2A1-2

A Stepwise Training Method for Complex Task Solving Using Reinforcement Learning

Margaret Kim[*1], Chyon Hae Kim[*1] (Waseda Univ.), Johane Takeuchi[*1], Hiroshi Tsujino[*1]

[*1] Honda Research Institute Japan Co., Ltd.

When teaching a task to a robot in real world situations, the robot's state space can become quite complex, and simply using reinforcement learning in such a space can require an unreasonably long amount of time. Previous reports have demonstrated that using an external trainer speeds up the learning process for a reinforcement learning algorithm, but the trainer needs to be able to get the robot to its goal without allowing it to forget the steps it took to get there. In addition, we do not want to sacrifice the autonomy and adaptability of the robot. This paper suggests a stepwise method that allows a sheepdog robot to learn the complex task of how to put a sheep robot into a cage, while minimizing the involvement of the human and of the external trainer.

1. Introduction

Having a robot learn a task using reinforcement learning (RL) allows the robot to discover how best to use its own capabilities and knowledge to complete the task. With complex tasks, the state space of the environment becomes large enough that it is nearly impossible for the robot to learn the task in a reasonable amount of time. In order to encourage faster learning of the task, we are suggesting a stepwise method of educating the robot. By training the robot using a number of separate rewards, we expect to be able to guide the robot into learning how to complete a complex task.

1.1 The Sheepdog Robot Task

The task we have chosen to teach our robot is that of a sheepdog trying to capture a sheep, similar to the experiment performed previously by Vaughan [Vaughan 98]. In Vaughan's experiment, a robot was programmed with an algorithm to chase a flock of ducklings to a specified area of a circular arena, using an overhead camera to monitor positions. In our variation, a single dog robot uses online RL to chase a single sheep robot into a cage, located in the corner of a 3 meter square field, shown in Figure 1. Our robot only uses a camera mounted on its head, making the task more realistic and thus, more difficult.

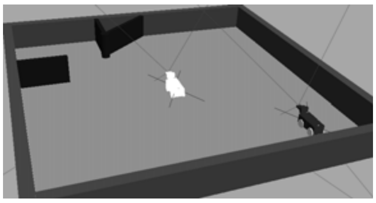

Figure 1. The environment. The environment includes a sheep robot, a dog robot, and a cage encased in walls with a small opening for the sheep to enter.

The dog robot is about 34 [cm] long and 12 [cm] wide, with IR sensors on the corners of its body that it uses for wall avoidance. The head of the robot rotates from side to side, and on the head is a camera that it uses for exploring the environment. The dog robot chooses actions using the method described in section 2.1.

The sheep robot is physically the same robot, but colored white for identification purposes. The sheep robot wanders the environment randomly, except when the dog is nearby or when it encounters a wall. When the dog is nearby, the sheep turns away from the dog and runs in the opposite direction. When the sheep encounters a wall, it turns away from the wall in order to avoid it.

2. Method

2.1 The Stepwise RL Method

The dog robot is programmed with an online SMDP Q-learning algorithm and an ε-greedy action selection strategy whose inputs are the x coordinates of the sheep and cage in the image captured by its camera, and the angle of the robot's head with respect to its body. The SMDP Q-learning algorithm uses Equation 1 to approximate the value function Q^*, where $Q(s,a)$ is the estimated value for a state-action pair, α is a constant step size parameter, r is the reward, and γ^τ is the discount factor, where τ is the time step between actions [Sutton 99].

$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha[r_{t+1} + \gamma^\tau \max_a Q_{t+1}(s,a) - Q_t(s,a)] \qquad \text{(Eq. 1)}$$

Actions were selected from those listed in Table 1, with probability 1-ε that the action with the highest Q-value is chosen.

Table 1. Available Actions for Sheepdog.

#	Action Description
A0	Do nothing
A1	Follow sheep
A2	Turn clockwise around sheep
A3	Turn counterclockwise around sheep
A4	Move away from sheep
A5	Look for sheep
A6	Move away from cage
A7	Look for cage

Contact: Margaret Kim, HRI Japan Co., Ltd., 8-1 Honcho, Wako-shi, Saitama 351-0188, Phone: 048-462-5219 , Fax: 048-462-5221, kimm@jp.honda-ri.com

The robot is trained in a number of subsequent steps, using RL. We begin by offering the robot a reward that can be achieved over a large state space, and change the reward as it learns to achieve each step in the process. We used three steps to teach our robot how to capture the sheep.

