Robots and Animatronics in Disney
Lecture 2: Geometric Algorithms for Robotics

Katsu Yamane
kyamane@disneyresearch.com
Yu Zheng
yu.zheng@disneyresearch.com

Goals

• Introduce geometry-based algorithms and their application in robotics
• Convince you that they are compelling alternative to numerical algorithms

Geometric Problems in Robotics

• Grasping / locomotion
  – Feasible total contact wrench is a convex set in 6D space
  – Contact force optimization
  – Grasp quality

• Collision detection / distance computation
  – Most algorithms deal with polygon models
  – Lose global shape information
  – Large data for accurate representation
  – CAD software work with parametric surfaces (NURBS etc.)

Outline

• Contact force optimization in grasping
  – Basic concepts for geometry-based algorithms
  – Introduction to ray-shooting algorithms
• Grasp evaluation
  – Compute the largest inclusion in 6D convex set
• Distance computation
  – Generalized distance
  – Compute the minimum scaling factor for two convex objects to touch

Contact Force Optimization with Ray-Shooting

[Zheng and Yamane 2012]
Ray-Shooting in CG

Support Mapping and Function

GJK Distance Algorithm

ZC Distance Algorithm
**ZC Distance Algorithm**

\[ b \in CO(A) : d_{CO(A)}(b) = 0, v_{CO(A)}(b) = b, b \in CO(V_{CO(A)}(b)) \]

Run ZC Algorithm with \( b = r \)

\[ r \in CO(V_{CO(A)}(r')) \]

---

**ZC-Based Ray-Shooting Algorithm**

- \( Z_A(r) \) — farthest intersection point of \( A \) with \( R(r) \)
- \( Z_A(r) \) — subset of \( A \) whose convex hull contains \( Z_A(r) \)
- \( n_a \) — normal of the hyperplane that passes through \( Z_A(r) \) and supports \( A \)

---

**Internal Expanding (IE)**

- \( Z_A(r) \) — farthest intersection point of \( A \) with \( R(r) \)
- \( Z_A(r) \) — subset of \( A \) whose convex hull contains \( Z_A(r) \)
- \( n_a \) — normal of the hyperplane that passes through \( Z_A(r) \) and supports \( A \)

---

**Hybrid Use: ZC-IE**

**Contact Force Optimization**

**Contact Force Optimization: test 1**

**TEST 1: Accuracy**

- grasp a sphere against the gravity \( G \)
- four contacts have the same elevation angle \( \alpha \) on the sphere

**Ground Truth**

1. No feasible contact forces if \( \alpha \geq \tan^{-1} \mu \)
2. The minimum values are
   \[ d_{\min} = \mu \cos \alpha - \sin \alpha \]
   \[ d_{\max} = \frac{4G \cos \alpha - \sin \alpha}{G} \]

Gradually increase \( \alpha \) to the limit by setting

\[ \alpha = (1 - \lambda) \tan^{-1} \mu \] with \( \lambda = 10^{-1}, 10^{-2}, \ldots, 10^{-6} \)
**Contact Force Optimization: test 1**

<table>
<thead>
<tr>
<th>J</th>
<th>GJK-based</th>
<th>IE</th>
<th>ZC-based</th>
<th>Bi-GJK</th>
<th>ZC-IE</th>
<th>Active-Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>10^3</td>
<td>6.30 x 10^-10</td>
<td>2.90 x 10^-5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3.65 x 10^-9</td>
</tr>
<tr>
<td>10^-2</td>
<td>1.55 x 10^-14</td>
<td>1.98 x 10^-9</td>
<td>0</td>
<td>0</td>
<td>1.63 x 10^-8</td>
<td></td>
</tr>
<tr>
<td>10^-3</td>
<td>1.38 x 10^-15</td>
<td>6.88 x 10^-10</td>
<td>0</td>
<td>1.63 x 10^-8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10^-4</td>
<td>6.82 x 10^-15</td>
<td>6.89 x 10^-10</td>
<td>6.89 x 10^-10</td>
<td>3.92 x 10^-9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10^-5</td>
<td>6.90 x 10^-15</td>
<td>infeasible</td>
<td>infeasible</td>
<td>infeasible</td>
<td>infeasible</td>
<td>infeasible</td>
</tr>
</tbody>
</table>

