Lecture 04/05/2017

Karthik Ramachandran, Jin Zhu

Reinforcement Learning

Reinforcement learning in a nutshell is to learn from trial and error. The standard way to approach reinforcement learning problems is to use bellman functions to maximize reward based on policy using dynamic programming.

Reinforcement Learning has a few challenges as listed below:

1. The standard approach is computationally expensive due to curse of dimensionality.
2. No accurate models of the environment is available.
3. Defining the reward function is extremely hard.

Some ways of solving the above problems are listed below:

1. Use learning approaches to learn policy based on reward.
2. Use compact state-action estimation to reduce the computation cost.
3. Restructure the task to make it computationally tractable.
4. Use prior knowledge to reduce the uncertainty and the computation graph

Correspondence Problem

The correspondence problem is related to the fact that the robot might not be able to observe everything in its environment and the morphology of the robot could be very different from the source of learning and the robot might not be able to imitate the source.

There are a few ways to solve the correspondence problem listed below:

1. One way to ensure reliable imitation is to hold the robot arm and move it which is called kinesthetic teach-in. (Though there might be the problem that, on its own robot might not be able to create the same kind of accelerations).
2. The robot could rely on a visual system to imitate the source.
3. Tele-operation could be another option to teach the robot to imitate.
4. Verbal instruction could be yet another medium to teach the robot.

Imitation Learning

Imitation learning involves moving from hand engineered solutions to autonomous learning systems. Learning from demonstration is essential for building adaptive systems. Formally,

imitation learning can be defined as follows: Given a set of labeled training data, learn a function that maps observed state to an action.
Key question to be answered for imitation learning is what to imitate

1. Object target
2. Object movement
3. Force interaction
4. End-effector movement
5. Joint configuration over-time

Another key question is how to demonstrate:
1. Observing a teacher e.g., with camera, kinect, motion capture
2.Teleoperating a robot, e.g., with joystick
3. Kinesthetic teach-in
4. Verbal instruction

Case Study Robot Playing Table Tennis

What makes it an interesting case study?

1. Uncertainty about the ball state
2. Adaptation to new targets
3. Different time critical hitting movements

How can we learn a manipulation tasks?
1) Learn Motor Policies

Step 1: Representation and Initialization of Behavior
Movement as state-action pairs?
- Limits us to one specific movement (not adaptable)
- Stuck with policy

Learning what the movement optimizes (reward function)?
- Unclear if we can apply this on a robot

Need a representation that allows us to encode the movement depending on the goal such that we can adapt it to new goals and length
Step 2: Dynamic System Motor Primitives
General Idea: Encode the shape of the movement
Motor policy representation that performs an automatic mapping of states to actions over time

Types:
Canonical system – action synchronization
Transformed system – desired state

Advantages:
Arbitrarily shaped smooth movements
It is possible to include object avoidance
Simple to adapt
Stable and robust

Linear in parameters w
straightforward to learn by imitation learning
well suited for reinforcement learning

How can we learn a manipulation tasks?

Step 1: Representation and Initialization of Behavior
Step 2: Learning from demonstration
Step 3: Selecting and Generalization of Behavior

Initialization of behavior

• Starting point for robot learning
• Goal: reproduce the demonstrated behavior
• Record a movement from demonstration
• Parameter can be estimated with Locally Weighted Regression (LWR)

Why is this not enough?

• Variety in hitting points
• Variety in approaching directions

Limitations for Imitation Learning

Generalization of DMP is limited
Undemonstrated situations
Quality of demonstrations
Rigid policy

Overcoming the limitations:

Collect demonstrations for different situations
Learn mapping from situation to primitive
Generate new skills from existing ones

Evaluation on the real system

Recorded 25 hitting motions on the real system
High variation in their overall performance
Using the MoMP, we could reduce the performance error •
Increased performance with increasing number of MPs