Autonomous RL: Autonomous Vehicle Obstacle Avoidance in a Dynamic Environment using MLP-SARSA Reinforcement Learning

Conference Paper · May 2019

DOI: 10.1109/ICMSR.2019.8835462

2 authors:

Arvind Srinivasa

20 PUBLICATIONS **189** CITATIONS

SEE PROFILE

J. Senthilnath

Agency for Science, Technology and Research (A*STAR)

167 PUBLICATIONS **3,926** CITATIONS

SEE PROFILE

Autonomous RL: Autonomous Vehicle Obstacle Avoidance in a Dynamic Environment using MLP- SARSA Reinforcement Learning

C.S.Arvind
Computer Science Department.
Dr. Ambedkar Institute of Technology
Bengaluru, India
csarvind2000@gmail.com

J. Senthilnath
Institute for Infocomm Research,
Agency for Science, Technology and Research
(A*STAR), Singapore, 138632
senthil.iiscb@gmail.com

***Abstract*— This paper presents a Multi-Layer Perceptron-State Action Reward State Action (MLP-SARSA) based reinforcement learning methodology for dynamic obstacle detection and avoidance for autonomous vehicle navigation. MLP-SARSA is an on-policy reinforcement learning approach, which gains information and rewards from the environment and helps the autonomous vehicle to avoid dynamic moving obstacles. MLP with SARSA provides a significant advantage over dynamic environment compared to other traditional reinforcement algorithms. In this study, a MLP-SARSA model is trained in a complex urban simulation environment with dynamic obstacles using the pygame library. Experimental results show that the trained MLP-SARSA can navigate the autonomous vehicle in a dynamic environment with more confidences than traditional Q-learning and SARSA reinforcement algorithms.**

Keywords- Autonomous Vehicle; Ultra-sonic Radar; SARSA learning; Q-learning; Multi-level Perceptron

I. INTRODUCTION

Obstacle detection is a basic system from providing drivers and vehicle safety. Autonomous vehicle navigation is a challenging problem with the increase in vehicle density and lack of driving ethics. Safety of passengers, vehicle and surrounding environment are the top priorities for safe navigation. We find different types of the obstacle on road (i) static obstacles like vehicles parked on road and (ii) dynamic obstacles are vehicles travelling on road at different speed and randomly moving obstacle such us animals.

To detect and avoid static and dynamic obstacles are of importance for application such as adaptive cruise control (ACC), autonomous emergency braking (AEB) and forward collision warning (FCW). For safety and convenience, we need information from multiple sensors like vision sensor, ultrasonic radar and lidar. Vision sensors are used to classify and recognize obstacle but cannot accurately specify the distance of the obstacle to take action. To get distance information ultrasonic radar or lidar information need to be fused with a vision sensor to improve the accuracy in detecting the obstacle at different scenarios. For a given instance, the supervised learning approach gives more accurate detection using vision and ultra-sonic radar than traditional approaches [1]. One drawback of this methodology is it needs a lot of ground truth data of different scenarios for training machine learning model [2]. To overcome this limitation supervised learning can be integrated with reinforcement learning technique to reduce the effect in generating the ground truth of obstacles manually.

Reinforcement learning (RL) is a machine learning technique in which agent such as autonomous vehicle will learn its environment based on action, state and reward it obtain from the previous action [3]. There are two modes of learning (i) model-free (ii) model-based. In model-based learning the agent exploits prior learned model to complete the mission whereas in model-free learning dynamic of the environment is learnt using trial-error to update its knowledge. As a result, it does not require space to store all the combination of state and actions to learn about the environment.

