Learning Options through Human Interaction

Kaushik Subramanian, Charles Isbell, Andrea Thomaz

Speaker - Luis Carlos Cobo Rus

Link - http://dl.dropbox.com/u/2110300/IJCAI_KCAL.pdf

Georgia Tech College of Computing
School of Interactive Computing

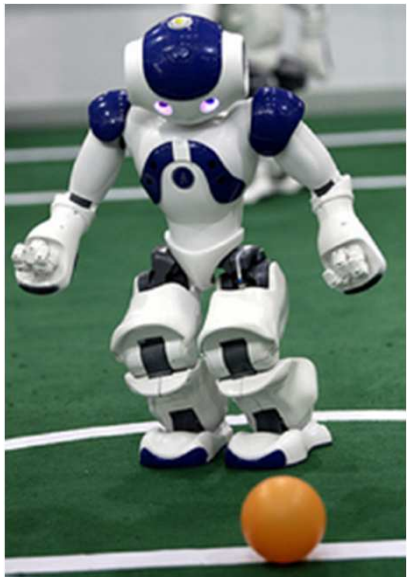
Outline

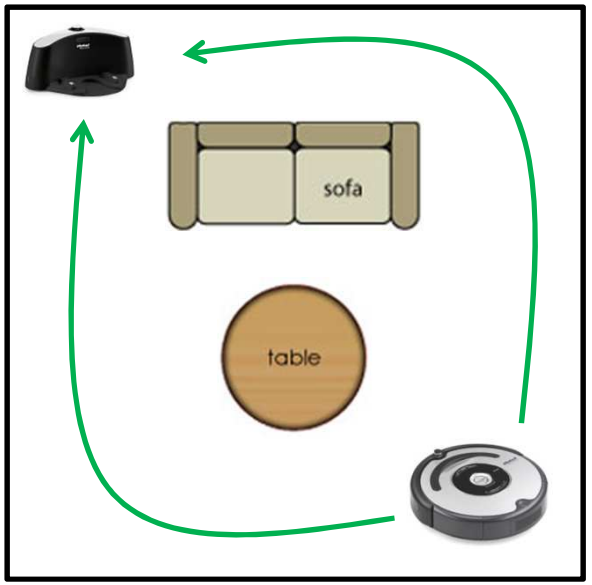
- Options Framework
- Motivation
- Interaction and Learning Phase
- Experiments
- Insights
- Results and Conclusions

Options Framework

- Hierarchical Reinforcement Learning – solves problems by decomposing it into hierarchies
- Option – temporally extended sequence of actions
- Components
  - Initiation Set, I
  - Termination Set, $\beta(s)$
  - Policy, $\pi(s,a)$

Sample Options





- Hit the question mark
- Move to the ball and kick it
- Navigate to the charger

Temporally Extended Actions

Motivation

- Time consuming to be hand-defined by humans.
- Automatically learning such options is also hard.
- Can we learn when to interrupt an option?
- Would like options to generalizable well across the state space.

Using Human Help

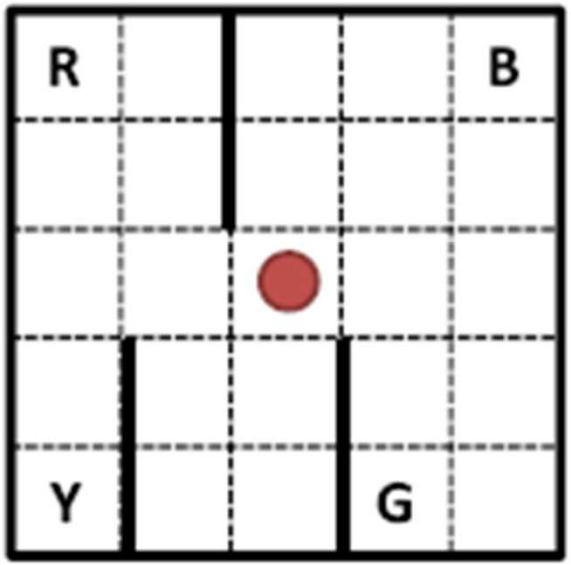
- Is the way humans decompose problems consistent with the options framework?
- Can humans teach options that lead to performance gains?
- What interesting properties do the human-defined options exhibit?

