# PATH PLANNING OF A ROBOT IN PARTIALLY OBSERVABLE ENVIRONMENT USING Q-LEARNING ALGORITHM

**Hafiz Abdul Muqeet**
Department of Electrical Engineering, University of Engineering and Technology Lahore.
Correspondent email: abdulmuqeet2020@yahoo.com

***ABSTRACT*:** *Autonomous mobile robots have many industrial applications. Mobile robots have to do path planning when performing a task in partially observable environment. It is a challenging problem for the robot to follow the best path for an environment which is unknown to the robot. Our goal is to approach shortest possible path with intelligent planning in partially observable in the presence and absence of obstacles. Q-Learning is special class of machine learning that lies at the intersection of supervised and unsupervised learning. The solution to the Q-learning problem is based on dynamic programming. Its follows an iterative process, which uses online data to find the best possible shortest path, in the presence of obstacles while providing collision avoidance in an partially known environment. The proposed environments have some static obstacles. The purpose of path planning is to determine that how the robot navigates from starting state to goal state without any collision. The discount rate gamma and learning rate alpha of Q-learning algorithm act very important role for this purpose. The results of Q-learning algorithm show that it is an efficient way for optimal path planning of robot.*

**KEYWORDS:** Q-Learning, Partially observable environment, Dynamic Programming, Path Planning

## INTRODUCTION

A robot is such a device that is controlled by a computer program. There are many types of robots on the basis of their functions. Some robots are fully autonomous and some robots are semi-autonomous. The applications of robots are increasing day by day. Some of the common applications are: in industrial applications, office works, in remote areas, working in dangerous areas, military applications, under the water and in space. Navigation of robot is very important. Path planning in partially known area is a very special issue now a day.

### I. Robot Localization

It means that the robot must know its current position with respect to its surroundings. For this purpose, it makes a map of surrounding and determines its position with respect to that point. This can be achieved using the GPS system. In latest technology, GPS system is not useful due to its drawbacks. The main drawback of this system is its accuracy. It gives the information within a few meters of accuracy. For a sensitive and in the nanotechnology industry areas, this system is not acceptable. The robot sensors act a very important role in localization. Sometime these sensors and effectors give inaccurate values. So it is also a challenge of optimality for this issue. There are many reasons of these drawbacks, such as misalignment resolution factor, and variation in contact point. There are two types of probabilistic localization methods. The first one is kalman filter localization and Markov Localization.

The difference between these two methods is their initially known and unknown positions. In kalman filter the robot knows its initial position while in Markov model it takes a start from unknown position. There are many solutions for the navigation of robot. They also have similarities in their algorithm. The main difference is to allocate the environment into small units and cells. It is very interesting to deal with the situation in which the environment of robot partially known or unknown. The robot tackles the situation in such a way that to adapt itself according to environment. There are many real life examples in which the robot has to do the path planning for navigation. The sensors, sense the surrounding environment, but their limitations make them helpless in some situations. The path planning algorithms guide the robot in a specific way and provide the knowledge that how to tackle with different obstacles. In our case only static obstacles are come across to our robot. The purpose of path planning is to reach the final point without collision with any obstacles. This is only possible if the learning ability of robot is high. The proposed algorithm is reinforcement learning. One another task of our algorithm is optimal path. The optimal path can be defined in terms of cost, time and energy. Autonomous mobile robot is a device which can work and navigate without external help. The environment consist of all components such as static obstacles, robot and mobile robots and having the two dimensional area. Now days robots are cheaper, reliable, precise, having good sensors faster and tireless. A robot is used to find the optimal and best paths to reach the goal in its trajectories .The purpose of path planning are to explore the shortest path.

