APPLICATION OF REINFORCEMENT LEARNING IN AGENT NAVIGATION

L. Wijerathna[1], A. S. Karunananda[2]

[1,2]Department of Computational Mathematics, University of Moratuwa, Sri Lanka., Email:[1]laksri.w@gmail.com, [2]asokakaru@yahoo.com.

ABSTRACT

Agents interacting with the environment and acting accordingly is one of the challenges in the Agent based technologies. The capability of giving the agent the knowledge about the environment is challenging in the learning process of the agent. Reinforcement learning is a technology in Machine learning which gives the capability of learning from the actions the agent is taking in the environment. In reinforcement learning the learning agent will get the rewards based on the actions taken by the agent and the agent will be trying to maximize the rewards by achieving the maximum rewards. This paper discusses modeling learning agent navigation in a benign environment by using reinforcement learning. The agent will be placed in the benign environment with the zero initial knowledge and then the agent will explore the environment and learn the environment from the rewards getting from the environment. Q Learning algorithm is used to implement the learning process of the agent. The system uses the user defined source and goal as the input and output the optimal navigation path from the given source to the goal by learning the environment itself.

***Key words*:** Reinforcement Learning, Q learning algorithm, learning agent

1. INTRODUCTION

Agent is a unit that acts according to the given set of instructions. The more advanced agent is the one who receives the input from the environment and react accordingly [1]. The agents used in the computer based systems are expected to operate autonomously, perceive the environment, adapt according to situation and sometimes expected to follow the goals. Mostly, the agents are expected to be rational in which they can get the best outcome or the expected outcome [1]. On the other hand, human beings achieve their learning process with less effort due to the cognition which is defined as the mental processing that includes the attention of working memory, comprehending and producing language, reasoning and decision making. Human can gain the intelligence easily due to this cognition. Human gathers knowledge by experience or through communications.

In the process of developing an agent that could navigate from the environment is to be expected to have this level of intelligence of which human has already acquired. The challenge of making afore mentioned achievement is on mapping the intelligence to the computer aided agent.

Machine learning is programming computers to optimize a performance criterion using example data or past experience [2]. In the reinforcement learning, the learner is a decision-making agent that takes actions in an environment and receives reward for its actions in trying to solve a problem. When the agent explores the environment over and over, with respect to the rewards and the penalties that agent getting, the agent will decide the best policy. The best policy is the set of actions which made the agent to get the maximum rewards in the process of operating in the environment [3].

The agent's behavior is challenged by the environment. Such environments can be categorized as fully observable or partially observable Deterministic or stochastic, Sequential or Episodic, Static or Dynamic, Benign or Adversarial [4].

The paper is focuses a benign environment which might be random, stochastic but it has no other intelligent agent in the environment that is actively trying to hurt or fail the learning agent. Benign environment will not contradict with the learning agents own objectives [5].

Many studies have been carried out in the domain of machine learning. One of the successful research is explained by the paper "Learning to win by Reading Manuals in a Monte-Carlo Framework" by S.R.K. Branavan et al explains a machine learning system which reads the manual of the strategic game called Civilization-II and wins the game with high performance [6].The approach model explains about how to get the domain knowledge to effective performance in control systems. In both studies of Roy and Pentland [7] and Yu and Ballard [8] discussed the object names based on images paired with corresponding language. Both studies have the common base of primary operations on parallel corpora of text and grounding contexts. In the domains to which these methods have been applied, the natural language text is tightly and directly linked to the grounding context, allowing the methods to use the parallelism in the data to learn language analysis and gather the required knowledge and act accordingly.

2. METHODOLOGY

Reinforcement learning is a division of Machine Learning that can be taken as the learning process which is led by an agent who is referred to as called as “learning agent” here after, in this context. This learning agent interacts with its environment and observes the results for its interactions. The learning agent also does the decision-making in order to take actions in an environment and receives reward for its actions from its environment in trying to solve a problem without any form of supervision other than its own decision making policy. After a set of trial-and error runs, it should learn the best policy, which is the sequence of actions that maximize the total reward. The learning agent attempts to produce policies that maximize the reward signal. Without prior knowledge with reinforcement learning, a learner must figure out, through multiple attempts which action lead to reward and those that lead to failure. It is about learning what to do - how to map situations to actions, so as to maximize a numerical reward signal.

Similar to most forms of Machine Learning, the learner is not told which actions to take; instead, it must discover which actions yields the most reward. In the most interesting and challenging cases, actions may affect not only the immediate reward but also the next situation and, through that, all subsequent rewards. These two characteristics trial-and-error search and delayed reward are the two most important distinguishing features of reinforcement learning [9].

