Efficient Learning Scheme: To Enhance Shortest Path Learning Through QRL Search Techniques

Shashikala Mishra[1], Varun Singh[2]

1. M Tech Scholar, 2. Assistant Professor

Rewa Institute of Technology, Rewa M.P.

Abstract - In present Query based RL search techniques are a new and challenging field for agent learning. QRL is became one of the most vital approaches to artificial intelligence. Now QRL is broadly use by diverse research field as quick control, robotics and human computer interaction. It provides us capable solution within mysterious environment, but at the same time it is very much essential to take care of its decision because RL can autonomously learn that is self learner without previous knowledge or training and it takes decision by learning experience through trial-and-error interaction with its environment. In recent time many investigate works was done for RL and researchers has also proposed various algorithm, which tries to solve sequential decision making problems of continuous state and action space. This paper proposed query based RL search techniques and algorithm for making training matrix in the form of look-up-table or state-action pair or Q-table that containing learner (decision making agent) as efficient scheme to enhance the shortest path learning that takes actions over an environment and receive reward for (or penalty). During agent learning, learner requires a lot of training inputs of execution cycle in the form of state action pairs. In order to assess and comparison with other learning scheme, of QRL over number of episodes in the grid world problem in the context of discount rate, learning time, memory usage.

Keywords: Reinforcement learning system, Decision making system, Look up table, Reward, Agent, Discount rate.

I. INTRODUCTION

In Reinforcement learning system is an agent, which can a recognize environment, learn to get the most favorable action to attain the objective. In the machine learning reinforcement learning agent based on trial and error interaction responds in terms of rewards or penalty over given environment.

When the agent makes any action, system learns from environment and given response in the form of signal, which is called reward in terms of positive values[26][28][29]. RL is deals with, in case of multi agent learning and their interaction to make best use of as a mathematical return signal.

Decision making system studies the interactions over an agent and its environment. Look up table represented in the form matrix as an environment and also distinguish and take actions to change[21][22]. Learning, allows the agent to activate in initially unidentified environments and to become more competent than its original awareness alone strength allow towards reinforcement learning. They try to find possible solution in the form of solstices on the basis of scalar evaluation. During number of episodes it's trained and learns to achieve that entire goal [1][2][5][8].

Dynamic programming (DP) and reinforcement system are usually formulated function approximation to train state action values through commutative rewards [17][18][19].

In [33][30][35]describes about state action pairs and the agent move current state a_1 andvia row wise and column using box as sensory input. The agent moves from one box to another by selecting between four moves (Up,Down,Left,Rright) and the agent's gain is increased by the reward indicated in each box. The objective is to discover a strategy that maximizes the cumulative reward. In contrast with the algorithms model-free methods do not require priori known transition and reward models. In short a representation of the MDP [31][32][25]. The lack of a model generates a need to sample the MDP to gather statistical knowledge about this unknown model. There are many learning agent based on query based on reinforcement learning that the environment by doing actions, estimating the same kind of state value and state-action value functions as model-based techniques. There are popular methods to estimate Q-value functions in a model-free fashion are the query based algorithm [34][23][20].

The determination of self learning defined grid world problem for environment declaration, it is crucial in reinforcement system so, a lot of time has to be spent to designing the state breathing space. It will discuss function approximation as a Q-learning method, which overcome this problem. In many learning algorithms call for a huge integer of learning trials, specifically if the grid world space size is high so, the problem has been about how to apply reinforcement to real world tasks like robot learning, grid world.

Now emphasis query based RL search techniques and algorithm required training matrix in the form of look-up-table or state-action pair or Q-table that containing learner (decision making agent) as efficient scheme to enhance closest nearest path performed action over an environment

and receive reward for (or penalty). During agent learning, learner requires a lot of training inputs of execution cycle from look up table. In the description and comparison with other learning scheme, of QRL over many trials over grid world regarding discount rate, learning time, memory usage.

