Robot Navigation in Densely Populated Environments

Anirudh Vemula
Robotics Institute
Carnegie Mellon University

January 26, 2017
Research Focus

Enabling robots to navigate dense human crowds in a safe and predictable fashion

Advisors: Dr. Katharina Muelling and Dr. Jean Oh
Outline of the talk

Problem Statement

Navigation in Fully-observable dynamic environments
  Past Work
  Adaptive Dimensionality in dynamic environments

Navigation in Partially-observable dynamic environments
  Past Work
  Modeling cooperative navigation in human crowds

Future directions
  Improving prediction accuracy
  Safety

Conclusion
Problem Statement

Given a start position $S$ and a goal location $G$ in a densely populated dynamic environment, find a safe, feasible and socially-compliant path.
Problem Statement

Navigation in Fully-observable dynamic environments

Past Work
Adaptive Dimensionality in dynamic environments

Navigation in Partially-observable dynamic environments

Past Work
Modeling cooperative navigation in human crowds

Future directions
Improving prediction accuracy
Safety

Conclusion
Assumptions and Goals: Fully-observable environments

**Assumptions**

- Trajectories of obstacles known

**Goals**

- Quick planning times
Planning in Dynamic Environments

Figure: Planning without time dimension

Figure: SIPP

Figure: RRT
Motivation

Figure: Counter-intuitive heuristic in the presence of dynamic obstacle
AD graph construction

**Core Idea**: Consider time dimension only in regions where there is a potential dynamic obstacle collision. Plan in low-dimension elsewhere.

---

**Figure**: Adaptive dimensionality graph, [2]

**Figure**: High-D region

Anirudh Vemula
Main Loop

PLANNING PHASE
Search in current AD graph to get path $p$

- If no path $p$ is found, no feasible path exists. Exit
- Construct high-D tunnel around $p$

TRACKING PHASE
Search in the tunnel around $p$ to get path $t$

- If no path $t$ is found, introduce/expand high-D region
- If $\text{cost}(t) < \epsilon_2 \text{cost}(p)$, return $t$. Else, above.
Theoretical properties

- **Completeness**: If a path exists, the algorithm will find it.
- **Termination**: Will terminate after finite iterations.
- **Bounded cost suboptimality**: If search is done using weighted A* with inflation $\epsilon_1$, then cost of resulting path is no more than $\epsilon_1 \cdot \epsilon_2$ times the optimal cost.
Results: 4D \((x, y, \theta, t)\) planning

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Number of Success (Out of 50)</th>
<th>Epsilon</th>
<th>time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>mean</td>
<td>std dev</td>
</tr>
<tr>
<td>Adaptive</td>
<td>40</td>
<td>1.1</td>
<td>6.7 0.8</td>
</tr>
<tr>
<td>4D</td>
<td>5</td>
<td>1.1</td>
<td>91.0 71.2</td>
</tr>
<tr>
<td>Adaptive</td>
<td>41</td>
<td>1.5</td>
<td>11.7 14.0</td>
</tr>
<tr>
<td>4D</td>
<td>21</td>
<td>1.5</td>
<td>70.3 86.7</td>
</tr>
<tr>
<td>Adaptive</td>
<td>46</td>
<td>2.0</td>
<td>18.5 26.6</td>
</tr>
<tr>
<td>4D</td>
<td>23</td>
<td>2.0</td>
<td>35.8 69.8</td>
</tr>
</tbody>
</table>

Table: Results on 50 indoor environments with 30 dynamic obstacles.

Anirudh Vemula, Katharina Muelling, Jean Oh. Path Planning in Dynamic Environments with Adaptive Dimensionality. SoCS 2016
Problem Statement

Navigation in Fully-observable dynamic environments
Past Work
Adaptive Dimensionality in dynamic environments

Navigation in Partially-observable dynamic environments
Past Work
Modeling cooperative navigation in human crowds

Future directions
Improving prediction accuracy
Safety

Conclusion
Navigation in dense human crowds

Accurately predict the future trajectories of humans in the crowd and adapt its own trajectory in a human predictable manner.
Challenges: Freezing robot problem

Predictions need to:

- be accurate
- account for interactions, [3]

Jointly predict trajectories of all agents ✓
Navigation in human crowds: Past Work

Figure: Interacting Gaussian Processes

Figure: Inverse Reinforcement Learning
Modeling Joint distribution

Independent prediction model

\[ P(f_R, f_1, \cdots, f_n) = \prod_{i=R}^{n} P(f_i) \]

Interacting Gaussian processes, [3]

\[ P(f_R, f_1, \cdots, f_n) = \frac{1}{Z} \psi(f_R, f_1, \cdots, f_n) \prod_{i=R}^{n} P(f_i) \]

Using occupancy grids

\[ P(f_R, f_1, \cdots, f_n) = \prod_{i=R}^{n} P(f_i | O_i, g_i) \]
Data-driven solution using occupancy grids

Agent considered

Surrounding agent

Count of agents occupying this cell

Goal (G)

Occupancy grid (O)

Learn $P(v|O, G)$ from training data
Inferring goal and multi-step prediction

\[ G = \arg \max_G \prod_i P(v_i | O_i, G) \]

Use the learned model to calculate the above likelihood

Predict velocities and compute occupancy grids at each time-step
Results

Figure: Collision avoidance

Figure: Cooperative behavior

Anirudh Vemula, Katharina Muelling, Jean Oh. *Modeling Cooperative Navigation in Dense Human Crowds.*

ICRA 2017
Problem Statement

Navigation in Fully-observable dynamic environments
Past Work
Adaptive Dimensionality in dynamic environments

Navigation in Partially-observable dynamic environments
Past Work
Modeling cooperative navigation in human crowds

Future directions

Improving prediction accuracy
Safety

Conclusion
Deep recurrent models: Accuracy

Each human trajectory modeled using RNN

Hidden state $h$ is a latent representation of individual behavior

How do you model interactions using these latent representations?
Model uncertainty : Safety

▶ “Importance of knowing what you don’t know”, [1]
▶ Example: Execute safety maneuver when model confidence in its own prediction is low

How do you obtain model uncertainty from current state-of-the-art learning models without losing scalability and accuracy?
Navigating a robot through a dense crowd in a **safe** and **socially-compliant** way requires **modeling interactions** accurately and accounting for **uncertainty** in its own predictions.

Questions?
References I