Experiment with Humans

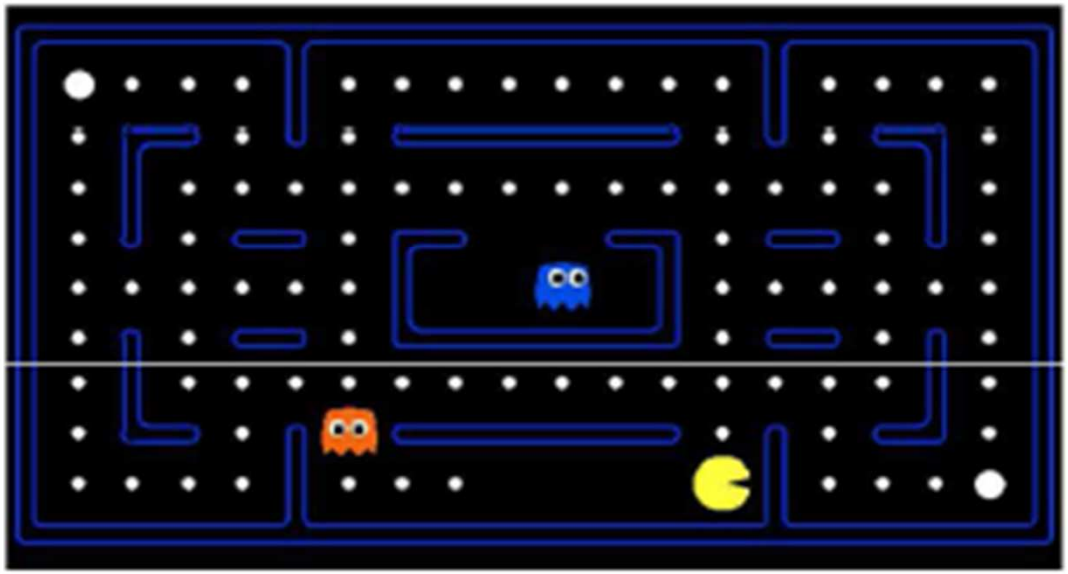
1. 10 volunteers from the campus community.
2. Each participant was assigned a domain.
3. Described as an interactive game with buttons.
4. Getting familiar with the game controls.
5. Suggest modified buttons to make winning the game “easier and faster”.
6. Number of buttons was restricted.

Domains

Taxi Domain



Pac-Man



- 3 Buttons for Taxi
- 4 Buttons for PacMan

Human designed “Options”

Taxi Domain	
Button Name	Percentage of participants who gave this button
Go to passenger and pickup	60%
Go to destination and dropoff	60%
Go to passenger	40%
Go to destination	40%
Pickup/Dropoff	40%
Move away from obstacles	20%
Pac-Man	
Go to the closest food	100%
Avoid ghost	100%
Go to the nearest power pellet	100%
Eat the ghost	40%

- Each button name can be thought of as a sequence of temporally extended low-level actions.

Instantiating Human Options

1. Interaction Phase
   – Define each option as a function of features

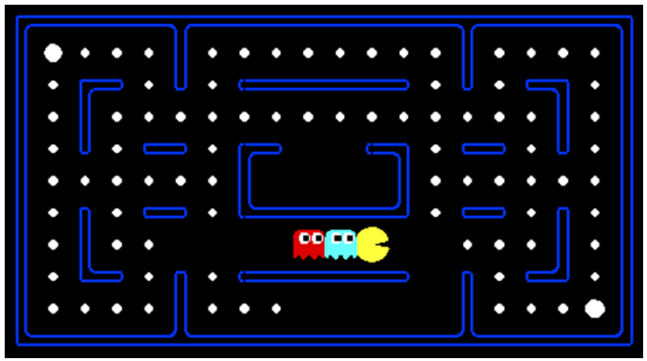
2. Learning Phase
   – Learn the components of each option

Interaction Phase

- Human-Options depend on specific features or attributes of the domain.
- “Go to passenger” depends only on the passenger location.
- We learn the option of the form – Go to passenger(x,y)
- Advantages
  - Allows for learning in a reduced state space
  - Makes the option generalizable

Interaction Phase

- Define the features.
- Construct the world with only those features.



- Generate a distribution of start states and have the human show sample trajectories.
- Trajectories are stored as state, action and reward pairs, from start to end.

