A Comparison of Two Algorithms for Robot Learning from Demonstration

Halit Bener Suay, Sonia Chernova
{benersuay, soniac} @ wpi.edu
Introduction

Learning from Demonstration

• Interactive Reinforcement Learning
• Behavior Networks
Purpose

- First empirical comparison of these algorithms.
- Shed light on the advantages and disadvantages of two Learning from Demonstration (LfD) algorithms from a researcher's point of view.
- Help researchers understand and explore these algorithms.
- Make Interactive Reinforcement Learning and Behavior Networks code available to robotics community for future implementations.
Interactive RL

A Comparison of Two Algorithms for Robot Learning from Demonstration
H. Ben Suay – Sonia Chernova
{benersuay, soniac} @ wpi.edu

Positive Reward
Guidance
Negative Reward
Behavior Networks

I'm holding a mug with my left hand

picked up the green bottle in the kitchen
Behavior Networks

BaseLoc: Pos1

BaseLoc: Pos2

Right Hand Obj: Bottle

A Comparison of Two Algorithms for Robot Learning from Demonstration

H. Ben Suay – Sonia Chernova
{benersuay, soniac} @ wpi.edu
Behavior Networks

A Comparison of Two Algorithms for Robot Learning from Demonstration

H. Ben Suay – Sonia Chernova
{benersuay, soniac} @ wpi.edu
Related Work


Interactive RL

Start

- Initialize the Q-Table

Choose an action (Epsilon Greedy) considering the guidance message

- Guidance?
  - Y: Choose an action (Epsilon Greedy)
  - N: Execute the action

- Execute the action

- Reward?
  - Y: Update the Q-Table
  - N: Reward = 0.0

- Task Accomplished?
  - Y: Stop
  - N: Task Accomplished?

Behavior Networks

Start

- Initialize the time-table

Wait until the next action request

- Execute the requested action

- Post-conditions met?
  - Y: Add the behavior(s) to the time-table
  - N: Task accomplished?

- Task accomplished?
  - Y: Stop
  - N: Task accomplished?
Experimental Task: Object Sorting
Experimental Setup
Action Space

<table>
<thead>
<tr>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPUp</td>
</tr>
<tr>
<td>LPDown</td>
</tr>
<tr>
<td>LLeft</td>
</tr>
<tr>
<td>LDrop</td>
</tr>
<tr>
<td>LRight</td>
</tr>
<tr>
<td>RPup</td>
</tr>
<tr>
<td>RPDown</td>
</tr>
<tr>
<td>RLeft</td>
</tr>
<tr>
<td>RDrop</td>
</tr>
<tr>
<td>RRight</td>
</tr>
<tr>
<td>TakePicture</td>
</tr>
</tbody>
</table>
State Space

<table>
<thead>
<tr>
<th>Small State Space (360 possible states)</th>
<th>Large State Space (6480 possible states)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Number of SURF*</td>
<td>• Number of SURF*</td>
</tr>
<tr>
<td>• Robot's hands' state and object location</td>
<td>• Entropy</td>
</tr>
<tr>
<td></td>
<td>• Smoothness</td>
</tr>
<tr>
<td></td>
<td>• Area of the bounding box</td>
</tr>
<tr>
<td></td>
<td>• Robot's hands' state, and object location</td>
</tr>
</tbody>
</table>

*Speeded Up Robust Features
Results
Time, Rewards, Guidance

Behavior Networks
Small State Space
5.7 min.

Behavior Networks
Large State Space
8.7 min.
Results

Behavior Networks

Small State Space

Large State Space
Behavior Networks

+ Robot doesn't make any mistake during learning.
+ Teaching is much faster.
+ Modification and correction after teaching the task is easier.
- Limited with what the teacher teaches (no exploration).
- Requires lots of domain knowledge.

Interactive Reinforcement Learning

+ Simpler to implement and to teach.
+ Automatic action selection.
+ Exploration.
- Teaching time increases remarkably as the state space grows.
- Passed states (or history) is not important when choosing an action.

Source-code coming very soon for ROS:
https://github.com/WPI-RAIL/wpi-rail-ros-pkg
Future Work

• **Machine Learning:**
  • Learning in continuous State Space.
  • Semi / fully autonomous state variable selection.
  • Comparison with other learning algorithms (learning time, performance, knowledge req. for teaching, for coding etc.).

• **User study:**
  • How can various user interfaces be implemented?
  • Which interface is most intuitive? For who?
  • What people think about different aspects of the teaching experience (such as speed, accuracy, performance etc.)?