PolyDepth: Real-Time Penetration Depth Computation Using Iterative Contact-Space Projection

Changsoo Je
Ewha Womans University and Sogang University

Min Tang, Youngeun Lee, Minkyoung Lee, and Young J. Kim, Ewha Womans University
Interpenetration Problem

- Modeling problem
- Numerical problem
- Random sampling

Penetration Non-penetration
How to separate them?

Penetration

Non-penetration
Penetration Depth (PD)

Minimum *translational* distance to separate intersecting *rigid* objects.
Applications
– Rigid Body Dynamics

Point of impulse application [Tang et al. 2009]
Applications
– Retraction-based Motion Planning

Retract in-collision samples to free space
[Zhang et al. 2008]
Applications
– Penalty-Based 6DOF Haptics

Penalty Forces
[Kim et al. 03]
Computational Complexity of PD

- Exact PD: $O(n^6)$ for $n$ faces

Minkowski Sums (Translational Contact Space)
Main Contribution

The first **realtime** PD algorithm for general polygonal models

Source codes available: google “polydepth”
Main Contribution

• Tight upper bound
• Single global or multiple local solutions
• Iterative optimization using in/out projections
• Efficient initialization strategies
Contents

• Previous work
• Preliminaries
• Out-projection
• In-projection
• Initialization
• Results
• Conclusion
Contents

• Previous work
• Preliminaries
• Out-projection
• In-projection
• Initialization
• Results
• Conclusion
Previous Work

• **Convex Polytopes**
  [Cameron and Culley 1986], [Dobkin et al. 1993], [Agarwal et al. 2000], [Cameron 1997], [Bergen 2001], [Kim et al. 2004]

• **Nonconvex Polyhedra**

• **Generalized Penetration Depth**
  [Zhang et al. 2007], [Nawratil et al. 2009], [Ong and Gilbert 1996], [Ong 1993], [Weller and Zachmann 2009]
Previous Work - Limitations

• Too slow for interactive applications
• Non-Euclidean
• Lower bound solution
  - Does not guarantee full separation
Contents

• Previous work
• Preliminaries
• Out-projection
• In-projection
• Initialization
• Results
• Conclusion
Minkowski Sum

• Minkowski sums for two sets \( \mathcal{A} \) and \( \mathcal{B} \)

\[
\mathcal{A} \oplus \mathcal{B} = \{a + b | a \in \mathcal{A}, b \in \mathcal{B}\}
\]

\[
\mathcal{A} \oplus -\mathcal{B} = \{a - b | a \in \mathcal{A}, b \in \mathcal{B}\}
\]
Minkowski Sum

\[ A \oplus -B \]
PD and Minkowski Sum

\[ \mathbf{A} \oplus -\mathbf{B} \]
PD and Minkowski Sum

\[ \mathbf{A} \oplus -\mathbf{B} \]

[Image of PD and Minkowski Sum diagrams]
PD Estimation

\[ \partial (\mathcal{U} \oplus -\mathcal{W}) \]
PD Estimation

\[ \partial (A \oplus -B) \]

Penetration depth

0
PolyDepth Algorithm

- Local optimization
  - Contact space localization

- Iterative optimization
  - Repeated in- and out-projections
Iterative Optimization

\[ \partial (A \oplus -B) \]

contact space
Iterative Optimization

\[ q^f \]
Iterative Optimization

Out-Protection

$q_0 \rightarrow q_f$

$\partial (A \oplus -B)$
Iterative Optimization

$q_0$

$\partial (\mathcal{U} \oplus -\mathcal{V})$
Iterative Optimization

In-Projection

$q_1$

$\partial (A \oplus -B)$
Iterative Optimization

Out-Projection

\(q_1\)  \(q_2\)  \(o\)

\(\partial (A \oplus -B)\)
Iterative Optimization

$q_2$

$o$

$d(A \oplus -B)$
Iterative Optimization

\[ q_3 \xrightarrow{\text{In-Pro}\text{jection}} \text{PD} \]

\[ d(\mathbf{A} \oplus -\mathbf{B}) \]

\[ \partial \]
Contents

• Previous work
• Preliminaries
• Out-projection
• In-projection
• Initialization
• Results
• Conclusion
Out-Projection

\[ q^f \]

\[ \partial (\mathcal{A} \oplus -\mathcal{B}) \]
Out-Projection

\( \mathbb{O} \) 