II. SYSTEM MODEL

In An agent interaction described through the above fig.1.0 [1] and their interaction reacts towards agent earning activities on the basis of their performance. The agent's goal is to behave in a way that maximizes potential reward. RL system [27][16]having five tuple :{ S, A, π, RF, VF}, defined as, environment defined on the basis of E, set of actions is described in terms of A, performed action over policy π, reward signal R respectively. Policy play roles towards establish relationship between agent and environment.

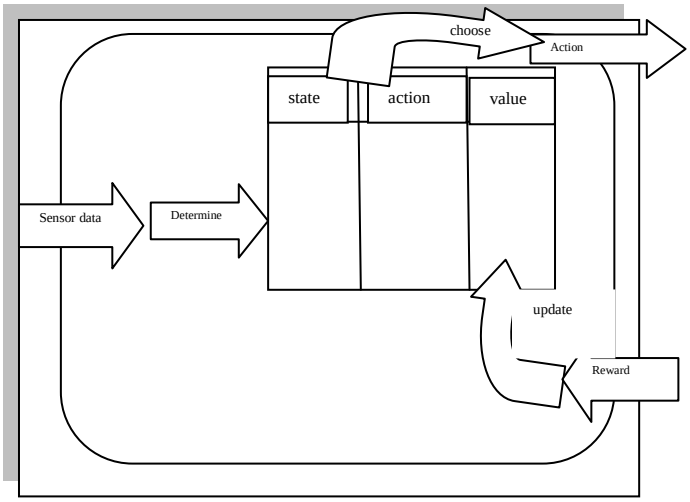
A policy is responsible for a mapping from a state of the environment action by the agent. They also defined and train agents behavior for taking quick action by agents in the form decision making.

A reward function is a mapping from the state or state action pair of the environment over generated values called reward signals. There is an indication of the desirability of the look up table. Important role of learning system is to diagnosis automatic decision making system and their movement. Discount rate defined using mathematical expression $St=s\gamma^t$ their range lies between 0 and 1. Other, it is decomposition of subsequences with contenting goal state from current position resulting to reproduced expected return values [27].policy also given the return value that received from environment.

A greedy method is one that always exploits. Exploration means performed movement under greedy policy. The purpose of investigation is to determine other actions that might be important than the greedy policy. The e-greedy method is one that performs both exploitation and exploration. With probability1-ε, where ε is a small positive number, it takes the greedy action. It creates a stroke randomly. Any learning system basically their movement towards 4 directions as follows: Environment: State defined by the environment in artificial intelligence.

Decision making agent is that perceives and selects movement over environment from system. Reinforcement learning has been important field of the artificial intelligence, human computer interaction and digital aviation worlds [36][24].

Element of reinforcement learning: Self learning agent to train in the form of reinforcement called the learning agent [1]. Agent is assumed to observe an initial state of the entire state. Select its next action by consulting its current policy and collect whatever reward is provided by the environment through trial and error interaction. That includes everything outside the agent. Agent has sensors to decide on its state to perform action over environment modifies its state.



In recent years, among most reactive control methods, RL had broadly applied into robot navigation field in unknown environment. Self learning and on line learning abilities. Query based RL is unsupervised and on line learning method in which agent is given feedback through interacting with the situation [4][7]. Their view is that actions correlated with high reward have higher repeated probability, while those correlated with low reward have lower repeated probability. Therefore learning is also acts as incremental and real time learning method [6]. It deals with how to interact agents to environment and it to maximize under greedy policy [9][11].

III. PREVIOUS WORK

There is reliable work concerning self learning decision making for an agent through tails-and-error interaction. Ethan al. [1] helps us for comprehensive survey based on query based reinforcement learning in the form of agent learning reward and penalty without former in sequence about system. Most of the offered research work in reinforcement learning focus on humanizing the reward value and episodes in query based learning [3][12]. But in our knowledge agent should be fast searching in less amount time with respect to civilizing discount rate and number of episodes. Existing work on [6][10][13] agent self learning is based on three main approaches: agent on training, agent on work and neural network classifier used for removing unnecessary information in look-up-tale. This helps agent for exact learning. The difference between these previous works is not equal in the context of solutions tool like ANN [14][15], but more applicable in application- oriented approach for learning as well as

searching. In these proposals, neural networks are used as decision helper or classifier.

