Summary of ***HogRider*** Agent for the Malmo Collaborative AI Challenge

How to run HogRider agent

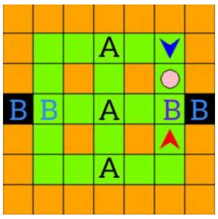
1. Download HogRiderAgent.py and pig_chase_eval_HogRider.py and put them into the default pig_chase folder.
2. Use command “python pig_chase_eval_HogRider.py” to run HogRider V.S. ChallengeAgent. Note that it’s necessary to specify PigChaseSymbolicStateBuilder for HogRider.
3. Results can be found in “pig_chase_HogRider.json”
4. If the program is aborted due to Minecraft error with TypeNone state transferred in. Please retry “python pig_chase_eval_HogRider.py”.

Approach

We view the challenge task as an ad-hoc team collaboration problem, where a set of agents need to collaborate without pre-coordination [1]. In such a scenario, agents must have an estimation of other agents’ behaviors and adapt its own strategies. Our approach mainly consists of two parts: (1) agent type recognition and (2) two-step decision-making based on situation evaluation.

(1) In order to recognize the type of the Challenge agent, our agent maintains a belief of the type of the Challenge agent and updates the belief every time receiving a new observation using Bayes rule. We denote the two types of the Challenge agent as $\theta_1 = random, \theta_2 = focused$. The initial belief is $p(\theta_1) = 0.25, p(\theta_2) = 0.75$ and the belief update rule is $p(\theta_i|a) = \frac{p(a|\theta_i)p(\theta_i)}{p(a)}$, where prior probability $p(a|\theta_i)$ is computed based on the agent’s current direction and action policies (i.e., random or Astar).

(2) Our agent takes an action based on (i) belief about the Challenge agent’s type, (ii) the positions of the two agents and the pig, (ii) steps taken so far. Specifically, at each step, our agent will evaluate the current situation and decide an action to take. We first develop a mechanism to evaluate the current situation based on the pig’s position. We categorize the pig’s position into three kinds: uncatchable (A in the figure), catchable by one agent (B in the figure) and catchable by two agents (all other positions). In particular, we regard the B in the green lattice as the same kind with the B in the black lattice, because we are likely to catch the pig when it moves into the black lattice. Even though the pig does not move in, it does not hurt much for our agent to move towards the pig.



After evaluating the current situation, our agent will first decide whether to collaborate with the Challenge agent or just quit the game. If the expected reward of collaboration is negative, our agent would move to the nearest exits along the shortest path. Otherwise, our agent would move to a destination position along the shortest path, with consideration of the other agent’ s destination.

Result

HogRider is tested against ChallengeAgent for over 10 times and each time 100 episodes. The per-step mean score varies from 1.8 to 2.2 and per-episode score is about 14. HogRider judges the type of the other agent with accuracy higher than 90%.

Reference

[1] Stone, Peter, Gal A. Kaminka, Sarit Kraus, and Jeffrey S. Rosenschein. "Ad Hoc Autonomous Agent Teams: Collaboration without Pre-Coordination." In AAAI. 2010.