Roots and humans teaming up in pursuit-evasion games: modeling and learning issues

Shen Li

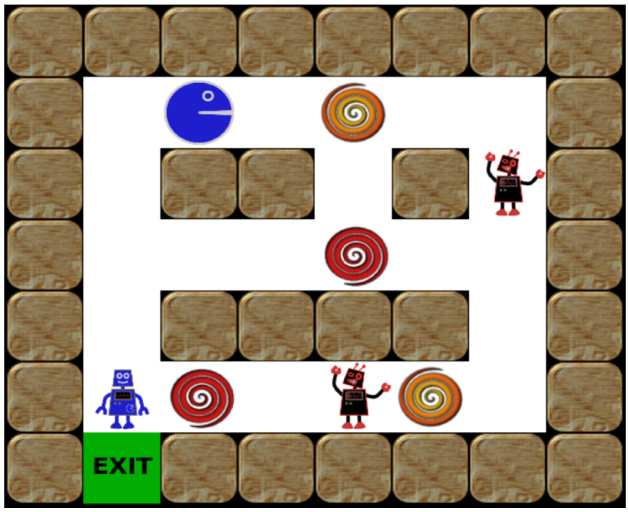
October 17, 2016

1 Introduction

1.1 Pursuit-evasion Game

1.1.1 Game

- World = undirected graph with exits
- Pursuers and evaders take turns to act
  - ♢ Pursuers move at the same time triggered by human
    - ⋆ Human pursuer has an imprecise model of the world
    - ⋆ Robot pursuer adapts it π^* for the non-optimality of its human teammate
  - ♢ Robot evaders move at the same time rationally
- Robot evaders will be removed when they cannot move except stepping onto a pursuer or evader SL:



(a)

Figure 1: The pursuit-evasion game (agents can be teleported through swirls)

so evaders have to be cornered before removed? Yes! It has to be cornered by other pursuers or evaders

1.1.2 Challenges

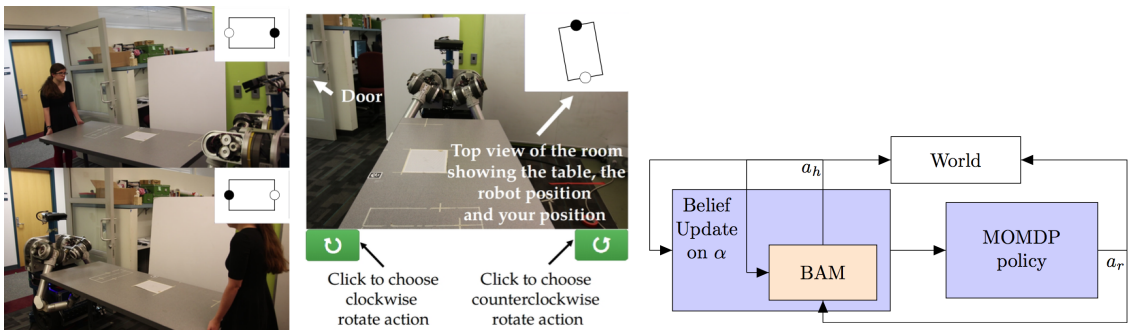
- It is hard for 1 human and 1 robot to predict each other's behavior
- It is harder for multiple humans and multiple robots to predict each other's behavior
- Simplification: more pursuers than evaders and robot pursuers cannot capture all the evaders without human help
- SL: Future work: we can model the fact that the human will also try to interpret and adapt to the robots' behaviors. And also we can model if human trusts his robot teammates or not.

1.1.3 Motivation

- Investigate how a robot can account for human teammates and adapt

1.1.4 Related Work

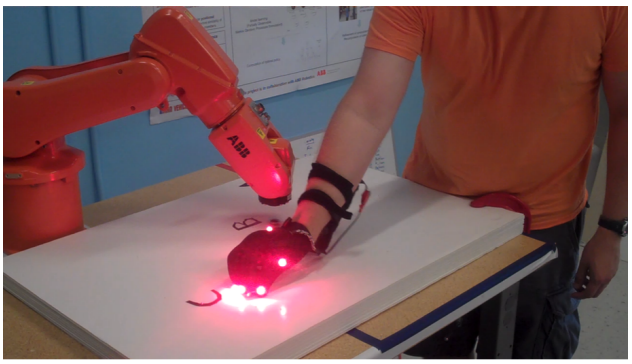
- Bounded memory model for table carrying task in Fig. 2 ([1]).