### II. Path planning Process

Path planning is the process of motion from one point to another point. In robot path planning, many issues are includes. Some of the issues are dynamic obstacles avoidance, uncertainties and multiple robots. Generally the basic purpose of path planning is to design the algorithms for optimal path. It also includes the automation of mechanic systems that have many parts such as sensors, actuators and many other capabilities. The purpose of algorithm is to determine how to move starting point to goal point without any collision. Usually robots are used in indoor environment. Various methods have implemented to solve such path planning problems. For example, intelligent vision system that enables the robots to perceive vision by de-centralizing and re-using the learning of each and every robot in the multi robot system. While in improved poly-clonal artificial immune network: memory units are used for preserving antibodies in the specific situations. The other technique is Simultaneous Localization and Mapping (SLAM) for a differential mobile robot along with an optimal control system. A new approach in this field is option-based hierarchical learning in which basic actions are applied in order to accomplish the robot motion planning task. Each behavior is independently learned by the robot in the learning

phase. Afterward, the robot learns to coordinate these basic behaviors to resolve the motion planning task. In our proposed method, if the learning rate and reward discount rate is optimized, the solution of shortest path planning will achieve. The tradeoff between learning rate and discount factor is a big task for proposed environment. If the robot has prior knowledge about the surrounding, then it is called exploitation. On the other hand if the environment completely unknown, robot have to explore the path.

**III. Path planning in partially known environment**

Path planning in unstructured environment is the need of time. The environment where human access not possible. The examples of such areas are: disaster areas, the extreme temperature areas, far away in remote dangerous areas. We can navigate robot in such areas. The robot can sense the situation and send the information to us. The up and down surfaces can stop the robot. But the path planning algorithm controls the robots and assists it to reach the end point without collision. Reinforcement learning is the solution of such problems. This can be used for multi robot cooperation's, and can be used in video games. For unknown environment some roles already define for easiness. The robot can move in four directions only.

**III Reinforcement Learning**

Basically there are two types of algorithms: offline and online algorithm. If agent already knows the environment and its obstacles, it is called offline. If the agent has ability to generate new paths after changing in environment then it is called online path planning algorithm [1]. The gigantic applications of machine learning make it more popular in different fields. In computer sciences a large set of data can be analyze by machine learning. This learning can evaluate the data and predict for future values. The machine learning theory can be categories in three different types, which are supervised, unsupervised and reinforcement learning. We will discuss only reinforcement learning.

Reinforcement learning is a good method to solve the optimization problems. This learning method deals with many practical situations. It can do various operations such as robot navigations, industrial tasks and many more such optimal problems. The basic elements of reinforcement learning are states actions and rewards. Our robot is an agent which has to move in an environment. The robot observes the surrounding area and its current state, where action can change the states [2]. Reinforcement learning is the important type of machine learning. It trains the agent that moves in an environment. The agent can take the decision to select the high value reward. Every reinforcement learning problem has its own situation. The robot (agent) senses the environment and select best reward on the basis of predefined policy. Every algorithm can also be saying reinforcement learning if it solves its problem by itself provided some predefined parameters. The robot prefers to choose such action that already found in previous iterations and have high reward. There are four building blocks of reinforcement learning first one is policy which is a specific value of action for input. The second is reward function, where the third and fourth are value function and model of environment respectively [3]. The most important element of reinforcement learning is policy , that provide the knowledge of future reward [4]. Reinforcement learning can be implemented in markov decision based system. Reinforcement learning also gives the solution for a very large number of states of markov decision process. Reinforcement learning is a simulation based method [5]. The beauty of reinforcement learning is the power of self-learning even in the occurring of minor changes in environment. The exploration and exploitation are control parameters of reinforcement learning. The goal is to have an agent that can take set of action for maximum reward to reach the target [6].

The performance of reinforcement learning is measured on the basis of these two parameters. Exploration usually selects an action, having nonzero probability value. In exploitation agent have to select best action on the basis of existing knowledge by selecting greedy action. The next section explains the Q-learning algorithm, which is the proposed algorithm.

