Collision Detection for Deformable Models – Techniques for both Discrete and Continuous

A particularly difficult benchmark. The flamenco dancer's dress consists of seven layers of cloth. All of those layers are tested for self-collision as well as with each other and the body.

Highlights

- **Offers an efficient technique for maintaining a high-quality bounding volume hierarchy.**
- **Provides superior culling for self-intersection using a continuous formulation of the normal cone.**
- **For adjacency composition, orphan sets provide a tight bound on the work required on adjacent triangles.**
- **Provides a simple, computationally and memory efficient to improve culling and eliminate duplicate queries for all types of collision detection.**
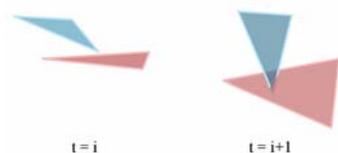
The Challenge

Collision detection (CD) is a fundamental query in many aspects of computing: motion planning, robotics, physical simulation, etc. Naively implemented, collision detection is an incredibly expensive task. There has been a great deal of research over the years in methods to accelerate it.

Much of the early research focused on collision detection between rigid objects at discrete times. As our tools and methods have become more sophisticated these initial techniques have proven to be insufficient.

Current applications use a more complex formulation of the basic problem. We are interested in simulation deformable objects, such as cloth, human tissue, or other deformable materials. Techniques for finding collisions on rigid objects could rely on expensive pre-processing algorithms whose results could be cached for later use. With deformable models, the cached data becomes invalidated and the algorithms would have to be run at every time step, precluding the possibility of using expensive pre-processing tasks.
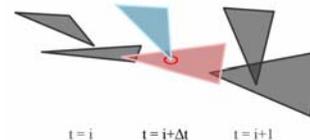
Second, instead of performing discrete collision detection, contemporary applications perform *continuous* collision detection. Discrete collision detection only detects collisions at discrete moments in time. For thin surfaces or fast moving objects, this can cause collisions to be

Discrete collision detection looks for collisions at fixed points in time. No collision at $t = i$, but there is a collision at $t = i + 1$.

Continous Collision detection finds the instance of collision *during* an interval. In this case, collision occurs at $t = I + \Delta t$.

missed. For high-fidelity simulation, it is necessary to catch these collisions. Continuous collision detection performs a higher-order test to determine if there is any intersection between two objects during the interval defined by two points.

We show a collection of algorithms for generally performing collision detection between deformable models, with particular emphasis on the more sophisticated formulation of continuous collision detection.

Selective Restructuring

Bounding volume hierarchies (BVHs) built on a triangle mesh offer the ability to cheaply cull collision tests. If the bounding volume (BV) of two triangles, or two *regions* of triangles, do not overlap, then the triangles cannot overlap.

When a model deforms, the BVH can degrade. Regions whose bounding volumes offered excellent culling initially may no longer cull efficiently.

Selective Restructuring offers an efficient means to identify the cause of such degradation and an inexpensive method to correct the localized degradation.

Connectivity-based Culling

Sometimes it is necessary to test for collisions between triangles of the same mesh; simulating cloth is a prime example of this. In these cases, traditional bounding volume techniques prove insufficient. If two triangles share an edge or a vertex (i.e. they are adjacent), their bounding volumes *must* overlap. This means the BVH will not be able to cull these tests and we must turn to an alternative method.

If we examine the normals of all the triangles in a small region of a mesh, we can provide an alternative test to determine if self-intersection is possible. If the mesh is planar, all normals point in the same direction. The only way for there to be self-intersection is if some of those normals are facing different directions. The concept of normal cone is a cone in which all normals lie. If the cone has a small enough angle, then self-intersection is impossible.

We introduce the concept of Continuous Normal Cone (CNC) to apply the same principle for continuous collision detection. The CNC encapsulates all of the normal directions which will arise in the interval.
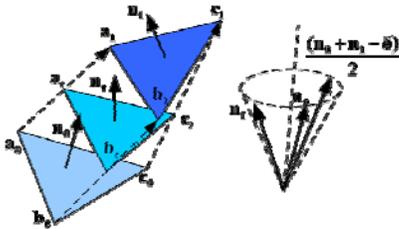

The triangle moves from the position at t_0 to t_1. The normal sweeps out a cone as the triangle sweeps along a path. The normal cone captures all directions in that path.
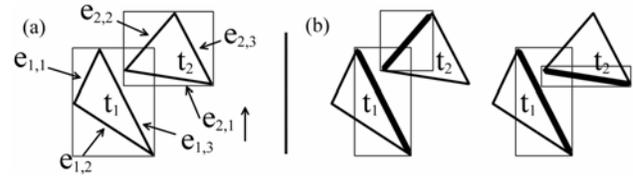
Representative Triangles

Most collision detection techniques operate on triangle pairs. However, the actual collision detection test usually becomes tests between triangle features (vertices, edges and faces.) This disjoint model can lead to two artifacts: duplicate queries and overly coarse culling resolution.

Duplicate queries arise because a single feature can be shared by multiple triangles. If the CD algorithm is focused on triangle pairs and a vertex intersects a face, then the algorithm will perform the test multiple times – one pair for every triangle which is incident to the vertex.
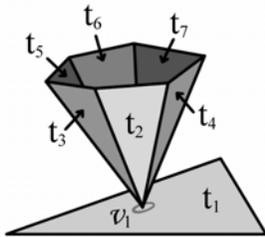
Vertex v_1 intersects the triangle. Six triangle-pairs intersect but only one vertex-face pair. This leads to duplicate tests and results.

If we only cull with triangle BVs, we miss the opportunity to cull more aggressively. Features are smaller than their triangles. By performing culling on BVs based on the features we can improve culling without changing the nature of the culling tests.

The triangle BVs on the left require nine edge-edge tests. On the right, only two are necessary.

Representative Triangles (RTris) addresses both of these issues by eliminating duplicate queries without any expensive database queries and improves culling by exploiting feature-level BVs.

Current Project Members

Dinesh Manocha, Phi Delta Theta/Matthew Mason Distinguished Professor, UNC-Chapel Hill
Min Tang, Visiting Scholar
Sung-Eui Yoon, Assistant Professor at Korea Advanced Institute of Science and Technology (KAIST)
Sean Curtis, Graduate Research Assistant

Collaborators

Rasmus Tamstorf, Walt Disney Animation Studios

Research Sponsors

DARPA/RDECOM, ARO, NSF, Intel, Walt Disney Animation Studios

Selected Publications

Tang, M., S. Curtis, S.-E. Yoon and D. Manocha. "Interactive Continuous Collision Detection between Deformable Models using Connectivity-Based Culling," *ACM Solid and Physical Modeling Symposium (SPM)*, 2008. To appear.

Curtis, S., R. Tamstorf and D. Manocha. "Fast Collision Detection for Deformable Models using Representative-Triangles," *Proc. of Symposium on Interactive 3D Graphics and Games (I3D)*, 2008.

Yoon, S.-E., S. Curtis and D. Manocha. "Ray Tracing Dynamic Scenes using Selective Restructuring," *Eurographics Symposium on Rendering (EGSR)*, 2007.

Keywords

Collision detection; continuous collision detection; simulation

For More Information

Sean Curtis
Phone (919) 962-1835
E-mail: seanc@cs.unc.edu

http://gamma.unc.edu/RTRI - http://gamma.unc.edu/CBC - http://gamma.unc.edu/SR