(a) (b)

Figure 2: In a, the one on the left is the human-robot table carrying task. Rotating the table so that the robot is facing the door (top, Goal 1) is better than the other direction (bottom, Goal 2), since the exit is included in the robot's field of view and the robot can avoid collisions. The one on the right is the UI with instructions. In b, Integration of BAM into MOMDP formulation.

- Cross training Fig. 3 ([2]).



(a)

Figure 3: In a,

- my project on when to intervene

1.1.5 Contribution

- Usually people have to use reinforcement learning to train a robot model used for this particular person, which requires lots of data before start. We want the game to play at start. We need to learn the parameters on-line quickly to get a team policy.
- We have a problem.
  ⇒ What model we use?
  ⇒ How to take into account that human acts optimally based on limit information? Human less capable ⇔ more random in the transition function of human action
  ⇒ What parameters we want to use to shape the model in real time based on the human input?
  ⇒ We can make the robot act optimally, but optimality means making the best decision **based on the current state**. Not the absolute best decision in all time.
  ⇒ This is very expensive computationally so we have to do approximation, which will make robot pursuers suboptimal
  ⇒ How to deal with robot pursuers suboptimality vs human suboptimality?
- A trust model of human
- SL: applications? see the last section
- SL: Future work
  - ♢ Does explanations, demonstrations, or hints help to make collaboration more fluent?
  - ♢ Now we are enabling robots to collaborate with people and adapt to people, but we can also make people adapt to robots. The robots have to assist people when only people, not the robots, can capture the evaders. So the robots will give explanations, demonstrations, or hints to change human behaviors so that people can follow optimal policy computed by the robots. In which way the people can follow the commands from the robots? In natural language or demonstrations?
  - ♢ The reason why the robots cannot just give orders to the people about what to do is that robots and people are better at different things. For example, robots are better at computing but people are better at heuristics. How to make them do what they are good in collaboration?

1.1.6 Assumptions

- Full knowledge: all agents know all agents' positions
- The number of people and robots are fixed
  SL: Future work: we can relax it? Randomly adding new members or removing members
- Centralized control
  SL: Future work: distributed?
- Evaders do not cooperate and acting based on a fixed policy to minimize the pursuers' reward
  SL: Future work: human + robots vs human + robots. We can also collect data about how people play against people without robots.
- SL: How to make sure human can win (winning strategy)?

1.1.7 MDP

- Human $h \in H$, robot pursuer $p \in P$, robot evader $e \in E$
- $s_h \in S_H, s_p \in S_P, s_e \in S_E$
- $\forall x \in H + P + E$, $A(s_x) = N(v(s_x))$ - neighbor of occupied vertices of x (deterministic)

- R_{shared} = shared by $\forall p \in P, h \in H \propto \frac{\text{number of captured evaders}}{\text{number of steps } p,h \text{ have taken}}$

  SL: common reward for pursuers and single MDP for each evader

- $\forall x \in H + P + E,\ T(s_{x,t+1}|s_{x,t}, a_{x,t}) \leftarrow Pr(s_{x,t+1}|s_{x,t}, a_{x,t}) \leftarrow \{s_{x',t} \mid \forall x' \in H + P + E\ s.t.\ x' \neq x\}$

- $\forall e \in E$ (semi-deterministic)

$$A^*(s_e) = \begin{cases} \text{move\_away} & \forall x \in H + P, ||s_e - s_x|| < \epsilon \\ \text{argmin}_{a \in A(s_e)} R_{shared} & \text{otherwise} \end{cases}$$

  SL: 1 - maximize the distance to that particular pursuer? Yes
  SL: 2 - minimize the total reward of all the pursuers? Each evader has its own MDP

- $\forall p \in P$ (deterministic)
  $A^*(s_p) = \text{argmax}_{a \in A(s_p)} R_{shared}$

  SL: each agent acts based on the latest state? Not on other agents' actions? Yes! trying to be optimal based on the current state
  SL: Would the pursuers and the evaders lock each other and no one is gonna win??? No! because the evaders never cooperate. So each of them will have a single MDP to make the optimal decisions. But the pursuers are sharing 1 single big MDP to make the optimal decisions together, which means that they cooperate. Human is also part of the pursuer model.

- Discounted
  SL: why?