Madhu et. al, [4] has developed an autonomous robot which uses q-learning to calculate the shortest path from a present state to goal state in a completely dynamic environment using camera input. Khan et. al, [5] has considered a mobile robot to learn dynamic environment using the neural network with bootstrapping and dropout combined with reinforcement learning with risk-averse collision estimator to avoid collision with an obstacle. Hong et..al, [6] in the study of real hexapod robot system has used ultrasonic sensors data to detect obstacles in an dynamic environment using fuzzy q-learning algorithm & has conducted several groups of experiments to verify the performance of the proposed approach. Farias et. al, [7] has used the neural network model to detect obstacles by fusing proximity sensors data. Chu et. al, [8] has demonstrated a reinforcement learning algorithm can make robot intelligent and navigate autonomously by avoiding collision with static and dynamic obstacles using reinforcement learning. Xia et. al, [9] has also used model-free, off-policy based q-learning for obstacle avoidance for industrial mobile robot navigation. Nihal et. al, [10] has developed that uses state and action set to increase performance in guiding the mobile robot to the desired goal by avoiding obstacles with high success rate using Q-learning and State Action Reward State Action (SARSA). Lucas et. al, [11] has considered all the constraints in detecting obstacles for autonomous driving based on lidar data using Q-learning and SARSA reinforcement learning algorithm under simulation environment.

The present research work of dynamic obstacle is conducted in a controlled environment for a robotic task. The autonomous vehicle has a lot of challenge compared to robotic task (i) precise vehicle control and (ii) traffic rules constraints. To improve continuous action space problem, in this research work, dynamic obstacle detection methodology by combining neural network architecture with a reinforcement learning algorithm for autonomous vehicle navigation. A dynamic urban scenario consists of dynamic obstacles are considered. Ultrasonic radar distance measure is used to determine the obstacles. Traditional SARSA learning with multi-level perceptron neural network can handle continuous action space by efficiently predicting the next action based on present vehicle state, speed and heading angle using distance information. A comparative study is carried out between q-learning, SARSA and SARSA with multi-layer perceptron neural network for dynamic obstacle detection. Firstly, analyze to understand which of the two traditional reinforcement algorithms can understand dynamic scenario faster based on their learning policy. Secondly, compare the reinforcement learning algorithm which is efficient to handle dynamic obstacle scenario.

The paper is organized as follows: Section 2 describes the reinforcement learning algorithms for the detection of dynamic obstacles. In section 3 the details of the experiment and result are presented. Section 4 discusses the conclusion and future work**.**

II. Methodology

In this study, a reinforcement agent is an autonomous vehicle fitted with sensors which need to understand the surrounding area, which is its environment. The agent need to understand the environment using reinforcement learning based on future state (t+1) of the vehicle and reward R(t+1) it obtains from current actions A(t) {Move forward, Stop, Turn Left, Turn Right}. Understanding the dynamic environment faster is key for confident stable navigation in a dynamic environment is explained in detail using reinforcement algorithm such as Q-learning, SARSA learning and combining SARSA learning with Multi-layer perceptron (MLP).

A. Dynamic Obstacle Detection based on Traditional SARSA Learning

Navigation of vehicle autonomously in a dynamic environment is by detecting obstacles and other road anomalies like road border, road divider is based on the distance measure information from ultrasonic sensors. The agent (vehicle) will perform action 'A' like {move forward, stop, turn left, turn right}. The present state of a vehicle is based on 'A' action. SARSA resembles Q-learning. The key difference between SARSA and Q-learning [12] is that SARSA is an on-policy algorithm. On-policy agent learns the value-based on its current action a derived from the current policy instead of the greedy policy. The equation 1 shows based on on-policy Q'(s',a') the future new Q(s,a)' is predicted.

$$\text{New } Q(s,a)' = Q(s,a) + \alpha[R(s,a) + y(Q'(s',a') - Q(s,a)] \qquad (1)$$

where α is the learning rate and 'y' is the discount factor.

SARSA algorithm uses two action selections using the current policy. Current policy will predict future new state (s') and future action (a') is stored in Q-table which help in autonomous vehicle navigation. The positive reward points are awarded for correct action. The negative reward is penalized for the wrong action. The algorithm for dynamic obstacle detection and avoidance using SARSA learning with a learning rate of 0.1 and a discount factor of 0.9 is shown in algorithm 1.

