Learning Using Behavioral Primitives
Darrin Bentivegna (GT, ATR)
Gordon Cheng (ATR)
Chris Atkeson (CMU, ATR)

Points I want to make
- Personal robot needs “natural” end user programming.
- Memory-based approach makes learning to choose policy primitives from observation easy.
- Memory-based approach can learn from practice (reinforcement learning) by scaling distance function by value of using a memory.
- Alternative of using policy parameters has slow learning from observation.

NSF ITR: Using Humanoids to Understand Humans

Task 1: Air Hockey

Task 2: Marble Maze

Policy Primitives
- Behavior indexed by state
- Learn from observation and from practice
- Learn to select policy
- Learn to generate subgoals
- Fixed action generation
- Existing library of policy primitives
Framework

Primitive Policies in Air Hockey
- Right Bank Shot
- Straight Shot
- Left Bank Shot

Learn from Observation

Primitive Policies in Marble Maze
- Roll To Corner
- Roll Off Wall
- Leave Corner
- Free Roll
- Roll From Wall

Learn Models from Observation

Raw Data

Observed policy primitives

Wall Contact Labeled

Parsing Marble Maze Data
Learning From Observation

- Memory-based learner: learn by storing and recalling specific observations
- Policy selection: k-nearest neighbor
- Sub-goal generation: kernel regression (distance weighted averaging) based on remembered primitives of appropriate type
- Fixed action generation

Learning from experience

- Need task specification as reward function
- Learn by adjusting distance to query: scale distance function by value of using a memory
  \[ d'(x,q) = d(x,q) \times f(x,q) \]
- \( f \) (experience location, query location) related to Q value: \( 1/Q \) or \( \exp(-Q) \)
- Implementation 1: lookup tables at each experience
- Implementation 2: locally weighted projection regression (LWPR)

Video
Conclusions

- Memory-based approach makes learning to choose policy primitives from observation easy.
- Memory-based approach can learn from practice (reinforcement learning) by scaling distance function by value of using a memory.

Distance Metric Matters

- Not Euclidean (other side of wall)
- Need to take into account time (goal achievement)
- When and what to forget
- What matters (symmetry? Range of generalization)

What properties should a (set of) primitive(s) have?

- Recognizable, with recognizable subgoals. Segmentable?
- Support (fast) learning
- Separate data into smooth subsets
- Maximize data efficiency (right level of generalization)
Biology: Primitives?

- Eye movements: Saccades, smooth pursuit, OKN, VOR, vergence
- Gaits: walk, run, trot, canter, gallop, pace, prance, rack, jump, limp, hop, skip
- Arm movements: bell shaped velocity profiles; minimum jerk, minimum torque change, minimum error
- Hand movements: ?

Robotics/AI: Primitives?

- Macro actions, operators (STRIPS)
- Schemas (psychology)
- Behavior-based X
- Options (reinforcement learning)

Creating Primitives

- Manually generate (this talk)
- Observe (cluster, PCA, bottlenecks, failures)
- Make many plans, then observe
- Have standard library/generate, and test

Using Primitives While Performing the Task

Roll Off Wall > Free Roll > Roll To Corner