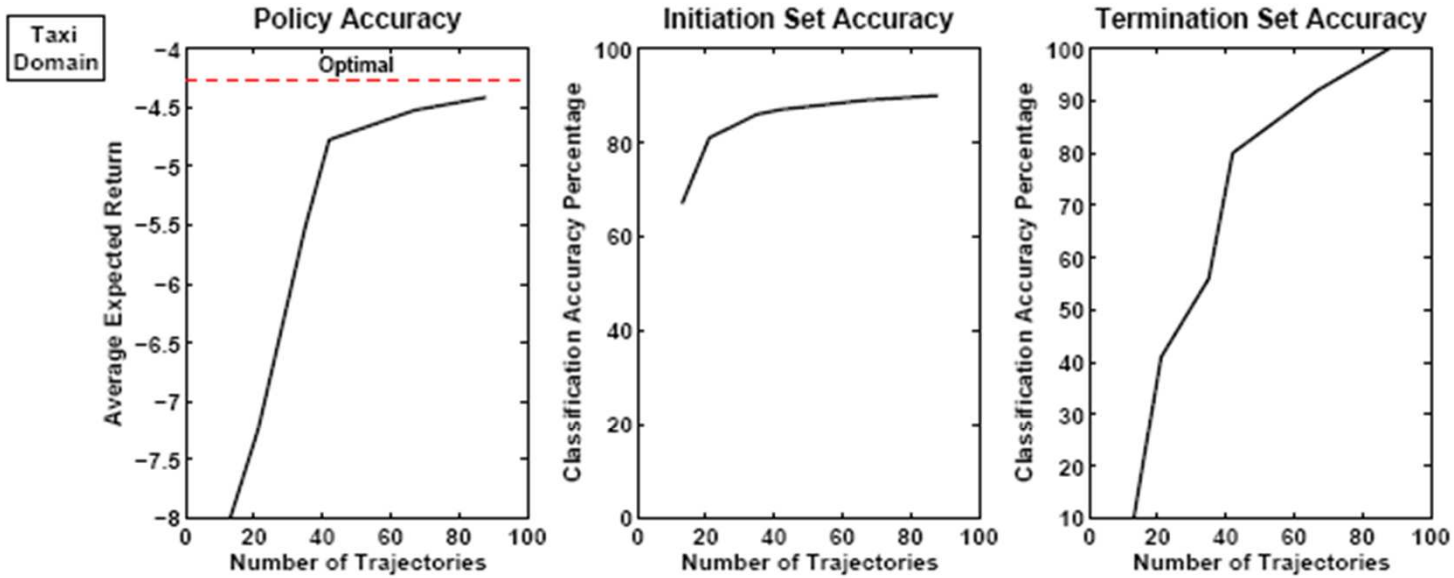
Learning Phase

- Construct 3 decision tree learners, one for each option component.
- The labels are either
  - Action to be performed,
  - Binary label for whether it belongs to I or β
- Each decision tree provides us with a rule-based model of each option component.