Reinforcement learning is the main role played by the agents who work as the learner and the environment. Other than that, there are four main sub elements in reinforcement learning system: policy, rewards, values and model of the environment [9]. Policy defines the behavior of the learning agent for a given state. It is a mapping from perceived states from the environment to the actions that are to be taken when in those particular states. The rewarding function defines the goal in the reinforcement learning problem. It maps each perceived state of the environment into a reward depending on the appropriateness of reaching that state. The main objective of the learning agent is to increase the rewards it receives in the process of reaching to the goal. Meanwhile, rewarding function works as a basis for altering the policy. Value function specifies what is good in the long run. The value of a state is the total amount of rewards an agent could expect to accumulate over the future, starting from that state. [9]

The system uses Q Learning algorithm to map the agent navigation with the reinforcement learning. Q-Learning learns the optimal policy even when actions are selected according to a more exploratory or even random policy [9]. The Q-value calculation is shown in Figure 1.

$$\boldsymbol{Q(s,a) \leftarrow Q(s,a) + \alpha[r' + \gamma max_a\, Q(s',a) - Q(s,a)]}$$

Figure 1: Q-value iteration for Q-learning algorithm

In this Q value iteration, the learned action-value function Q, directly approximates, the optimal action-value function, independent of the policy being followed. This simplifies the analysis of the algorithm and enabled early convergence proofs. The policy still has an effect in that it determines which state-action pairs are visited and updated. In this agent based navigation, the agent has to interact with the environment. Figure 2 shows the interaction between the agent and the environment.

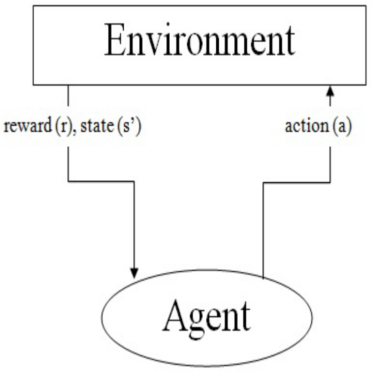

Figure 2: Agents basic interaction with the system

In the agent navigation system, the agent and the environment iteratively collaborate with each other. At the initiating stage, the agent has no knowledge about the environment. Therefore it takes random actions. The action could be moving up, down, right or left. The current state ID is the first knowledge that an agent gathers. Then, the agent takes an action and moves to the next state. Environment rewards the agent and the reward could be either negative or positive based on the state that is visited by the agent.

Figure 3 shows the system architecture which explains the high level architecture of modeling the navigation of agents using reinforcement learning. The reinforcement learning agent contains several components to support the learning and decision making process such as State Recognizer (SR), Goal Detector, Perceptual model Generator (PMG), Policy, Action Selector and Action Optimizer.

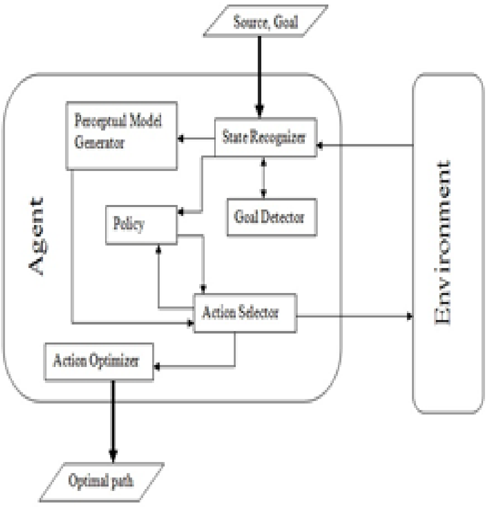

Figure 3: System Architecture

The initial input to the system is the Source and the Goal. State Recognizer initially gets the source and goal details and try to find the goal to start the process with. The state recognizer is the component which is responsible for identifying the Status and the state ID of the each and every goal that the agent is experiencing. The Goal Detector (GD) is responsible of identifying the goal when the agent is navigating through the environment. GD is a special component of the State Recognizer. Perceptual Model Generator (PMG) will get the status of each state and will map the visual field of the agent with respect to the agent's current location. The policy component is responsible for learning the optimal policy to navigate from the given source to the goal. This policy is mapped with the Q-Learning algorithm. Action selector is the component in the learning agent which will be selecting which state to move. The Action selector will get the necessary facts from the PMG and Policy components to make the action decisions. Then the action selector will take an action which will be interacting with the benign environment. The Action optimizer component is responsible of selecting the optimal state sequence to reach to the goal from the given source. The final output of the system is the optimized state sequence to reach to the goal from the destination in the given benign environment.