IV. PROPOSED METHODOLOGY

Inquery based RL search techniques and algorithm for making training matrix in the form of look-up-table or state-action pair or Q-table that containing learner (decision making agent) as efficient scheme to enhance the shortest path learning that takes actions over an environment and receive reward for (or penalty).

Q-Learning Algorithm:

The proposed algorithms fulfill the requirement of agent learning mechanism Agent can be trained in the form of sate-action pair for function approximation as Q-function. The training process requires state input and target output as a goal state. State-action value evaluated in terms of reward value using discount rate γ.The following algorithm that is used in learning process in every episode, in n^{th} time-

Step 1: create arbitrarily initial state (s_n)

Step 2: explore available actions (a_n)

Step 3: selects any one action arbitrarily.

Step 4: if checks previously taken same action then repeat from step one

Step 5: now check for target

Step 6: if goal achieved than next episode

Step 7: else, store (s_n , a_n) in look up table

Step 8: update Q_{n-1} (s_n , a_n) according to.

$R_{1=}R\gamma^{(x_1-i)}$

Step 9: now generate next state (s_{n+1}), using state action pair.

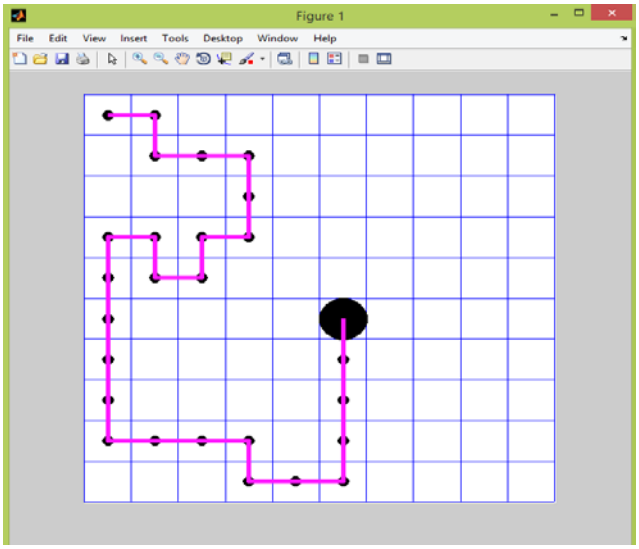
Step 10: if the target achieved

Step 11: else repeat above step-2 until stop criterion is satisfied

V. SIMULATION/EXPERIMENTAL RESULTS

Random Selection via Shortcuts Path

An agent move one step down, up, to the right or to the left whenever if the action is possible. Now look at the result of a random move by agent in a mentioned above. See the Fig.2. Where an agent arrives at the goal cell, it gains the reward 100.The worth of discount rate parameter is set to be 0.9. Comprehensive way to remove loops and find shortcuts from episode for speeding up convergence, While the start cell is (1, 1) and fixed, the goal cell (6, 6) and is determined at random, shown in fig.2. 10x10 grid 26 steps 10x10 grid 11 steps