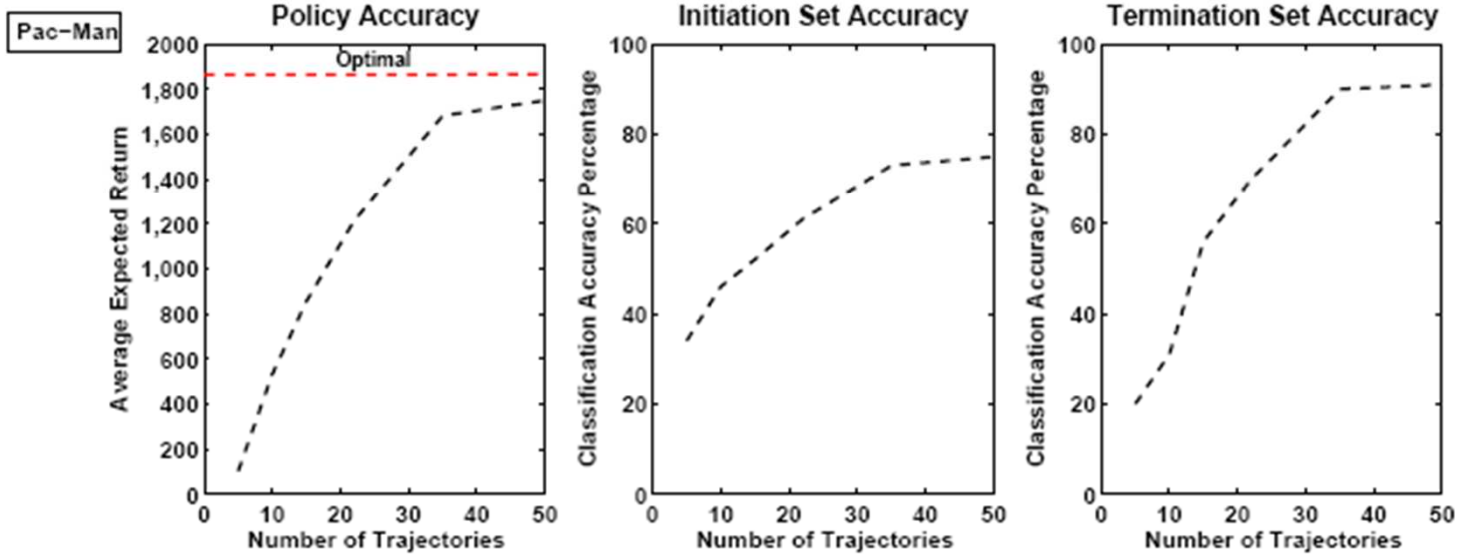
Learning Phase

- Criteria
  - Sample trajectories and match the rewards obtained with the human trajectories.
  - Use the cross validation error to compute the error of the Initiation Set and Termination Set on unseen data.

- Higher errors indicated more human demonstrations were required.



(a) Optimality of "Go to destination and dropoff" human-option in the Taxi Domain



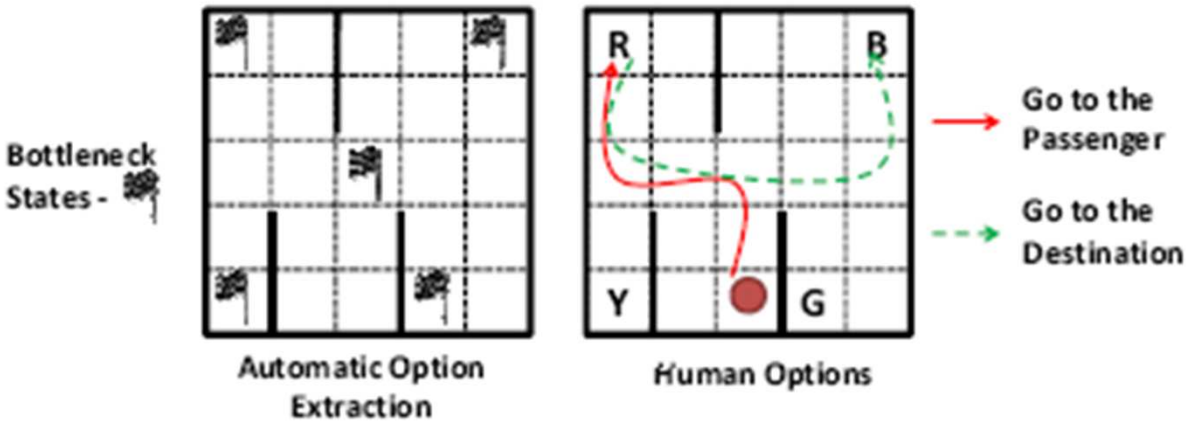
(b) Optimality of "Go to nearest food pellet" human-option in the Pac-Man domain

Experiments

- Automatic decomposition
- Human options vs Automatic options vs Primitive actions
  - Planning Time
  - Learning Rates

Automatic Decomposition

- We make algorithm described by Stolle and Precup in 2002.
- Extract options using a notion of bottleneck states.
- Options obtained,