3. RESULTS

For evaluation, we compare the results generated by the agent to manually constructed sequences of actions on navigating from the given source to the specific goal. An optimal path selected by the agent is correct if it matches with the manually constructed optimal path with the usage of the human knowledge. The agent behavior changes on the number of episodes that the learning agent is trained on. The agent is checked for 25 epochs of finding navigation from a source to goal. And the accuracy is checked by comparing the manually constructed optimal path.

The evaluation process is initiated with training for the 5000 epochs, placed it in the environment to navigate from a given source to the goal. The agent is tested from three separate agent attempts and accuracy is noted down. To calculate the accuracy, the agent is asked to follow the goal from the source given in 25 times. In each time the navigation agent selected is mapped with the human selected optimal path. If the navigation selected by the agent is similar to the path that the human expert selected, then it is a successful attempt. Next, the agent is trained for 5000 more epochs which make altogether 10000 epochs. Again the agent is asked to find the goal from the same source and checked for 25 times. Same as earlier, three separate agent attempts are considered. In the same way, the number of epochs will be increased by 5000 at a time and checked for the accuracy values until agent is trained for 90000 epochs.

4. CONCLUSION

The system is developed to show how the reinforcement learning can be used in the scenario of modeling agent navigation in a benign environment. In particular to implement the reinforcement learning we have used agent who is more often called as a learning agent that uses the temporal difference learning to implement the reinforcement learning. The learning agent uses off-policy TD algorithm called Q-learning. Several trials have been run and to evaluate the accuracy of the learning agent. It is identified that, when the agent learn more from the environment, the accuracy of the decisions that the agent makes is increasing.

One of the major challenges in implementing the learning agent is the selection between the exploration and exploitation. The main objective of the system is to create a learning agent which can learn how to navigate in a benign environment without having any initial knowledge of the environment. The objective is achieved by using the reinforcement learning. We have implemented a class which can mimic the behavior of the agent in the benign environment
As problems encountered in the process are that though the agent can perceive from the environment, the agent which is simulating in the system has neither an actual physical sensors to perceive nor physical actuators to take any actions.

We expect the system to improve by using multi agent reinforcement learning rather than using single learning agent. Using single agent having several limitations as it will limit the scope that particular agent can explore. However, increasing it to the multi agent system that communicates with each other, the system will be able to handle stochastic and adversarial environment. Each agent in the multi agent system can cover a particular part of the environment in such they can communicate with each other and share the knowledge experience from the environment. With this set up the agents will be able to gather more knowledge about the environment within very small time frame which in turn increases the performance of the system.

5. REFERENCES

[1] S. Russell and P. Norvig, *“Artificial intelligence a modern approach”*, Third edition, Pearson Education, Inc , Chapter 1, 2010

[2] E.Alpaydin, *“Lecture slides for Introduction to machine learning”*, The MIT Press 2004, [Online].Available: http://www.cmpe.boun.edu.tr/~ethem/i2ml [Accessed: December 2013]

[3] E.lpaydin, *“Introduction to Machine Learning”*, Second edition, The MIT Press,Cambridge, Massachusetts, Chapter 18., 2010

[4] S. Russell and P. Norvig, *“Artificial intelligence a modern approach”*, Third edition, Pearson Education, Inc , Chapter 27, 2010.

[5] J. Fleuriot, *“Intelligent Agents and their Environments”*, School of informatics, University
Of Edinburgh, [E-book]Available: http://www.inf.ed.ac.uk/teaching/courses/inf2d/timetable/01_Intelligent_Agents.pdf

[6] S. R. K. Branavan, D. Silver and R. Barzilay. *“Learning to win by Reading manuals in a Monte-Carlo Framework”*, Journal of Artificial intelligence Research43 (2012) 661-704, AI Access Foundation, 2012.

[7] D. K. Roy and A. P. Pentland, *“Leaning words from sights and sounds: a computational model”*. Cognitive Science 26, Pages 113-146, 2002

[8] C. Yu and D. H. Ballard, *“On theintegration of grounding language and learning objects”* Department of Computer Science - University of Rochester.

[9] R. S. Sutton and A. G. Barto, *“Reinforcement Learning: An Introduction”*, A Bradford Book, The MIT Press, Cambridge, Massachusetts London, England.