Deformable Materials 3

Adrien Treuille
Overview

• Elastic Collision Detection
• Collision Detection for Reduced Models
• Surface-Based Elastics
• New Question
Overview

- Elastic Collision Detection
- Collision Detection for Reduced Models
- Surface-Based Elastics
- New Question
Collision Detection

- **Broad Phase:**
  - Guess collisions between objects.

- **Narrow Phase:**
  - Determine collision points.
Broad Phase
Fast Interval Operations

- Temporal coherency: keep list between timesteps.
- Use insertion sort. **Expected O(n) runtime.**
- Update overlaps *during* insertion sort.
- Three cases:
  - A minimum and a maximum flip. **Toggle overlap bit.**
  - Two minima flip. **Don’t toggle.**
  - Two maxima flip. **Don’t toggle.**

```cpp
class BroadIntersection {
  int body_1_index;
  int body_2_index;
  bool x_overlap;
  bool y_overlap;
  bool z_overlap;
}
```
Narrow Phase

• Find exact collision point.
• Use a geometric partitioning algorithm.
• Two types:
  • Bounding Volume Hierarchies
  • Spatial Partitioning
BVH vs. Spatial Partitioning

**BVH:**
- Object centric
- Spatial redundancy

**SP:**
- Space centric
- Object redundancy

(From Doug James’s Slides.)
Friday, November 12, 2010
BVH vs. Spatial Partitioning

**BVH:**
- Object centric
- Spatial redundancy

**SP:**
- Space centric
- Object redundancy

(From Doug James’s Slides.)
BVH vs. Spatial Partitioning

**BVH:**
- Object centric
- Spatial redundancy

**SP:**
- Space centric
- Object redundancy

(From Doug James’s Slides.)
BVH vs. Spatial Partitioning

**BVH:**
- Object centric
- Spatial redundancy

**SP:**
- Space centric
- Object redundancy

(From Doug James’s Slides.)

Friday, November 12, 2010
Bounding Volume Hierarchies

- How to create a BVH:
  - Geometric Subdivision
  - Topological Subdivision
    - How implement?
    - Which is better?

- How to update a BVH:
  - Bottom Up (How?)
  - Directly (How?)
  - Which is faster?
Triangle Intersection

- Edge-Edge
- Vertex-Face
Summary

- **Broad Phase:**
  - Guess collisions between objects.

- **Narrow Phase:**
  - Determine collision points.
Overview

• Elastic Collision Detection
• Collision Detection for Reduced Models
• Surface-Based Elastics
• New Question
Overview

● Elastic Collision Detection

● Collision Detection for Reduced Models

● Surface-Based Elastics

● New Question
Collision Detection for

Figure 2: Example deformation: (a) Reference shape p (b) Displacement field $U_{*1}$ (c) Field $U_{*2}$ (d) Deformed shape $p'$. 

\[ p' = p + Uq \quad \text{or} \quad p'_i = p_i + \sum_{j=1}^{M} U_{ij}q_j. \]
1.1 Related Work

Recently, hardware-accelerated collision detection methods have been explored [Guibas et al. 2002; Brown et al. 2001; Larsson and Akenine-Matthies 2003].

Intelligent methods for updating the bounding hierarchies (e.g., [van den Bergen 1997; Ganovelli et al. 2000]). Alternatives have been drawn to hierarchies with easy to compute bounds each time step, before queries are performed. As a result, much attention has been drawn to hierarchies with easy to compute bounds (Secs. 3.2 and 3.5).

Collision detection with bounding volumes is a canonical example of a bounding volume hierarchy used for collision detection and proximity queries (e.g., [Quinlan 1994; Hubbard 1995; Brown et al. 2001; Guibas et al. 2002; Bradshaw and O'Sullivan 2004]). In our examples, we use an approach similar to [Quinlan 1994] for binary sphere tree construction.

2 Reduced Deformation Model

Suppose we have $\mathbf{p} = (p_1, \ldots, p_j, \ldots, p_N)$, so that (see Figure 2) $U_{ij}$.

Without loss of generality, we will assume that the columns of $U$ are orthonormal.

Note that even though this method and most others also compute the reduced coordinates, our method only requires touching a potentially small subset of columns of $U$.

1. To produce coarser approximations, since they are limited by the possibility of nonlinear black box process; therefore, although the shape modeling, or modal analysis.

1.1 Related Work

Collision detection with bounding volumes is a canonical example of a bounding volume hierarchy used for collision detection and proximity queries (e.g., [Quinlan 1994; Hubbard 1995; Brown et al. 2001; Guibas et al. 2002; Bradshaw and O'Sullivan 2004]).

In our examples, we use an approach similar to [Quinlan 1994] for binary sphere tree construction.

Build bounding spheres for tree nodes. This is not necessary at this stage. Note that even though this method and most others also compute the reduced coordinates, our method only requires touching a potentially small subset of columns of $U$.