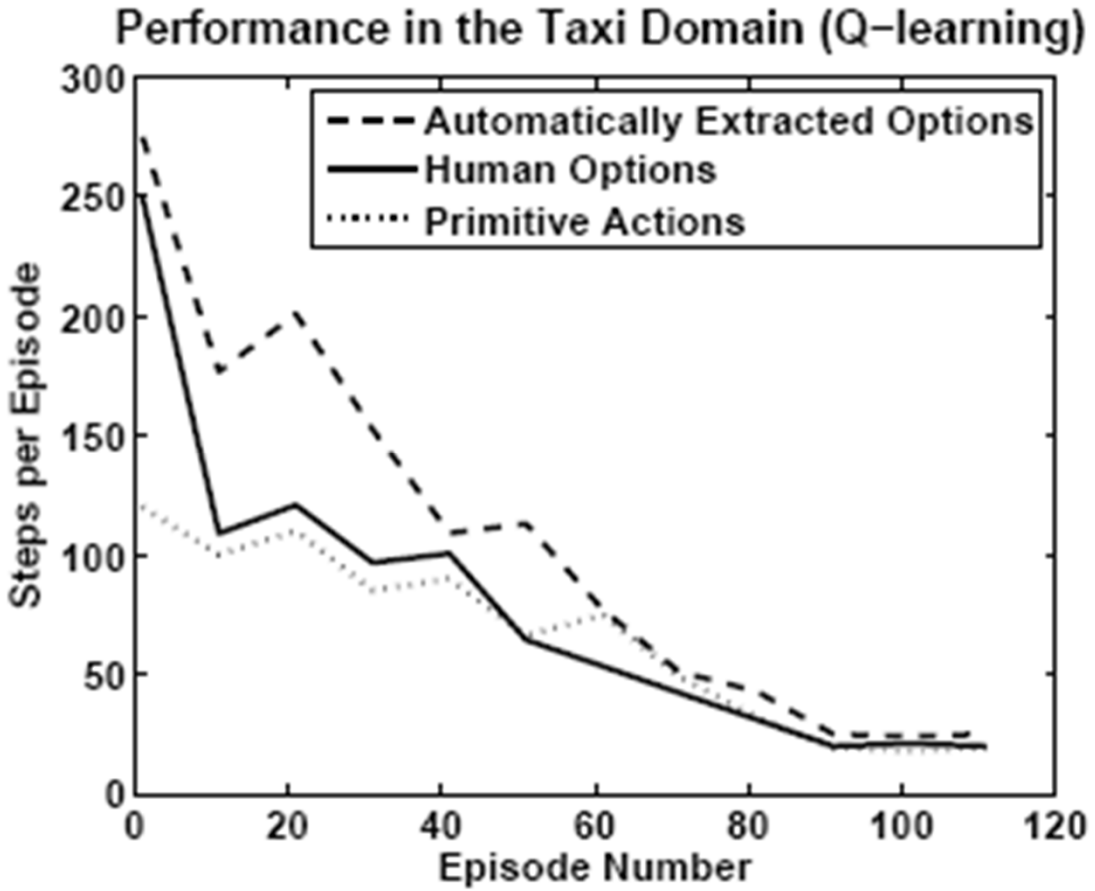


Performance

- The speedup in planning for Human Options is attributed to the reduced number of parametric options.
- The automatic options in PacMan were not complete.

Model	Computation Time (seconds)	Average Reward of Computed Policy
Taxi Domain		
Primitive actions	17.45	-2.294
Human options	10.45	-4.294
Automated options	25.75	-4.311
Pac-Man		
Primitive actions	600	1890
Human options	60.45	1790
Automated options	120.47	1442

Performance



Discussion Points

- Learning options in a reduced state space.
- Fewer parametric options that are easily generalizable.
- The need to refine options.
- “Move away from ghost” – option?
- What about options that do not end.

Conclusions

- Everyday human problem decomposition – consistent with the options framework.
- Option components – learnt using supervised learning.
- Options are parameterized and efficiently generalize across the state space.
- Provide significant speed-up in planning time.

Future Work

- Option Refinement
- Interruption of Options
- Transitions between Options

- Modular Reinforcement Learning

Thank you

Additional Details

Reward Function

- Taxi, 10 for reaching the goal, -1 for each step

- PacMan, 10 points for each food, 50 for eating a pellet, 200 for eating a ghost, 500 for winning and -400 for dying. The reward decreases as time goes on and PacMan is not eating.

Features

- Features for Taxi - position of the passenger, position of the destination, passenger is in the taxi boolean, taxi is at the destination boolean

- Features for PacMan - position of PacMan, position of the ghosts, distance to the ghosts, distance to the nearest food, distance to the nearest food pellet.

Interaction Phase

- The features for each option are extracted by a simple question answer round.

Learning Phase

- Learning Phase - Multiple rounds of K-Fold Cross Validation. These errors are averaged.