Figue 2. A snapshot 1 B snapshot finding the goal in 1st

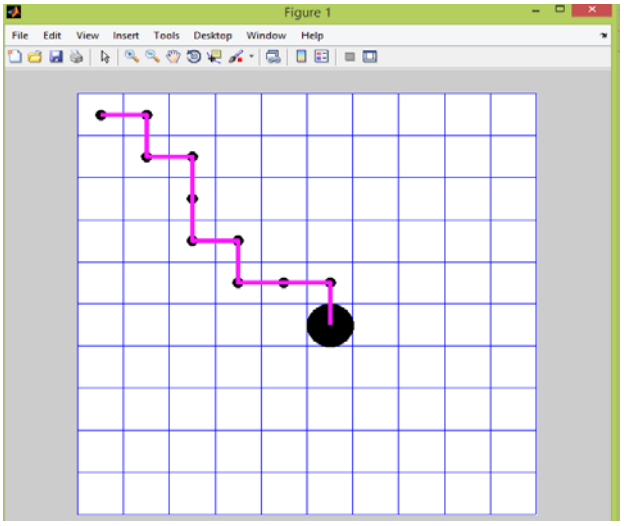
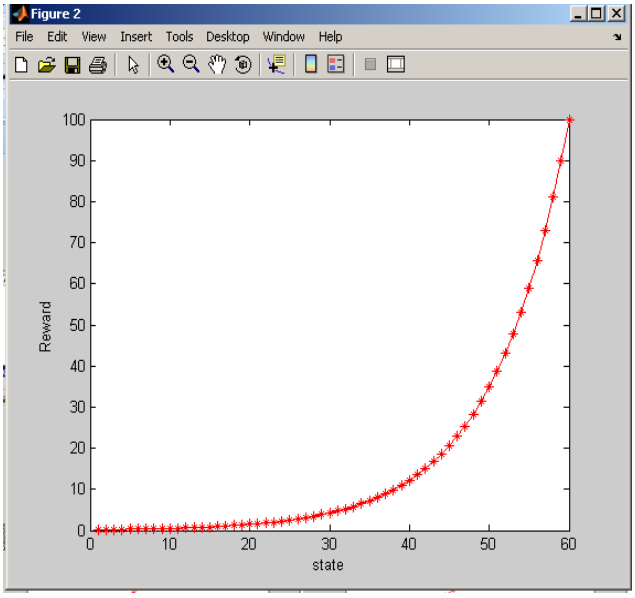


Figure 2. B snapshot finding the goal in 2nd



Figurre 2. C Snapshot graph reward and state

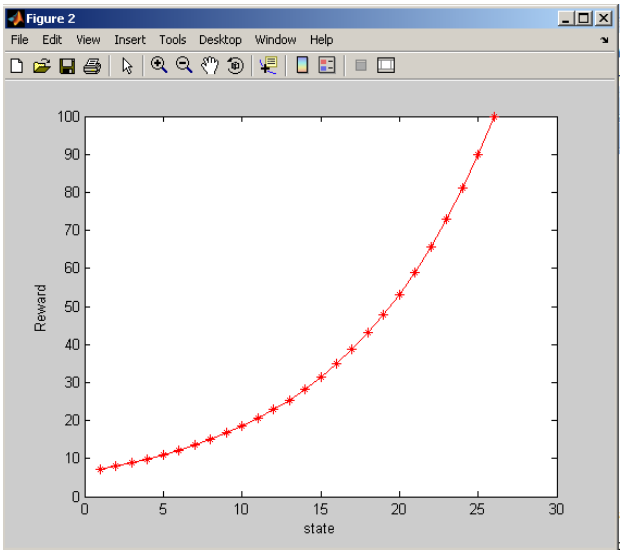


Figure 2. D Snapshot graph reward &state

Figure2 Grid-world of 10x10

Fig.2: In the grid-world of 10x10, starting from (1, 1) an Agent moves aiming the goal at (6, 6) of which the agent had no a-priori information. Left: A path chosen from 50 trials by random move to the goal and A´ is the corresponding performance graph. Right: B route of the shortest path to the goal and B´ is the corresponding performance graph.

Performance Comparison

Performance of the reinforcement learning algorithm using Q-learning measures can be used to assess Learning accuracy.