Algorithm1: Autonomous vehicle navigation using SARSA learning

```
Input: Action = A {Move forward, Stop, Turn Left ,Turn Right}, State = X { 1,...,Ns}
Output = Q(X,A) optimal State and Action
Let γЄ [0,1] →Discount factor = 0.9, αЄ [0,1] → learning rate = 0.1 and R→ Reward
Maximum Iteration = 17500
SARSA learning Parameters (X,A,R,T,α,γ)
Initialize S = arbitrary State, A = arbitrary Action, R = arbitrary reward
            Q: S*A→ R
For ii to maxIteration  do
        Start in currentState s Є S
        while s is not terminal do
        Calculate onPolicy ((Π(x) ← Q(x,a))
        currentAction← onPolicy (state)
        if Collision == True
              reward ← R(currentState,currentAction)
              reward = Negative
        else
              reward ← Positive
              update(reward)
         endif
          s' ← T(s,a)  // Receive new state
          Q(s',a) ← Equation 1
          s ← s'
        return
endFor
```

B. Dynamic Obstacle Detection based on MLP-SARSA

Traditional reinforcement learning algorithm struggle in solving complex problems due to (i) sparse nature of q-table handle large data (ii) Maintain two constraint (a) future state (b) future rewards for calculation of optimal future action. To overcome these drawbacks neural networks learning with SARSA algorithm can predict optimal on policy future state and reward values. Optimal Q-value is predicted by a single hidden layer multi-layer perceptron algorithm using gradient descent loss function as,

$$\text{Loss } (Q(S',A') = 1/n \, \Sigma \, |x_i - Q(S,A)| \qquad (2)$$

where x_i is the agent (autonomous vehicle) [present state, present action, reward, new state] and Q(S,A) is the previous iteration Q-value. Based on gradient descent optimization using mean square error loss function, optimal Q-value is predicted for future action. The advantage of Multi-Layer Perceptron (MLP) is it can handle multiple constrains and continuous data. The convergence for optimal Q-value is faster compare to traditional methods. The step taken to detect dynamic moving obstacles and other road anomalies based on ultra-sonic data is explained in algorithm 2 by combining MLP with SARSA algorithm.

Algorithm2: Dynamic Obstacle detection using SARSA with MLP (MLP-SARSA)

```
Input: Action = A {Move forward, Stop, Turn Left, Turn Right}, State = X {1,...,Ns}
Output = Q(X,A) optimal State and Action
Let γЄ [0,1] →Discount factor, α {0,1} → learning rate,
R→ Reward, E → Epochs, iteration = 10^6, minibatch = 64
and Parameters for SARSA (X,A,R,T, α, γ)
Initialize S = Random State, A = Random Action, R = Arbitrarily
Q: S*A→ R
For ii to Epochs do
        For jj to iteration do
                Start in state s Є S
                while s is not terminal do
                Calculate Policy (Π(x) ← Q(x,a))
                        action← Policy (state)
                if CollisionwithObstacle == True
                        reward ← R(state,action)
                        reward = -500
                else
                        reward ← R(state,action)
                        update(reward)
              endif
                        s' ← T(s,a) // Receive new state
             For kk to miniBatch do
                        find (s',a') using MLP()
      Q(s',a) = append (s',a')
    endFor
endFor
```

III. RESULT AND DISCUSSION

A. SIMULATION

Dynamic moving obstacle detection is conducted under a simulator environment, the urban scenario is been simulated using pygame [13], tensorflow [14] and python library is used for multi-layer perceptron neural network model. Figure 1 represents the urban scenario where an autonomous vehicle with ultra-sonic radar will navigate in dynamic moving obstacles with road anomalies like road circle at a road intersection and road borders are represented with black lines. The autonomous vehicle is moving at constant fixed speed at 10kmph. Traditional reinforcement and MLP-SARSA algorithm will learn about the environment from sensor input. The autonomous vehicle is considered with the number of sensors as 7. Positive and negative rewards are set for actions. A positive reward of +5 is set for each step of action without a crash. A negative reward of -500 is set for each crash to determine the obstacle and road borders.