## METHODOLOGY

**Q-Learning**

The best technique in reinforcement learning is Q-Learning. Basically it is online learning. There are three main elements of Q-Learning, which are states, actions, and rewards. The simple equation of Q-Learning expressed by:

$$Q(s, a) = Q(s_t, a_t) + \alpha[R_{t+1} + \gamma * \text{Max}[Q(s_{t+1}, a) - Q(s_t, a)] \quad (1)$$

Q-learning can compare the efficiency of actions even in unstructured environment. Where the rewards show the desirable state of proposed path. On the basis of final reward agent can take future decisions. The goal of Q-learning is to find the optimal rewards to get the optimal value. So, higher values of rewards have selected for path planning. Another method of Q-Learning is relative Q-Learning. This method compares the two immediate rewards. The expression for such algorithm is given below in equation 2.

$$Q(s, a) = Q(s, a) + \alpha\ [\max(r(s, a), r(s', a')) + \gamma \max i' Q(s', a') - Q(s, a)] \quad (2)$$

This algorithm keeps our agent near to goal and maximizes the performance due to its evaluation [7].

Q-Learning makes state-action brace, and update in

Q-table. Mobile robot senses its environment through different sensors [8]. The iterative procedure can be summarized as: $s_t$ is the agent current state, where $S$ is the states and $A$ is the action. The reward denoted by $R$ which is very important parameter of Q-Learning. The strategy of Q-Learning can be define as $\pi: S \quad A$ that is the relation between state and action [9]. By this Q-Learning algorithm we can navigate through unknown environment. Such environment where obstacles may arrive in its path [10]. For navigation purposes digital image processing have also been used now a day. But it is very tedious task [11]. The goal of all this designing and planning is to make the path planning of robot which can avoid the obstacles [12]. The situation for deterministic and non-deterministic is quite different. For converging purposes, both need some modification in basic Q-Learning algorithm [13].

**I Proposed Scenario**

In our case there are six actions and two states. The robot has to traverse these six states and find the optimal path. The environment is partially known. The robot has to traverse through its all paths and finally decide that the path having

maximum reward is the optimal path. For this purpose, we calculate all its values through equation 1, and update in look up table. We have to set the appropriate value of Q-Learning parameters. The discount factor gamma, for our case is taken as 0.5. Whereas learning rate alpha also have value 0.5. The greedy policy restricts its boundary for traversing. The result has shown in next section, where the reward value is different at various point.

## RESULTS

The iterative process of Q-Learning is very useful for finding the best path. The exploration and exploitation tradeoff make it possible to take the decision. The following table 1 has shown the Q-optimal values for different actions. The learning and discount parameters are taken for different results. The second last line shows the maximum value that is our optimal value.

**Table 1: Comparison for Different values of Gamma (Discount factor) and Alpha (Learning rates)**

| Iteration # | Discount Factors (Gamma) | Learning Rate (Alpha) | Greedy Policy (epsilon) | Current state | Next state | Taken action | Next reward | Q-optimal (Value) |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.5 | 0.5 | 0.9 | 1 | 2 | 1 | 0 | 0 |
| 2 | 0.5 | 0.5 | 0.9 | 2 | 1 | -1 | 0 | 0 |
| 3 | 0.5 | 0.5 | 0.8 | 4 | 5 | 1 | 5 | 1.2 |
| 4 | 0.5 | 0.5 | 0.8 | 3 | 2 | -1 | 0 | 2.4 |
| 5 | 0.5 | 0.5 | 0.7 | 3 | 2 | -1 | 0 | 2.5 |
| 6 | 0.7 | 0.5 | 0.7 | 2 | 1 | -1 | 0 | 3.5 |
| 7 | 0.7 | 0.5 | 0.6 | 4 | 5 | 1 | 5 | 3.5 |
| 8 | 0.7 | 0.5 | 0.6 | 2 | 1 | -1 | 0 | 3.5 |
| 9 | 0.7 | 0.5 | 0.5 | 4 | 5 | 1 | 5 | 5 |
| 10 | 0.7 | 0.5 | 0.5 | 3 | 4 | 1 | 0 | 5 |

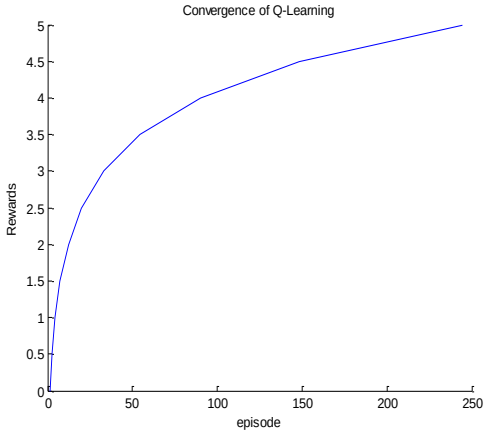


**Fig 1: Convergence of Q-Learning**

The graph shows the relationship between episodes and rewards. The too much low value of episodes shows the performance of proposed algorithm. The reward is in between 0 and 5, which achieve at near 250 iterations.