- Steps

  1. Robot pursuers compute the evader actions based on optimality (you can pre-compute their optimal actions), or locally optimal, or some simple heuristic (e.g. the evaders will go random or greedy or $\epsilon - greedy$ to maximize the distance to pursuers)
  2. Robot pursuers update the transition function (probability distribution) for the human
  3. After taking into account the randomness introduced by the human (flatten the probability distribution) and those fixed evader actions, we can do value iteration or some approximation to compute the optimal policy in the big MDP shared by all pursuers.
  4. At the same time, each robot evader will compute optimal action based on its single MDP and execute.
  5. We can learn some parameters and how to represent human ability

1.1.8 Human Model

- Human ability value prior ⇐ human input

  ♢ ability to discriminate among choices = difficulty of the decision ⇐ $||a_h||$

  ♢ ability to look ahead = ability to foresee the impacts of decisions ⇐ min(number of turns before a capture)

  ♢ ability to grasp useful knowledge about the current state = $Q_h(s, a)$ ⇐ $s_{x,t} \forall x \in H + P + E$

  ♢ SL: ability to understand the robot teammates' reasonings = look back in the past

  ♢ SL: trust or confidence on robot teammates' ability to win the game

  SL: Do we assume these abilities? Do we have to prove them from some psychological literatures? Or are these 3 abilities parts of our contributions? We have to get them from literatures about how to model human ability in playing these games!

- update $Pr(s_{h,t+1}|s_{h,t}, a_h)$

1. update trust τ
2. $Pr(s_{h,t+1}|s_{h,t}, a_h) \leftarrow$ normalize$(Q(s_{h,t}, a_h))$
3. smooth $Pr(s_{h,t+1}|s_{h,t}, a_h)$ based on $||a_h||$
   The more actions, the harder to make choice, the smoother the probability distribution will be. SL: this is flattening the probability distribution in the transition function to be uniform, which will increase the entropy
4. Distance to go = $\tau_{DTG} = min$(number of turns before a capture) $\leftarrow$ optimal human MDP
5. Look ahead τ_{LA} = number of steps the system thinks the human can predict

SL: ???????

SL: human swarm interaction people?

1.1.9 Applications

- We can evaluate our model by compare our prediction about what human will do and what human really do. If it is getting closer and closer as the human keeps playing the game, we are good.

1.1.10 Applications

- In reality, the game will be amazing to study how human play with robots in real time.
- Specific application
  - ♢ This is a task where 1 person and multiple robots are chasing multiple robots. We can apply this directly to scenarios like 1 policeman and multiple robot cars chasing criminals.
- Looking at it from a bigger picture
  - ♢ Multiple evaders = multiple goals in this task
    - ⋆ It is harder than [1] because we have multiple goals because it SL: involve some task planning to choose goals
    - ⋆ It could be used in the table-clearing task where 1 person and multiple robots are collaborating in removing multiple objects from the table
  - ♢ Evaders are moving = the goals are changing in a dynamic environment
    - ⋆ It is harder than table-clearing task because the objects on the table are not moving.
    - ⋆ It is more like the real situation where you have to SL: adapt your goal to the current situation dynamically every time after you achieve a sub-goal because the situation might have changed after you achieve this sub-goal.
  - ♢ Evaders are escaping from pursuers = there are distractions naturally in the task so the task is even harder and the game is more interesting.
    - ⋆ SL: This makes the collaboration between human and multiple robots necessary to finish the task
  - ♢ Task = multiple human collaborate with multiple robots in finishing multiple goals which are changing all the time.
  - ♢ We can also verify the abilities of different evaders and different robot pursuers.
    - ⋆ All the robots have the same ability, but some are good and some are bad.
    - ⋆ All the robots have different abilities. Some are good at chasing or running (speed × 2), while the others are good at capturing (Pr(successful_capturing) is higher).
    - ⋆ Human and robot are good at different things. How do we leverage that? What human abilities and what robot abilities work together more efficiently?
  - ♢ More questions to ask

- ⋆ Human-robot collaboration strategy - split or not?
  - → Rule: whenever a pursuer touches an evader, the evader will be removed.
  - → Question to ask: do human-robot team focus on 1 target at the same time to corner it make sure it will never run away or split to chase different targets?
- ⋆ Task switching
  - → Rule: The evader will be removed only when all the human and robot pursuers circle the evaders so that the evader has no way to go.
  - → Question to ask: human and robot will first focus on 1 target. When do they decide to switch a target because the first target is escaping efficiently?
- ⋆ Dynamic vs static?
- ⋆ Multiple people/robot pursuers/evaders or not?

References

[1] Stefanos Nikolaidis, Anton Kuznetsov, David Hsu, and Siddhartha Srinivasa. Formalizing human-robot mutual adaptation via a bounded memory based model. In *Human-Robot Interaction*, March 2016.

[2] Stefanos Nikolaidis and Julie Shah. Human-robot cross-training: Computational formulation, modeling and evaluation of a human team training strategy. In *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*, pages 33–40. IEEE Press, 2013.