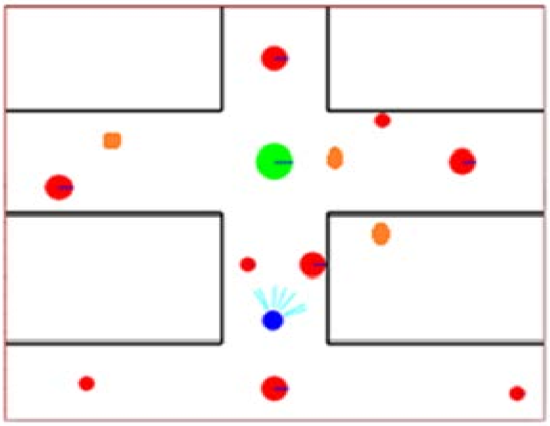

Figure 1. Dynamic Obstacle detection simulation setup of the urban scenario where blue circle represents autonomous vehicle, red circle represents dynamic moving obstacle following road constraints and orange circle represent dynamic random moving obstacles.

TABLE 1. TRAINING Q-LEARNING AND SARSA LEARNING FOR DYNAMIC RANDOM OBSTACLE DETECTION

Algorithm	Iteration	TP	FP	FN	F1
Q-Learning	4000	900	2850	250	0.3673
SARSA		1178	2647	175	0.4550
Q-Learning	8000	4942	3345	287	0.7312
SARSA		5654	3178	268	0.7660
Q-Learning	12000	9861	1982	157	0.9021
SARSA		10185	1687	128	0.9181
Q-Learning	16000	14215	1653	132	0.9409
SARSA		14483	1398	119	0.9502
Q-Learning	17500	16377	998	125	0.9668
SARSA		16450	952	98	0.9690

B. Traditional Reinforcement Learning Dynamic Environment

Q-Learning and SARSA resemble the same but it differs in policy making to predict the future states. There is always a question of how much does the reinforcement need to learn for confident and stable navigation. In our numerical simulation autonomous vehicle with 7 ultra-sonic sensors placed at an angle of -75,-60,-30,0,30,60,75 degrees is made to travel about 17500 km where each frame is considered as 1 km. The simulation is conducted to understand which of the two traditional reinforcement learning algorithm can understand the dynamic scenario faster and gain confidence in collision-free navigation. The result of better understanding is calculated using the F1 score. The correct action is considered as true positive (TP). Collision with the obstacle is considered (FP), colliding with road border is considered as (FN). Table 1 shows the F1 score of different stages of iterations during the learning phase of Q-learning and SARSA learning under dynamic random moving obstacles. From Table 1 and Figure 2, Performance of Q-Learning and SARSA algorithms for dynamic obstacle detection for each iteration intervals wherein figure, the red line (SARSA) indicates faster learning rate compared to the blue line (Q-Learning) at the early stages of learning.

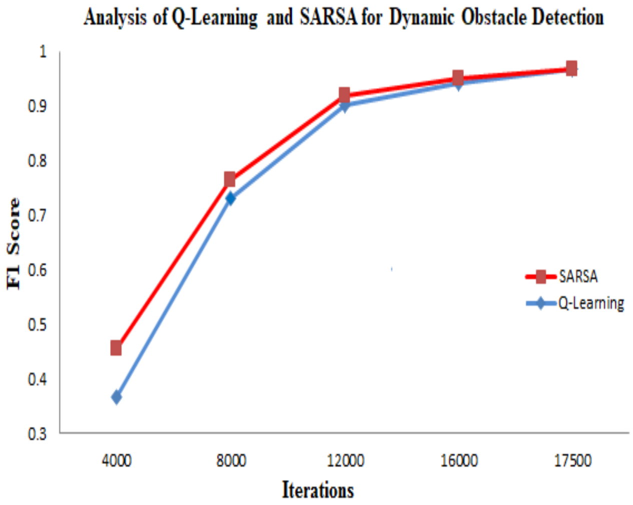

Figure 2. Analysis of Q-Learning and SARSA algorithms for dynamic obstacle detection

C. MLP-SARSA Learning Dynamic Environment

In order to have optimal dynamic obstacle detection, multi-layer perceptron with SARSA learning with the hyper-parameters are set for a single hidden layer neural network. MLP with on-policy SARSA will find optimal future state and action. Table 2 show the different hyper-parameters used in training the MLP-SARSA model. Figure 3(A) shows an instant of training using MLP-SARSA. Figure 3(B) shows the training loss function and figure 3(C) show the learning of dynamic environment.

