Fast Medial Axis Approximation via Maximum Margin Pushing

Guilin Liu and Jyh-Ming Lien

George Mason University, USA
Medial Axis in Motion Planning

- Medial-axis provides higher safety for robot by maximizing its clearance
Medial Axis in Motion Planning

• In sampling-based motion planning, medial-axis provides provably higher possibility in solving narrow passage problem [Wilmarth et al. ICAR 1999, Lien et al., ICRA 2003, Yeh et al. ICRA 2014]
Problems

- Only samples are on the MA, but connections are NOT
  - Many samples are required
- In high (>3) dimensional space, computation of clearance and penetration depth is nontrivial
  - Following $N$ random directions until collision status changes

parabolic narrow passage
Overview of Our Solution

Generate training data

Train classifiers

1 vs 1 and 1 vs many SVM classifiers

Random sampling

Push
The Max-Margin Problem

- Medial Axis: has $\geq 2$ closest points
- SVM: find a linear hyperplane to separate positive from negative
  - Hyperplane has same distance to positive and negative support vectors
  - Key idea: We model the medial axis using SVM. But beyond the simple SVM!

Mathematical formulation:

$$\min_{\gamma, w, b} \quad \frac{1}{2}||w||^2$$

subject to

$$y^{(i)}(w^T x^{(i)} + b) \geq 1, \quad i = 1, \ldots, m$$
Obtain Training Data

Configurations from the contact space (mostly)

\(S_P\) is a set of points sampled from \(P\)

\(S_Q\) is a set of points sampled from \(Q\)

Replacing each point of \(S_Q\) using \(S_P\)
Modeling Medial Axis: Pushing

• Classifier (Decision Function)

\[ f(x) = \sum_i \alpha_i \Phi(x_i) \cdot \Phi(x) + b \]

\[ = \sum_i \alpha_i K(x_i, x) + b \]

• \( K \) is the kernel function; \( \alpha \) is coefficients for each Support Vector

\[ |f(x)|: \text{the points on the hyperplane (medial axis) have the minimum value of } |f(x)| = 0 \]
Modeling Medial Axis: Pushing

- SVM classification function:
  - \( f(x) = W^T x + b \)
- Pushing direction: \( f'(x) \)
- Pushing step size: \( f(x) \)

\[
x = x_0 + \gamma \frac{w}{|w|} = x_0 + \frac{w * (w^T + b)}{|w|^2}
\]

\[
x = x_0 + \frac{f'(x) * f(x)}{|f(x)|^2}
\]

\[
x_{n+1} = x_n - \frac{f'(x) * f(x)}{|f(x)|^2}
\]

This is basically a gradient decent *with step size automatically determined*.
Modeling Medial Axis: Pushing

- SVM classification function:
  \[ f(x) = W^T x + b \]
- Pushing direction: \( f'(x) \)
- Pushing step size: \( f(x) \)

changing of pushing step size
Handle More than Two Objects

One-vs.-One classifier

One vs. Others classifier
Combined Classifier

• A two-step combined classifier
  1. Use one-vs.-others classifier to find two closest objects
  2. Use one-vs.-one of these two objects for pushing
Pushing Procedure

**Data:** a figuration \( x \), stopping threshold \( \epsilon \)
pairwise one v.s. one classifier and one v.s. others classifiers

**Result:** updated on the approximate medial axis

**While** \( f_{st}(x) > \epsilon \) **do**

\[
x = x - \frac{f'_{st}(x) \cdot f_{st}(x)}{|f_{st}(x)|^2}
\]

s and t satisfying that \( g_s \geq g_t \geq g_i, \forall i \in \{1..n\} \setminus \{s, t\} \)

**End**
Results

2D

6D

3D
## Comparison

<table>
<thead>
<tr>
<th>Env</th>
<th>ObN</th>
<th>Method</th>
<th>Number</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>MKSum</td>
<td>Train</td>
</tr>
<tr>
<td>2D</td>
<td>354</td>
<td>MAPRM</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVMA</td>
<td>70800</td>
<td>70800</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3D</td>
<td>148</td>
<td>MAPRM</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVMA</td>
<td>69469</td>
<td>59200</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6D</td>
<td>2</td>
<td>MAPRM</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVMA</td>
<td>27478</td>
<td>3000</td>
</tr>
</tbody>
</table>
Conclusion

• Our new approach addressed two major challenges in Medial-Axis based probabilistic motion planners
  – Avoid expensive clearance & penetration depth in high dimensional C-space
  – Allow connection on the medial axis

• Limitations
  – Requires many classifiers n(n-1)/2 + n. (n(n-1)/2 for 1 v.s. 1)--------(possible to improve if compute 1 v.s. 1 for neighboring obstacles)
  - Building SVM/Max-Margin classifier is base on optimizing the object function.
    slack variable used when not total separatable, which maximize the objective function, but at the same time hurts the quality of MA