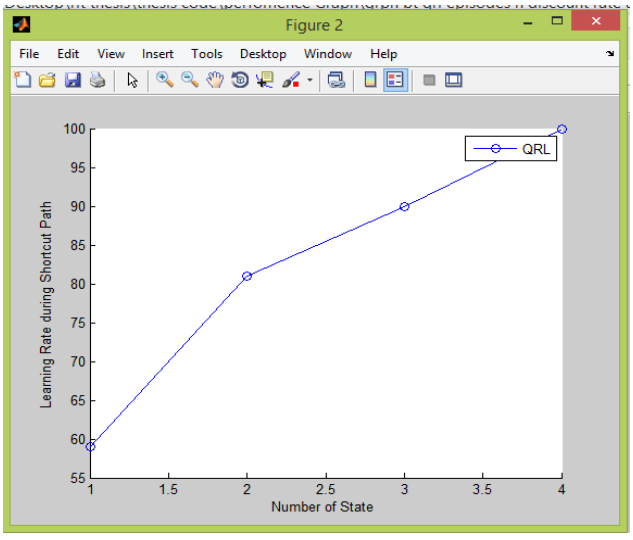


Figure 3. Graph between number of states & learning rate

When agent reaches to the goal, reward value is max 100. Note: that s_{t+1} is automatically determined if action is performed at the state s_t. For example, if the action is up at the state of (2, 8), then the next state will be (1, 8) in figure 2.A.This process of selection and updating of the Q-table is repeated until the goal state is reached. This is called epoch. This epoch is repeated from one agent to the next until the Q-table will not be changed any more until the Q-table becomes stable.

VI. CONCLUSION & FUTURE SCOPES

In this research emphasis on efficient learning scheme to enhance shortest path learning of the reinforcement learning techniques using grid world problem. We showed that knowledge acquired by the agent from environment and corresponding decision path. On the basis of evaluation, we have found that Query based learning is capable of producing trained agent in the form of state action pair as a decision table.

Reinforcement learning techniques and comparison of QRL in the context of learning rate, number of episodes.this section conclusion of the research work should be explained.

REFERENCES

[1] Ethan Alpayadin, “Introduction to machine learning”, MIT press Cambridge, 2005.

[2] Lucian Robot and Bart,” A comprehensive survey of multi agent reinforcement learning”, vol-38, No.2, IEEE 2008, pp. 157-172.

[3] Taraire Taniguchi,”Role differentiation process by division of reward function in multi agent reinforcement learning”, IEEE 2008, pp.387-393.

[4] Hong-Yen Wu, Joe Liu,” A New Navigation Method Based On Reinforcement and Rough Sets”, Proceedings of the Seventh International Conference on Machine Learning and Cybernetics, Kenning, IEEE, 2008 pp. 1093-1098.

[5] Hitoshi Ima and Yaouk Karo, “Swarm Reinforcement Learning Algorithms Based on Sara Method”, IEEE, 2008, pp.2045-2049.

[6] Tomohiro Yamaguchi, Takuma Nishimura, “How to recommend preferable solutions of user in interactive reinforcement learning”, The IEEE, 2008, pp.2050-2055.

[7] Zhao Jin, Wei Yi Liu, Jean, Jin,” Finding Shortcuts from Episode in Multi-agent Reinforcement Learning”, Proceedings of the Eighth International Conference on Machine Learning and Cybernetics, Baoding, IEEE, 2009, pp.2306-2311.

[8] Keita Halmahera, Tadahiro, “Effective integration of imitation learning and reinforcement learning by generating internal reward”, Eighth International Conference on Intelligent Systems Design and Applications, IEEE 2008pp.121-126.

[9] Zhan Shang, Doan Hominy Chen,” Can Reinforcement Learning Always Provide the Best Policy”, IEEE, 2007, pp.224-228.

[10]] Wang Xian, Yang Chinua,” Decoupling control using a PSO-based Reinforcement Learning”, International Conference on Computational Intelligence and Natural Computing IEEE, 2009, pp.170-173.

[11] Renate R., dam Silva, Claudio A," An Enhancement of Relational Reinforcement Learning", IEEE, 2008, pp.2055-2026.

[12] Tom Mitchell, "introduction to machine learning", MIT presses Cambridge, 2005.