1. To produce coarser approximations, since they are limited by the possibility of nonlinear black box process; therefore, although the shape modeling, or modal analysis.
Sphere Center Update

\[ c' = c + \sum_{i \in \Lambda} \beta_i u_i = c + \sum_{i \in \Lambda} \left( \sum_{j=1}^{M} U_{ij} q_j \right) \]

\[ = c + \sum_{j=1}^{M} \left( \sum_{i \in \Lambda} \beta_i U_{ij} \right) q_j \]

\[ = c + \sum_{j=1}^{M} \bar{U}_j q_j = \bar{c} + \bar{U} q \equiv c' \]

Sphere Center Update

\[
\max_{i \in \Lambda} \| \mathbf{p}'_i - \mathbf{c}' \|_2 = \max_{i \in \Lambda} \| (\mathbf{p}_i - \mathbf{c}) + \sum_{j=1}^{M} (\mathbf{U}_{ij} - \bar{\mathbf{U}}_j) \mathbf{q}_j \|_2 \\
\leq \max_{i \in \Lambda} \| \mathbf{p}_i - \mathbf{c} \|_2 + \sum_{j=1}^{M} \left( \max_{i \in \Lambda} \| \mathbf{U}_{ij} - \bar{\mathbf{U}}_j \|_2 \right) | \mathbf{q}_j | \\
\equiv R + \sum_{j=1}^{M} \Delta R_j | \mathbf{q}_j | = R + \Delta R^T \mathbf{q}^{ABS} \equiv R' \quad (7)
\]

Overview

- Elastic Collision Detection
- Collision Detection for Reduced Models
- Surface-Based Elastics
- New Question
Overview

• Elastic Collision Detection
• Collision Detection for Reduced Models
• Surface-Based Elastics
• New Question
Surface-Based Elastics

\[ A = \sum_{j \in nbr(i)} (x_j(t) - x_i(t)) (x_j^0 - x_i^0)^T. \]

\[ c_i(t) = \frac{1}{|nbr(i)|} \sum_{j \in nbr(i)} \left( R(x_i^0 - x_j^0) + x_j(t) \right). \]

\[ L_i(t) = \frac{c_i(t) - x_i(t)}{h^2}. \]
Volumetric Behavior

\[ B_i(t) = \frac{\left( \frac{\|x_{ib}^0\|}{\|x_{ib}(t)\|} - 1 \right) x_{ib}(t)}{h^2}, \]

Source: Xiaohan Shi, Kun Zhou, Yiyung Tong, Mathieu Desbrun, Hujun Bao, Baining Guo. 
Example-Based Dynamic Skinning in Real Time. ACM TOG (SIGGRAPH 2008)
Example

our result

physically-based simulation
Overview

- Elastic Collision Detection
- Collision Detection for Reduced Models
- Surface-Based Elastics
- New Question
Overview

• Elastic Collision Detection
• Collision Detection for Reduced Models
• Surface-Based Elastics
• New Question
Questions

- How could we represent a human body on a computer with few dimensions.
- What kind of optical technology could we use to capture a human body?
- How can we convert the captured data into the human body representation.
- In animation, what do you think are the most important aspects of human motion to capture / model?
- Physically / Stylistically?
1. How could we represent a human body on a computer with few dimensions.
   - use a skeleton
     - without removing joints, we can't get any more reduced than that
     - certain joints only have so many degrees of freedom
       - and bound constraints on these degrees of freedom
     - model skeleton with muscles

2. What kind of optical technology could we use to capture a human body?
   - skeletal motion capture technology (based on points located on the body)
   - how could you capture a face?
     - capture the motion of surface points on the face
     - this is probably the most important aspect (or at least the most difficultly)
   - Need at least two viewpoints to see every "part of the body"
     - but other than that, there many different sensors that we could use
   - Microsoft Kinnect
     - get some depth estimate from some cameras
     - USE previous knowledge of human motion to estimate skeleton

   - How can we convert the captured data into the human body representation.
     - come up with a function which maps skeletal points to surface points
       - invert this model?
     - use a direct surface representation without a skeleton
       - why do we need a skeleton?!
       - especially for faces!

   - In animation, what do you think are the most important aspects of human motion to capture / model?
     - capturing faces
     - capturing locomotion

   - Physically / Stylistically?
     - both _skeletal_ and _surface_ motion of the body are important!
       - muscle and tendon bulges, etc.
     - _smoothness_ constraints!!!
     - try to match captured acceleration and deceleration
     - human motion has lots of tiny movements... do we need to capture those
       - especially on human faces
       - "micro"-expressions
   - the most important of human motions:
     - look at the motion of the center of mass
   - when is it possible to break physical laws for stylistic reasons?
   - to what extent are human observers able to discern physical violations