TABLE 2. TRAINING PARAMETERS FOR SARSA WITH MLP-NN MODEL FOR DYNAMIC OBSTACLE DETECTION.

Parameters	Value
Number of Ultra-Sonic Sensors	7
Ultra-Sonic Sensor Placement Angle	-75^0, -60^0, -30^0 0^o 30^0, 60^0, 75^0
Neural Network Hidden Units	[256]
Number of epochs	1
Number of iteration per epoch	100,000
Batch Size	[64]
Buffer Size for Q-Values	[10000]
Learning Rate α	0.9
Discount Factor γ	0.1

D. Performance Analysis of traditional Q-learning, SARSA and MLP-SARSA

The performance analysis of dynamic obstacle detection using traditional reinforcement learning like q-learning, SARSA and MLP-SARSA is evaluated using ROC curve [15, 16, 17]. Figure 1 as mentioned above shows a simulated testing environment setup for evaluating three algorithms.

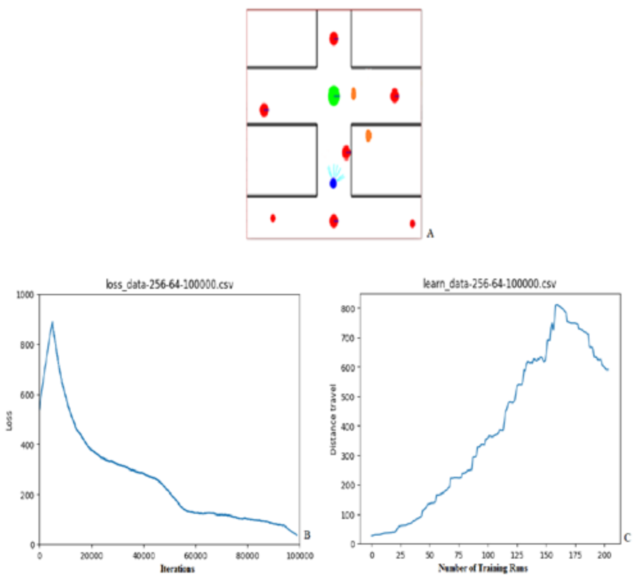

Figure 3(A). Training simulation environment of MLP-SARSA,. 3(B). Training loss of MLP-SARSA with 7 sensor input, 3(C). MLP-SARSA learning of dynamic environment where by 150th run autonomous vehicle is able to run without collision for 800 km.

Traditional SARSA algorithm can learn the dynamic environment faster with less false positive in the early stages of learning compared to Q-learning since it employs on a policy which helps in predicting correct action and reward than greedy policy as the later always look for maximum action and reward values for each iteration as shown in table 1. Even though traditional Q-learning and SARSA can detect and avoid dynamic moving obstacle with 7 sensors with high false positive and false negative. Traditional reinforcement algorithms cannot confidently learn complex scenarios. For confident navigation 7 sensors input, 256 hidden layers MLP with SARSA can understand complex scenarios faster with an accuracy of 0.9878 F1 scores as shown in Table 3.

TABLE 3. DYNAMIC OBSTACLE DETECTION F1 SCORE

Algorithm	Iteration	TP	FP	FN	F1
Q-learning	17500	16377	998	125	0.9668
SARSA	17500	16450	952	98	0.9690
MLP-SARSA	17500	17375	358	68	0.9878

IV. CONCULSION AND FUTURE WORK

In this paper, a reinforcement learning technique, namely, SARSA with MLP for dynamic obstacle detection and avoidance is been developed. The proposed method is proven to be an effective and efficient way of collision detection and avoidance for autonomous vehicle navigation. Simulation experiment results depict SARSA with MLP is able to understand and detect dynamic complex urban scenarios better than conventional Q-learning and SARSA techniques. Although the results are very promising, some challenges are there too early detection of random dynamic obstacles due to random ego-motion, lack of angular range of the ultrasonic sensor. To overcome this drawback, input from multiple sensors like Lidar can be integrated. Further, simulated reinforcement algorithms will be tested on hardware prototype.