(1) RL Steps

- A – Lining up with the Sheep – The sheepdog robot receives a reward when both sheep and cage are in the center of the robot’s view. This causes the robot to line up with the sheep and cage, and also pay attention to the sheep.
- B – Approaching the Sheep while Lined up – While lined up, the sheepdog receives a reward when it moves toward the sheep, thus making the sheep also approach the cage.
- C – Capturing the Sheep – With the final step as the goal, we expect to be able to complete the task.

Each step was trained for two hours, for a total of 6 hours of training time. In addition, a set of trials using only the final step, C, was run in order to compare this method with regular RL.

3. Results

We ran three trials of the simulation using the stepwise method training all three steps, and three trials of the simulation that only used the final reward, offered when the sheep entered the cage.

Table 2. Number of captures in each trial. Taken from the final two hours of the simulation for each trial. In the final two hours, both methods are offering reward C. The ABC method captures an average of 7.8 times per hour, while C captures an average of 2.2 times per hour.

ABC	Trial #	Captures
	1	20
	2	15
	3	2
C	Trial #	Captures
	1	5
	2	2
	3	6

The stepwise method makes 3.6 times more captures than the regular RL method according to the data acquired in these trials.

ABC trial 1, which performed the best, learned the policy shown below in Figure 2.

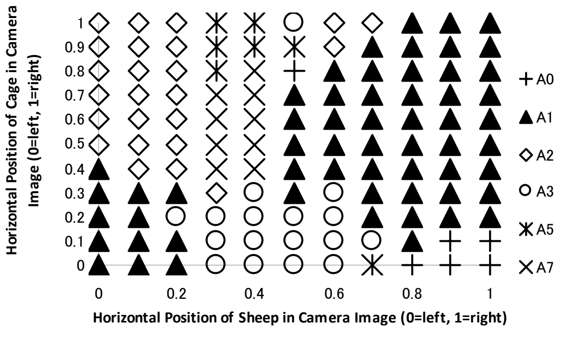

Figure 2. Actions learned in ABC trial 1. The actions listed in the legend on the right are listed in Table 1. The dog would ideally turn counterclockwise around the sheep when the sheep is to the left of the cage (diamonds ◇ in the top left corner of the graph), clockwise around the sheep when the sheep is to the right of the cage (circles ○ in the bottom right of the graph) and approach the sheep when the sheep and cage are lined up (triangles ▲ along the y=x line).

4. Discussion

These results show that it is possible for the stepwise reinforcement learning method to teach a complex task to a robot in a limited amount of time, and it may be more effective than traditional RL. In the same six hour time span, the stepwise method was 3.6 times more effective than the regular RL method at capturing the sheep.

However, this method is of course, not without its challenges.

- This method trains the robot to seek out the sheep agent, line up with it, and approach the sheep in order to put it into the cage. The greatest challenge in stepwise task learning is the transition from one step to another. After the system learns to achieve step A, it must then move on to step B. This is difficult because the system has already learned one policy, and in order to allow it to learn the new policy, we must encourage exploration despite the existence of an already well-learned policy.
- In Table 2, we can see that ABC Trial 3 is of interest, being the major outlier in this experiment. This particular trial had few captures although it used the stepwise method, while the other stepwise trials attained results that were up to 10 times better. This particular dog, unable to distinguish the outside and inside of the cage, spent about 25 minutes of its first hour of learning inside the sheep’s cage, preventing the sheep from entering, and causing the robot to learn the wrong policies. This is an important hidden state that the robot has a great deal of trouble finding.
- The current protocol for training each step of the task involves a fixed amount of time on each step. However, the steps vary in difficulty, which may cause us to overtrain one step and use time that may have been better spent on another step. In addition, the system does on occasion learn an ineffective strategy for achieving a goal. We may want to have the system vary the length of each step depending on the robot’s performance in order to optimize the effect of our training time, and/or re-train a previous step if the current step is showing little or no progress.

5. Future Work

- This stepwise system has the potential to improve performance in circumstances where a similar robot has to learn the same complex task in varying environments. This potential should be investigated.
- Finally, the current experiments have been run entirely in simulation, and should be attempted on real world robots.

References

[Sutton 99] Sutton, R.: Between MPDs and Semi-MPDs: A Framework for Temporal Abstraction in Reinforcement Learning , Artificial Intelligence, Vol. 112, pp. 181-211, (1999).

[Vaughan 98] Vaughan, R.: Robot Sheepdog Project achieves automatic flock control, Proc. of the Fifth Int. Conf. on Simulation of Adaptive Behavior, MIT Press, (1998).