[13] Zhao Xiao-hue, Zhao Keck.," Research and Application of Reinforcement Learning Based on Constraint MDP in Coal Mine", IEEE, 2009, pp.687-691.

[14] Shun Dam Wang, Shun Nine Wa1ng," Study on Multi-Agent Simulation System based on Reinforcement Learning Algorithm", IEEE, 2009, pp.523-527.

[15] Petra Kormushev, Kosher Nomo," Time Manipulation Technique for Speeding up Reinforcement Learning in Simulations", Volume 8, No 1, Sofia, 2008, pp.12-24.

[16] Takeshi Shibuya, Shingo Shimada," Experimental Study of the Eligibility Traces In Complex Valued Reinforcement Learning", IEEE, 2007, pp.1630-1635.

[17] Tao Wang, Michael Bowling," Dual Representations for Dynamic Programming and Reinforcement Learning", IEEE, 2007, pp.44-51.

[18] Kao-Shying Hwang," Self Organizing Decision Tree Based on Reinforcement Learning and its Application on State Space Partition", IEEE, 2006, 5088-5093.

[19] Sara Khodayari, M. J. Yazdanpanah," Network Routing Based on Reinforcement Learning in Dynamically Changing Networks", IEEE, 2005, pp.1-5.

[20] Hero, Hoosier," A Motor Learning Neural Model based on Bayesian Network and Reinforcement Learning", IEEE, 2009, pp.1251-1258.

[21] Yuan, Shoran, Rob Powers," Multi-Agent Reinforcement Learning: a critical survey", Computer Science Department Stanford University Stanford, 2003, pp.1-13.

[22] Takashi Kawakami, Masahiro Kinoshita," A Study on Reinforcement Learning Mechanisms with Common Knowledge Field for Heterogeneous Agent Systems", IEEE 1999,pp.469-474.

[23] Leslie, Pack Knelling," Reinforcement Learning: A Survey", Journal of artificial intelligence Research, 1996, pp. 237-277.

[24] Richard S. Sutton, "Reinforcement Learning Architecture for Animates", International Workshop on the simulation of Adaptive Behavior, 1991, pp.1-9.

[25] Habit Karbasian1, Maida N, "Improving Reinforcement Learning Using Temporal Deference Network EUROCON", IEEE, 2009, pp.1716-1722.

[26] Wei Chen, Zhenkun Zhan," Analysis and Design of an Improved R-learning", IEEE 2009, pp.48-52.

[27] Jun Wang Carl Trooper," Optimizing Time Warp Simulation with Reinforcement Learning Techniques", IEEE, 2007, pp.577-584.

[28] Liu Quinn, Cui Zhan Ming," The Research on the Spider of the Domain-Specific Search Engines Based on the Reinforcement Learning", IEEE 2009,pp.588-592.

[29] Zigong Fang and Li Tan," Reinforcement Learning Based Dynamic Network Self-Optimization for Heterogeneous Networks", IEEE, 2009, pp.319-324.

[30] Yang and David Grace," Cognitive Radio with Reinforcement Learning Applied to Heterogeneous Multicast Terrestrial Communication Systems", IEEE, 2009, pp.1-6.

[31] Tom Ere and William D. Smart," What does Shaping Mean for Computational Reinforcement Learning", IEEE, 2008, pp.215-219.

[32] Brian Sallies, Geoffrey E. Hinton," Reinforcement Learning with Factored States and Actions", Journal of Machine Learning Research, 2004, pp.1063–1088.

[33] David E. Moriarty, Alan C. Schultz," Evolutionary Algorithms for Reinforcement Learning", Journal of Artificial Intelligence Research , 1999),pp.241-276.

[34] Marten van Otter," A Survey of Reinforcement Learning in Relational Domains", A Survey of RL in Relational Domains, 2005.

[35] MATLAB 7.0.1 toolbox documentation help, especially network and GUI (Guide).

[36] Vorgelegt von, "Reinforcement Learning for Optimal Control Tasks", University of Technology, Graz, 2005.