**Contact Force Optimization: test 2**

Computing the minimum contact forces for 1000 random $\mathbf{w}_{\text{res}}$

<table>
<thead>
<tr>
<th>J</th>
<th>GJK-based</th>
<th>IE</th>
<th>ZC-based</th>
<th>Bi-GJK</th>
<th>ZC-IE</th>
<th>Active-Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>10^3</td>
<td>5.00 x 10^-10</td>
<td>6.30 x 10^-5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3.65 x 10^-9</td>
</tr>
<tr>
<td>10^-2</td>
<td>1.38 x 10^-14</td>
<td>5.39 x 10^-9</td>
<td>0</td>
<td>0</td>
<td>1.63 x 10^-8</td>
<td></td>
</tr>
<tr>
<td>10^-3</td>
<td>1.38 x 10^-14</td>
<td>1.38 x 10^-10</td>
<td>1.38 x 10^-10</td>
<td>3.92 x 10^-9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10^-4</td>
<td>5.39 x 10^-14</td>
<td>0</td>
<td>0</td>
<td>9.65 x 10^-10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10^-5</td>
<td>1.38 x 10^-14</td>
<td>infeasible</td>
<td>infeasible</td>
<td>infeasible</td>
<td>infeasible</td>
<td>infeasible</td>
</tr>
<tr>
<td>10^-6</td>
<td>1.50 x 10^-14</td>
<td>infeasible</td>
<td>infeasible</td>
<td>infeasible</td>
<td>infeasible</td>
<td>infeasible</td>
</tr>
</tbody>
</table>

**Summary**

- Four geometry-based algorithms for solving the ray-shooting problem
  - GJK-based
  - ZC-based
  - Internal Expanding (IE)
  - ZC-IE hybrid

- Efficient and accurate contact force optimization by ray-shooting and geometry-based algorithms

**Grasp Evaluation**

[Zheng and Yamane ICRA 2013]

**Grasp Quality Measures**

- Singular values of the grasp matrix [Li and Saffey, TRA88] does not consider friction constraint and force closure
- Geometry of contact polygons [Chiesa et al.; Ponce et al.] lacks clear physical meanings
- Size of force-closure contact regions [Nguyen, IJRR88] evaluates the safety margin of a force closure
- **Grasp force efficiency (GFE):** worst-case maximum wrench generated by unit contact forces [Kirkpatrick et al., DCG92; Ferrari and Canny, ICRA92] quantitative evaluation of a grasp beyond force closure Based solely on contact locations

**Fair Evaluation of a Grasp?**

Ability to exert contact forces also depends on
- Hand configuration
- Passive joints and joint coupling
Grasp Force Efficiency (GFE)

Grasp wrench set \( W \): consists of all resultant wrenches generated by unit contact forces.

Convex hull \( \sum \)

Minkowski sum \( \oplus \)

Example
Soft-finger contacts
Joints 3 and 4 (blue) coupled
Grasp wrench set from Minkowski sum

GFE with Same Contact Locations
Case 1
\( r = 0.5624 \)
Case 2
\( r = 0.5624 \)
Case 3
\( r = 0.5624 \)

Active Grasp Force Efficiency (A-GFE)

Finger Jacobian: \( \tau_f = J^T f \)
Active contact force: \( f_a = J^T \tau \)

Active grasp wrench set \( W_a \): consists of all resultant wrenches generated by unit active contact forces

How to Compute (A-)GFE?