\( \mathbb{O} \) 

\( \partial (\mathcal{A} \oplus -\mathcal{B}) \)
Continuous Collision Detection (CCD)

Find the first time of contact (ToC) $\tau$
Translational CCD

Find the first time of contact (ToC) $\tau$
Translational CCD

\[ \tau = \frac{d_v(\mathcal{A}, \mathcal{B})}{\|v\|} \]

\(d_v\) : Minimal directional distance along \(v\)
Contents

• Previous work
• Preliminaries
• Out-projection
• **In-projection**
• Initialization
• Results
• Conclusion
Contact-Space Localization

\( q^0 \)  
\( q^f \)

\( \partial (\mathcal{A} \oplus -\mathcal{B}) \)
Contact-Space Localization

\[ \partial (\mathcal{A} \oplus -\mathcal{B}) \]

\[ q_0 \]

\[ 0 \]
Contact-Space Localization

\[ Jq = c \]

\[ q_0 \]

\[ \partial (A \oplus -B) \]
In-Projection

\[ \partial (\mathcal{A} \oplus -\mathcal{B}) \]

\[ \mathbf{q}_1 \]

\[ \mathbf{o} \]
In-Projection

Minimize $\|q\|^2$

subject to $Jq \geq c$

$\partial (A \oplus -B)$
In-Projection

Linear Complementarity Problem

\[
\begin{align*}
\frac{1}{4} JJ^T \lambda + c &\geq 0 \\
\lambda &\geq 0 \\
\left(\frac{1}{4} JJ^T \lambda + c\right)^T \lambda &= 0
\end{align*}
\]
In-Projection

\[ J J^T \lambda = 4c \]
In-Projection

Projected Gauss-Seidel method

\[ \lambda_{k+1} = (D + L)^{-1}(4c - U\lambda_k) \]

where \( D + L + U = JJ^T \)
Contents

• Previous work
• Preliminaries
• Out-projection
• In-projection
• Initialization
• Results
• Conclusion
Collision-Free Configuration for Out-Projection

\[ q^f \]

\[ \partial (\mathcal{A} \oplus -\mathcal{B}) \]
Initialization

• Find a collision-free configuration as close as possible to the relative origin

• Strategies:
  - Centroid difference
  - Maximally clear configuration
  - Motion coherence
  - Random configuration
  - Sampling-based search
Centroid Difference

Centroid Difference

$q^S$

$q^\mathcal{A}$

$q^\mathcal{B}$

$q^\mathcal{G}$

$o^\mathcal{A}$

$o^\mathcal{B}$
Centroid Difference

Initial collision-free

In- & out-projections

Input in-collision
Maximally Clear Configuration

- Voronoi boundary
- Approximate the maximally clear configurations
Maximally Clear Configuration

1. AABB
Maximally Clear Configuration

2. Grid positions
Maximally Clear Configuration

3. Remove the points with small distances
Maximally Clear Configuration

4. Remove the closer points along $xy$-directions.
Maximally Clear Configuration

5. Remove the grids on the boundary of AABB
Maximally Clear Configuration

6. Remove the closer points along $z$-direction
Maximally Clear Configuration
Maximally Clear Configuration

Initial collision-free

PD

Input in-collision

Initial collision-free
Sampling-Based Search

Collision-free

$\partial (\mathbf{A} \oplus -\mathbf{B})$

Collision

Collision

$q_0^f$

$q_0^f$

$q_1^f$
Maximally Clear Configurations and Sampling-Based Search

Initial collision-free

Input in-collision

PD
Maximally Clear Configurations and Sampling-Based Search

Initial collision-free

Input in-collision

PD
Motion Coherence in Dynamics

Many collision-free configurations are found.
Motion Coherence

Initial collision-free

Out-projections

Input in-collision

PD
Contents

• Previous work
• Preliminaries
• Out-projection
• In-projection
• Initialization
• Results
• Conclusion
Benchmarking Models

10K   1K   40K    1K     1.33K   174K   105K
0.54K     0.94K     152K    1.34K   2K      3K
Performance with Random Configurations
Performance with Random Configurations

top: no. of contacts
mid: no. of iterations
bot: timing
## Performance with Random Configurations