REFERENCES

[1] S.D. Pendleton, H. Andersen, X. Du, X. Shen, M. Meghjani, Y.H. Eng, D. Rus, and M.H. Ang, ”Perception, Planning, Control, and Coordination for Autonomous Vehicles,” Machines, 5, 6, 2017.

[2] G. Fernando, M. David, A. De La Escalera, and J.M. Armingol, "Sensor fusion methodology for vehicle detection", IEEE Trans. Intell. Transp. Syst. Magazine, 2017, vol. 9, pp. 123-133,.

[3] R. S. Sutton, and A. G. Barto, “Reinforcement learning: An introduction”, MIT press Cambridge, vol. 1, 1998.

[4] V. M. Babu, U. V. Krishna, and S. K. Shahensha, "An autonomous path finding robot using Q-learning", IEEE International Conference on Intelligent Systems and Control, 2016.

[5] K. Gregory, V. Adam, P. Vitchyr, A. Pieter, and L. Sergey, “Uncertainty-Aware Reinforcement Learning for Collision Avoidance.” CoRR abs/1702.01182 (2010): n. pag.

[6] J. Hong, T. Kaiqiang, and C. Chunlin. “Obstacle avoidance of hexapod robots using fuzzy Q-learning.” (2017) IEEE Symposium Series on Computational Intelligence (SSCI) : 1-6.

[7] G. Farias, E. Fabregas, E. Peralta, H. Vargas, G. Hermosilla, G. Garcia, and S. Dormido, "A Neural Network Approach for Building An Obstacle Detection Model by Fusion of Proximity Sensors Data". Sensors, 18, 683, 2018.

[8] P. Chu H. Vu, D. Yeo, B. Lee., K. Um and K. Cho, "Robot Reinforcement Learning for Automatically Avoiding a Dynamic Obstacle in a Virtual Environment". In: Park J., Chao HC., Arabnia H., Yen N. (eds) Advanced Multimedia and Ubiquitous Engineering. Lecture Notes in Electrical Engineering, 2015, vol 352. Springer, Berlin, Heidelberg

[9] C. Xia., and E. Kamel, "A Reinforcement Learning Method of Obstacle Avoidance for Industrial Mobile Vehicles in Unknown Environments Using Neural Network". In: Qi E., Shen J., Dou R. (eds) Proceedings of the 21st International Conference on Industrial Engineering and Engineering Management 2014. Proceedings of the International Conference on Industrial Engineering and Engineering Management. Atlantis Press, 2015, Paris.

[10] N. Altuntas, E. Imal, N. Emanet, C. Nur and C.N. Ozturk, “Reinforcement learning-based mobile robot navigation”, Turkish J. of Electrical Engineering & Computer Sciences 24 1747-67, 2016.

[11] L. Manuelli, and P.R. Florence, “Reinforcement Learning for Autonomous Driving Obstacle Avoidance using LIDAR”. http://www.peteflorence.com/, 2015

[12] C.S. Arvind, and J. Senthilnath, “Autonomous Vehicle for Obstacle Detection and Avoidance Using Reinforcement Learning”. Advances in Intelligent Systems and Computing, Publisher: Springer, 2018.

[13] https://www.pygame.org/

[14] https://www.tensorflow.org/

[15] J Senthilnath, D. Kumar, JA Benediktsson, and X. Zhang, “A novel hierarchical clustering technique based on splitting and merging”, International Journal of Image and Data Fusion, 7(1), pp.19-41, 2016.

[16] J. Senthilnath, S. Bajpai, S.N. Omkar, P.G. Diwakar, and V. Mani, “An approach to multi-temporal MODIS image analysis using image classification and segmentation,” Advances in Space Research, 50(9), pp.1274-1287, 2012.

[17] A.T. Noman, M.A. Mahmud, and H. Rashid, “Design and implementation of microcontroller based assistive robot for person with blind autism and visual impairment,” 20th International Conference of Computer and Information Technology (ICCIT), 2017.