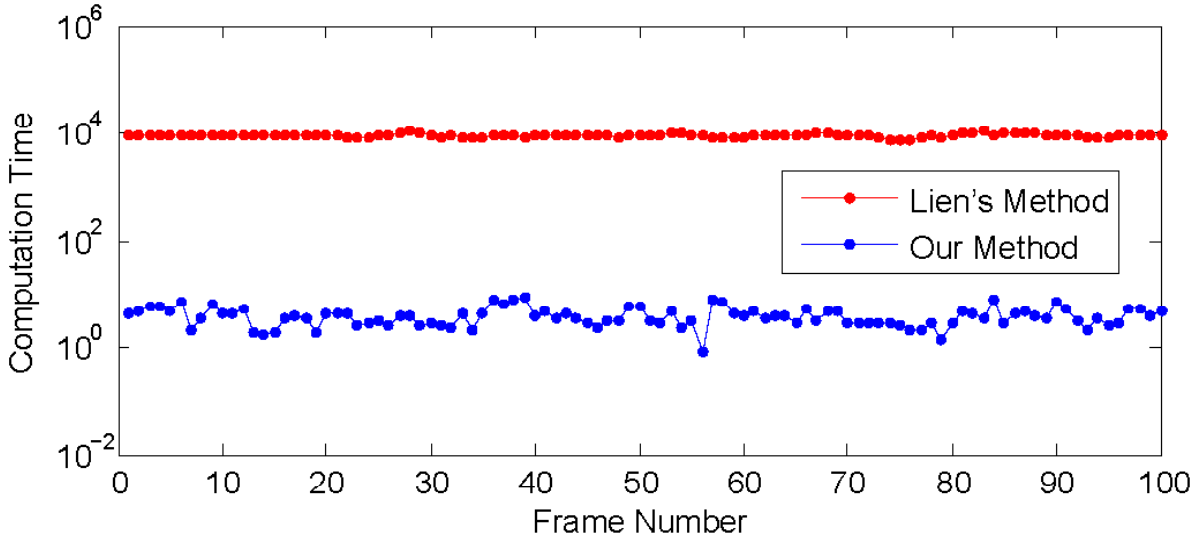


- Bunny: 40K triangles
- Dragon: 174K triangles
- Time: 2~15 msec

Comparison against Exact Solution



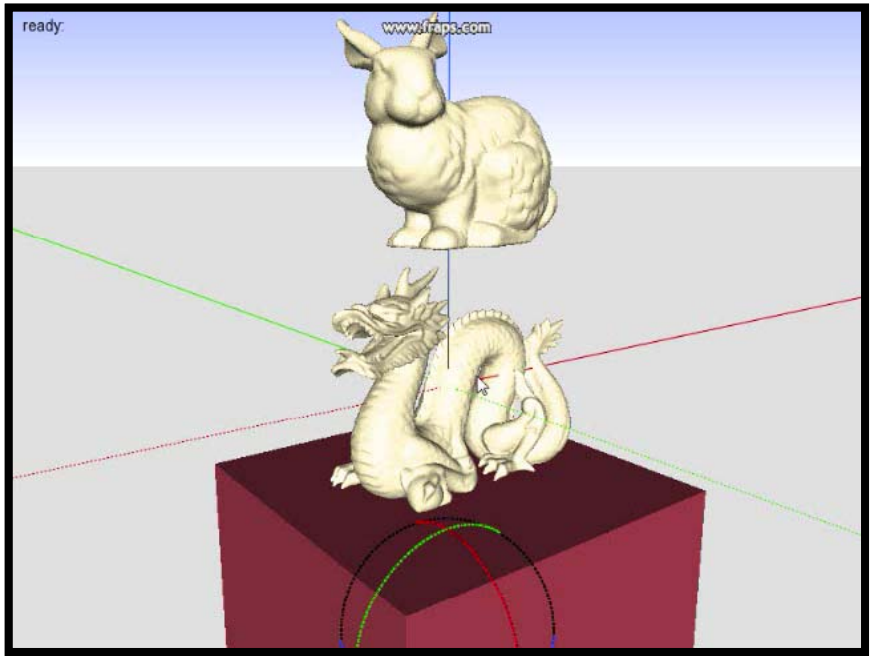
Accuracy



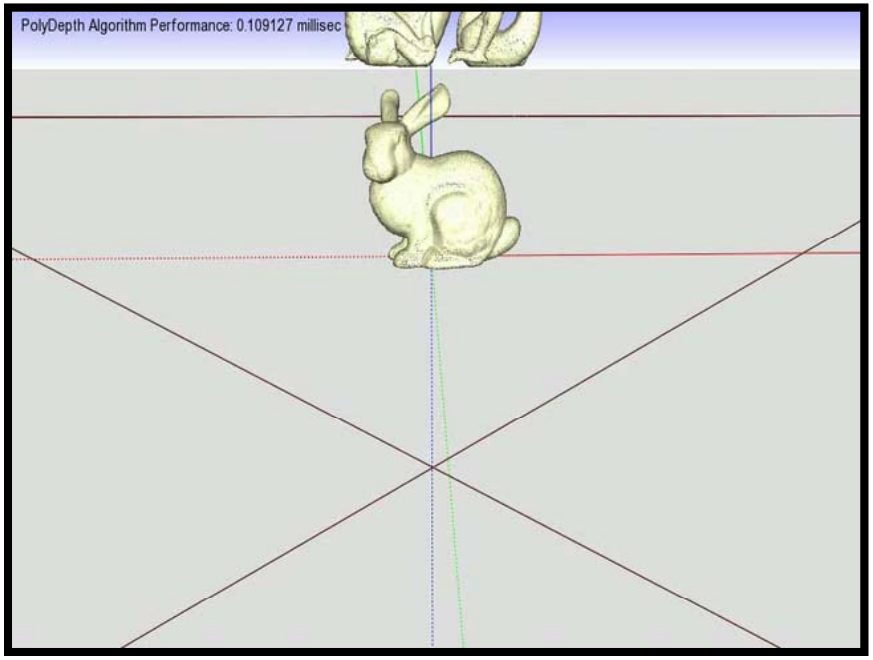
Performance

APPLICATIONS

Real-time Physics Simulation using PD



214K triangles in total

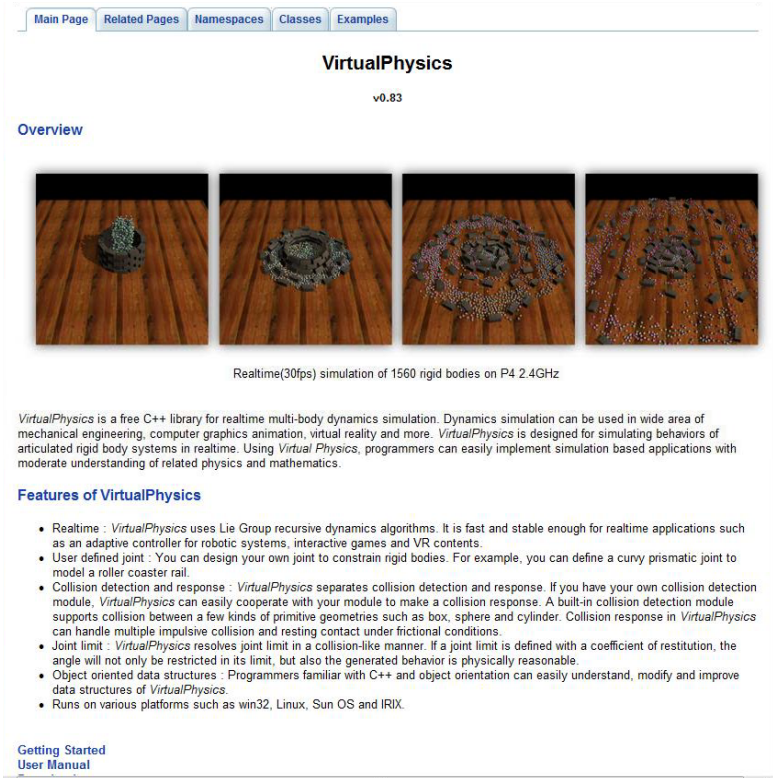


802K triangles in total

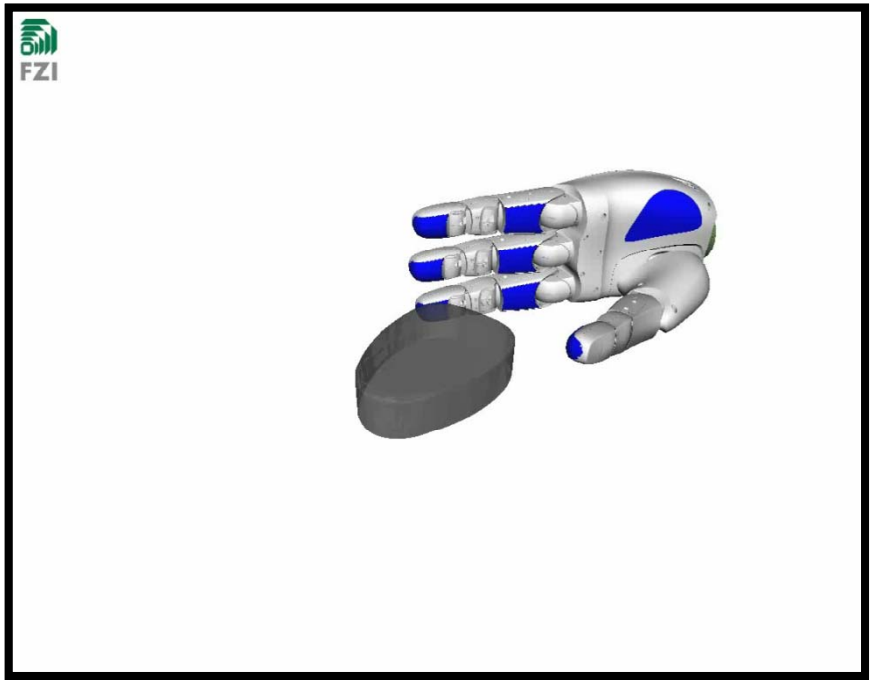
Integration with Physics Engine

http://virtualphysics.kr

VIRTUAL PHYSICS
REALTIME DYNAMICS SIMULATION LIBRARY

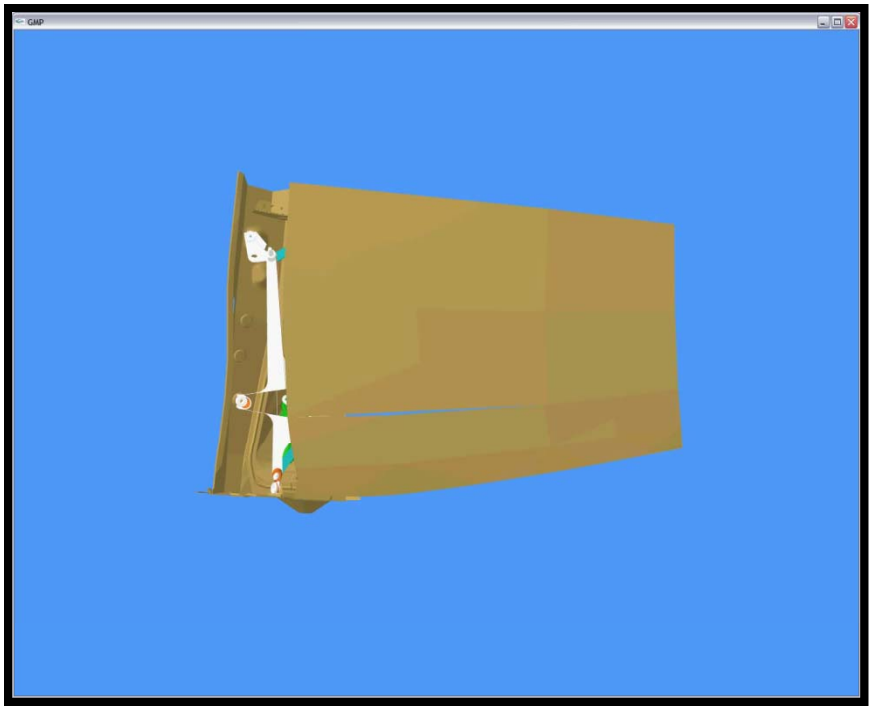


Robotic Grasping using CCD



With Zhixing Xue @ FZI