## DISCUSSION

The learning and discount parameters have great impact on Q-Learning algorithm. The selection of appropriate value of discount factor make sure the future reward. Higher the value of discount factor in starting phase, higher will be the chance of convergence in minimum iterations. The learning rate (alpha) also has importance due to its knowledge of environment. Higher the value of learning rate, shows higher its performance. The rewards and episode relationship shows the overall performance of proposed algorithm. The minimum iteration is the main goal of any path planning algorithm.

## CONCLUSION

The Q-Learning algorithm is very useful technique of path planning, especially in an unknown environment. The learning parameters can be adjusted in such a way to get the optimal value of path searching. Instead of some limitations, it is cost effective as compared to other path planning algorithms. The unknown environment path planning becomes a hot issue, so it is better solution. The performance can be more improve by using sparse matrix method that faster the processing of data.

## REFERENCES

[1] A. Konar, I. G. Chakraborty, S. J. Singh, L. C. Jain, and A. K. Nagar, "A deterministic improved Q-learning for path planning of a mobile robot," *Systems, Man, and Cybernetics: Systems, IEEE Transactions on*, vol. **43**, pp. 1141-1153, 2013.

[2] T. M. Mitchell, "Machine learning. 1997," *Burr Ridge, IL: McGraw Hill*, vol. 45, 1997.

[3] L. G. Hee and M. H. Ang Jr, "An integrated algorithm for autonomous navigation of a mobile robot in an unknown environment," *Journal of Advanced Computational Intelligence Vol*, vol. **12**, 2008.

[4] P. Pandey, D. Pandey, and S. Kumar, "Reinforcement learning by comparing immediate reward," *arXiv preprint arXiv:1009.2566*, 2010.

[5] A. Gosavi, "On step sizes, stochastic shortest paths, and survival probabilities in reinforcement learning," in *Proceedings of the 40th Conference on Winter Simulation*, pp. 525-531 2008.

[6] A. Buitrago-Martínez, R. De La Rosa, and F. Lozano-Martínez, "Hierarchical Reinforcement Learning Approach for Motion Planning in Mobile Robotics," in *Robotics Symposium and Competition (LARS/LARC), 2013 Latin American* , pp. 83-88 2013.

[7] S. Manju and M. Punithavalli, "An analysis of Q-learning algorithms with strategies of reward function," *International Journal on Computer Science and Engineering*, vol. **3**, pp. 814-820, 2011.

[8] Y. Zhang, L. Zhang, and X. Zhang, "Mobile Robot path planning base on the hybrid genetic algorithm in unknown environment," in *Intelligent Systems Design and Applications, 2008. ISDA'08. Eighth International Conference on*, pp. 661-665,2008.

[9] Z. Shi, J. Tu, Q. Zhang, X. Zhang, and J. Wei, "The improved Q-Learning algorithm based on pheromone mechanism for swarm robot system," in *Control Conference (CCC), 2013 32nd Chinese*, pp. 6033-6038,2013.

[10] N. Buniyamin, W. Wan Ngah, N. Sariff, and Z. Mohamad, "A simple local path planning algorithm for autonomous mobile robots," *International journal of systems applications, engineering & development*, vol. 5, pp. 151-159, 2011.

[11] H. N. Joshi and J. Shinde, "An Image Based Path Planning And Motion Planning for Autonomous Robot."2014.

[12] O. Hachour, "Path planning of Autonomous Mobile robot," *International journal of systems applications, engineering & development*, vol. **2**, pp. 178-190, 2008.

[13] Q. Zhang, M. Li, X. Wang, and Y. Zhang, "Reinforcement learning in robot path optimization," *Journal of Software*, vol. **7**, pp. 657-662, 2012.