Hierarchical Reinforcement Learning

Aijun Bai

Multi-Agent Systems Lab., USTC

Oct 10, 2011

Outline

Flat Reinforcement Learning Methods

- Suffer from the curse of dimensionality:
  - comupational complexity grows exponentially with the number of vaiables used to describe a state
- Ignore the internal structure of:
  - domain knowledge
  - problem itself

  leading to slow learning
- Closely concern about policies and value function which are hard to transfer for sharing and reusing

Hierarchical Reinforcement Learning Methods

Key Ideas:

- Impose constraints on the value function/policy
  - **State abstraction:** symmetries, state aggregation
  - **Temporal abstraction:** options, task hierarchies
- Develope learning algorithms that expoit these constraints

Purpose of Hierarchical RL:

- **Speed-up:** Leverage the structure within domain/problem
- **Scale-up:** Decompose large problems into smaller ones
- **Transfer:** Use hierarchical structure as pirior knowledge for similar new problems

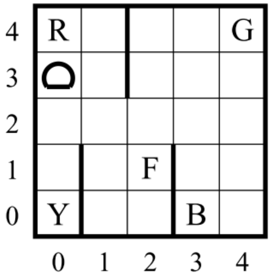
The Taxi Domain

States: $25 \times 14 \times 5 \times 5 = 8750$

- Taxi location: (x, y)
- Taxi fuel amount: $[0, 13]$
- Passenger location: R, Y, B, G and In
- Destination location: R, Y, B, G, F

Actions: 7

- North, South, East, West
- Pickup, Putdown
- Fillup



Problem size: $8750 \times 7 = 61250$

The Taxi Hierarchy

Policy structure

- Get: move to the passenger and pickup
- Put: move to the destination and putdown
- Refuel: move to F to fillup if needed

Elementary hierarchical method

- Solve all possible **Navigation**: $25 \times 5 \times 4 = 500$
- Solve **Get**, **Put** and **Refuel** using **Navigation**: $25 \times (4 + 4 + 1) \times (5 + 1) = 1350$
- Determine among **Get**, **Put** and **Refuel** in top level: $25 \times 14 \times 5 \times 5 \times 3 = 26250$

Semi-Markov Decision Process

SMDP is a generalization of MDP:

- Action selections are made at the *controlled* states
- Duration between controlled states is a random variable

Transition function becomes:

- $T : S \times A \times S \times N \to [0, 1]$

The Bellman equation become:

- $V^*(s) = \max_a [R(s, a) + \sum_{s', d} \gamma^d T(s', d|s, a) V^*(s')]$
- $Q^*(s, a) = R(s, a) + \sum_{s', d} \gamma^d T(s', d|s, a) \max_{a'} Q^*(s', a')$

RL methods correspondingly extend to SMDPs.

Options are temporally-extened actions, defined as $O = (I, \pi, \beta)$:

- $I \subseteq S$ is the initiation set of states
- π is this option's own policy
- $\beta(s)$ is the probability of termination in state s

Options are also called as *skills, behaviors, macro-actions*, etc.

Hierarchical RL using options:

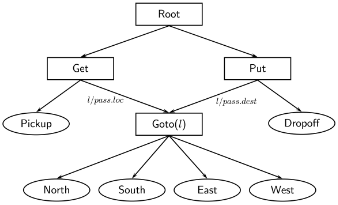
- MDPs augmented with predefined options become SMDPs
- The optimal policy over options can be learned using SMDP RL methods

The MAXQ Hierarchy

MAXQ hierarchy is orgnized in a *task hierarchies* $H : \{T_i : 0 \leq i \leq m\}$ represented as a DAG.

Decompose value function

- $V^i(s) = R(s, i)$
  Immediate reward of promitive T_i
- $V^i(s) = \max_a Q^i(s, a)$
  Total expected reward in task T_i
- $Q^i(s, a) = V^a(s) + C^i(s, a)$
  Reward if sub-task T_a selected
- $C^i(s, a) = E_{d,s'}[\gamma^d V^i(s')]$
  Expected reward after T_a finished



The MAXQ-Q Learning Algorithm

Overview of MAXQ-Q:

- Learn V for each primitive action
- Learn C for each composite action
- Use Q-learning-like update rules

Properites of MAXQ-Q

- Facilitates state abstraction:
  Different representation for each C, V
- Learning procceeds bottom-up
- Convergence with hierarchical optimality guarantee

Automatic Hierarchy Discovery Approches

Automatically inducing options or task hierarchies:

- **Bottlenecks in S:** connect two or more strongly-connected regions
- **Exit conditions:** satisfy the preconditions of actions
- **Factored representation**: analyze DBN models of factored MDP

Future Work

- Characterization of hierarchical structure
- Automatic discovery of hierarchical structure
- Transfer of hierarchical structure

References

- N. Mehta, "Hierarchical Structure Discovery in Reinforcement Learning," 2009.
- N.K. Jong and P. Stone, "Hierarchical Model-Based Reinforcement Learning : R- MAX + MAXQ," International Conference on Machine Learning, 2008.
- T.G. Dietterich, "Hierarchical Reinforcement Learning", Slides, 2000.
- T.G. Dietterich, "Hierarchical Reinforcement Learning with the MAXQ Value Function Decomposition," Journal of Machine Learning Research, 1999.