CAD Disassembly using CCD



With Liangjun Zhang @ Stanford/UNC

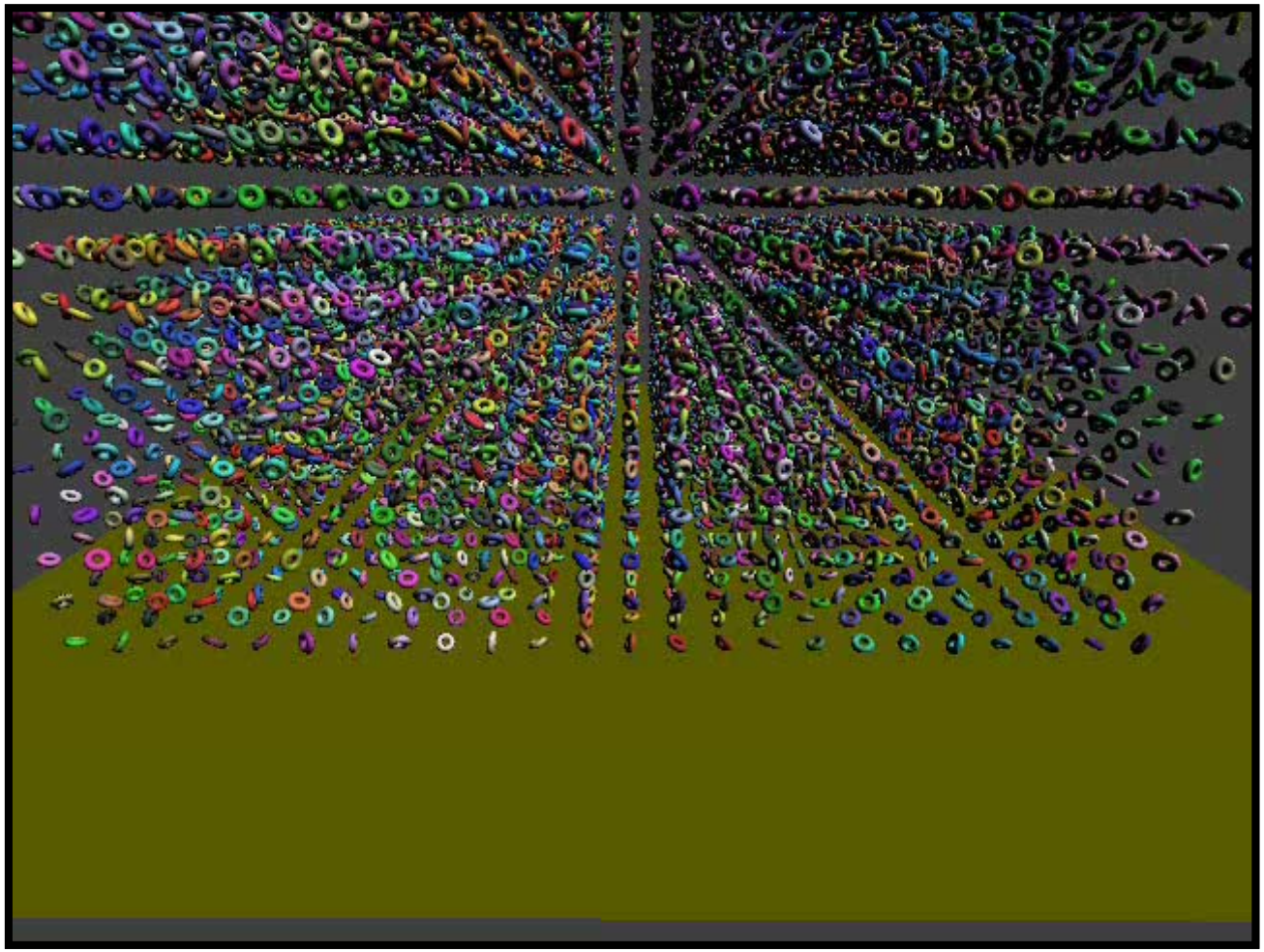
Summary

	CCD	PD
Concept	Collision avoidance	Collision correction
Usages	1. Constraint-based dynamics 2. Exact motion planning 3. Grasping	1. Penalty-, impulse-based dynamics 2. Retraction-based motion planning
Complexities	O(mn)	$O(m^3n^3)$

Future Work

- **Continuous collision detection**
  - N-body
  - Non-linear motion
- **PD**
  - Articulated body
  - Deformable
  - N-body

Collision Culling of a Million Bodies on GPUs



Real-time Dynamics Simulation of 16,000 Rigid Bodies

Acknowledgements

- Min Tang, Xinyu Zhang, Minkyoung Lee, Youngeun Lee (Ewha)