- Problem: local minima
- Existing algorithms for GFE: approximate the friction cones by polygonal pyramids
- New algorithm
  - Finds the generalized penetration distance
  - Global optimum without friction cone approximation
  - Applicable to (A-)GFE with convex hull and Minkowski sum
  - Requires only linear algebra calculations
Algorithm

Calculate the largest inscribed set in a polytope

Calculate the minimum of $d_1/d_2$ over all facets

Numerical Results

Test 1:
- two fingers
- joint 1 (red) is locked

Result:
- $W$ is 6-D, but $W^*$ is 3-D

Test 2:
- four fingers
- joint 1 (red) is locked

Result:
- $W$ is 6-D, but $W^*$ is 5-D

Test 3:
- no joint is locked

Result:
- both $W$ and $W^*$ are 6-D
- $r = 0.5624$, but $r^* = 0.3693$

Test 4: same contact locations, different hand configurations

Numerical Results

$r = 0.5624$

Numerical Results

$r^* = 0.1536$

$r^* = 0.0395$

$r^* = 0.3693$
Summary

• Proposed the Active Grasp Force Efficiency (A-GFE) measure that considers hand mechanism/configuration

• Developed an accurate and efficient algorithm for computing the generalized penetration distance, including the (A-)GFE measure

Generalized Distance

[Zheng and Yamane ICRA 2013]

Distance Functions

• Euclidean distance
  → Separation case [Gilbert et al., TRA88,90; Lin and Canny, ICRA91]
  → Penetration case [Zheng and Yamane, ICRA13]

• Generalized distance [Zhu et al., TRA03,04]
  → Separation [this paper]
  → Penetration [Zheng and Yamane, ICRA13]

• Growth distance [Zhang and Gilbert, TRA96; Zheng and Yamane, WAFR12]

• Distance in configuration space [LaValle 2006, Zhang et al., SPM06]

Definitions of Distance

Given: \( A \) — a compact convex set not containing the origin \( \emptyset \)

Euclidean distance between \( \emptyset \) and \( A \)

\[ dB \]

\( \emptyset \): unit ball

\( d \): minimum scaling factor such that \( dB \) and \( A \) are not disjoint

Separation [Gilbert et al., TRA88,90; Lin and Canny, ICRA91]

Penetration [Zheng and Yamane, ICRA13]

Generalized distance between convex objects

Given: \( A \) — a compact convex set not containing the origin \( \emptyset \)

Generalized distance between \( \emptyset \) and \( A \)

\( Q \): a “guage set”

\( d \): minimum scaling factor such that \( dQ \) and \( A \) are not disjoint

Generalized distance [Zhu et al., TRA03,04]

Separation [this paper]

Penetration [Zheng and Yamane, ICRA13]

\( d \geq 1 \): separation

\( d < 1 \): penetration
Application in Motion Planning

- More accurate sensors
- Less accurate sensors

- $A$ — an obstacle
- $Q$ — a robot

Algorithm Overview

$A(\sigma) = a_0 + \sigma(A - a_0)
$
Scale $A$ by factor $\sigma$ around $a_0$

Computing $\sigma_K$ and $n_K$

$A(\sigma) = a_0 + \sigma(A - a_0)$

1) $z_k \rightarrow \sigma_k = \frac{\|z_k\|}{\|z_k\|}$

2) $Z_k$: set of points that represent $z_k$ as their convex combination
$\rightarrow n_k$: normal of the supporting hyperplane
Accuracy and Speed

Red — A  Blue — Q  Green — dQ

- contact $d = 1$
- separate $d = 2$
- penetrate $d = 0.5$

3000 random pairs for each case

Error

The error of our algorithm is linear with the termination tolerance.
Our algorithm is as accurate as the active-set algorithm by MATLAB.

CPU Running Time

Our algorithm is faster than the active-set algorithm by MATLAB.

n-Dimensional Polytopes

A and Q are n-dimensional polytopes with $N$ vertices each on a unit ball

200 vertices for each 1000 vertices for each

3-D illustration

CPU Running Time

The CPU running time increases slowly along with the space dimension and the number of vertices.

Comparison with the Simplex Method

Our algorithm is much faster than the Simplex method by MATLAB especially in low-dimensional space with a large number of points.
Summary

• Generalized distance: minimum scaling factor for two convex objects to touch
• A geometry-based algorithm for efficient computation of the generalized distance

Discussion

• Advantages of geometry-based algorithms
  – Global optimum
  – Fast
  – Consists of a few basic geometric computation
  – Accuracy comparable to standard numerical optimization
• Applications
  – Planning
  – Contact simulation
  – Grasp analysis