<table>
<thead>
<tr>
<th>Model</th>
<th>Time (msec)</th>
<th>No. of Contacts</th>
<th>No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Torusknot</td>
<td>5.53</td>
<td>4.84</td>
<td>2.81</td>
</tr>
<tr>
<td>Buddha</td>
<td>6.23</td>
<td>5.04</td>
<td>1.92</td>
</tr>
<tr>
<td>Bunny2</td>
<td>7.55</td>
<td>8.25</td>
<td>2.16</td>
</tr>
<tr>
<td>Dragon</td>
<td>12.76</td>
<td>11.92</td>
<td>2.29</td>
</tr>
</tbody>
</table>
Comparison to Lien’s Method
[Lien 2009]
2,000 Times Faster
top : PD for
mid : timing for
bot : PD for
## Average PD Errors

<table>
<thead>
<tr>
<th>Model</th>
<th>Mean error (%)</th>
<th>Median error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bunny1</td>
<td>0.500</td>
<td>0.165</td>
</tr>
<tr>
<td>Distorted Torus</td>
<td>1.165</td>
<td>0.066</td>
</tr>
</tbody>
</table>
Predefined Path Scenario
Dragons with Predefined Path

top : no. of contacts
mid : no. of iterations
bot : timing
## Performance on Predefined Paths

<table>
<thead>
<tr>
<th>Models</th>
<th>Time (msec)</th>
<th>No. of Contacts</th>
<th>No. of Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spoon and Cup</td>
<td>0.96</td>
<td>4.49</td>
<td>1.02</td>
</tr>
<tr>
<td>Buddha</td>
<td>3.50</td>
<td>5.07</td>
<td>1.29</td>
</tr>
<tr>
<td>Dragon</td>
<td>5.84</td>
<td>10.32</td>
<td>1.45</td>
</tr>
</tbody>
</table>
Realtime PD Results

Spoon: 1.3K triangles
Cup: 8.4K triangles
Buddha 1: 10K triangles
Buddha 2: 20K triangles
Applications
– Rigid Body Dynamics

• Multiple contacts
• To stabilize dynamics
Local PD Computation

- Computed for each contact region
- Derived from the global PD
- Corresponding to contact feature pairs that are resolved as a result of the global PD
Local PD Computation

Two objects are interpenetrated.
Local PD Computation

d: global PD
Local PD Computation

\[ n_1, n_2, n_i : \text{contact normals} \]
Local PD Computation

\[ d_i = (d \cdot n_i) n_i \]

\( d \)

\( d_1 \)

\( d_2 \)

\( n_i \)

\( i \)-th local PD:
Dynamics Simulations
## Performance in the Dynamics Scenario

<table>
<thead>
<tr>
<th>Models</th>
<th>Global PD (msec)</th>
<th>Local PD (msec)</th>
<th>Total (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Torus</td>
<td>3.13</td>
<td>1.04</td>
<td>4.17</td>
</tr>
<tr>
<td>Bunny2</td>
<td>5.43</td>
<td>1.78</td>
<td>7.21</td>
</tr>
<tr>
<td>Golf-club</td>
<td>4.87</td>
<td>1.67</td>
<td>6.54</td>
</tr>
<tr>
<td>Santa</td>
<td>9.14</td>
<td>2.90</td>
<td>12.04</td>
</tr>
</tbody>
</table>
Penalty-Based Haptic Rendering

Box and Box

Spoon and Cup
Realtime Rigid Body Dynamics

1 Plane: 12 triangles
7 Bunnies: 40K each
3 Dragons: 174K each
Total: 802K

Performance:
10~50 msec
Conclusion

• Interactive PD algorithm for complicated polygon-soup models
• Several schemes for selecting an initial collision-free configuration
• Method of computing a local PD for realistic dynamics simulation
Limitations

• Highly complex models may require a large number of iterations
• Approximates an upper bound of PD
• Depends on initial collision-free configurations
Future Work

• Extension to \( n \)-body PD problems
• Extension to generalized PD problem
• Applying to haptic rendering and robot motion planning
Acknowledgements

• Graphics group, Ewha
• Stephane Redon, INRIA
• Jyh-Ming Lien, GMU
• Anonymous reviewers
• National research foundation of Korea
• IT R&D Program of MKE/MCST in Korea
Thank You!

http://home.sogang.ac.kr/sites/je

C. Je et al.

PolyDepth: Real-time Penetration Depth Computation.

ACM Transactions on Graphics (ToG), Vol. 31, No. 1, Article 5.

Presented in SIGGRAPH 2012.