- Stephane Redon (INRIA)
- Dinesh Manocha (UNC)
- Liangjun Zhang (Stanford)
- Zhixing Xue (FZI)

- KEIT/MKE (IT core research)
- KRF (Young investigator award)

Main References

- X. Zhang, M. Lee, Y. Kim, Interactive Continuous Collision Detection for Non-convex Polyhedra, Pacific Graphics 2006 http://graphics.ewha.ac.kr/FAST

- X. Zhang, S. Redon, M. Lee, Y. Kim, Continuous Collision Detection for Articulated Models using Taylor Models and Temporal Culling, SIGGRAPH 2007 http://graphics.ewha.ac.kr/CATCH

- M. Tang, Y. Kim, D. Manocha, C^2A: Controlled Conservative Advancement for Interactive Continuous Collision Detection, IEEE ICRA 2009 http://graphics.ewha.ac.kr/C2A

Main References

- M. Tang, M. Lee, Y. Kim, Interactive Hausdorff Distance Computation for General Polygonal Models, SIGGRAPH 2009
  http://graphics.ewha.ac.kr/HDIST

- C. Je, M. Tang, Y. Lee, M. Lee, Y. Kim, PolyDepth: Real-time Penetration Depth Computation using Iterative Contact-space Projection, ACM Transactions on Graphics 2012
  http://graphics.ewha.ac.kr/PolyDepth

- M. Tang, Y. Kim, D. Manocha, Efficient Local Planning using Connection Collision Query, Workshop on Algorithmic Foundations of Robotics 2010
  http://graphics.ewha.ac.kr/CCQ

Thank you for listening!

http://graphics.ewha.ac.kr