Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/4/2023321

Obstacle avoidance control algorithm of self-driving electric vehicle based on DRL

Yuhan Wu

Quarry Lane School, Dublin, California, USA 94568

wuyuhan2019@yahoo.com

Abstract. Facing the increasingly severe pressure of environmental pollution, road safety and traffic congestion, intelligent vehicle has become the development focus of modern automobile industry, and it is also the main direction and key competitive field of automobile technology development in the future. This trend puts forward higher requirements for vehicle control system. Under the background of the vigorous development of electric vehicle and automatic driving technology, through the overall planning of obstacle avoidance strategy of automatic driving electric vehicle, studying the correct control method and designing obstacle avoidance system under different working conditions, it has theoretical guiding significance and practical application value for the development of obstacle avoidance control system of automatic driving electric vehicle in the future. This chapter first introduces the basic theoretical knowledge of reinforcement learning, then introduces the DL theory and classical neural network, and finally introduces DQN and DDPG algorithms in DRL, so as to interact RL data. Combining DL with reinforcement learning, the feature representation of complex driving scene is established, which is used as input to train intelligent vehicle obstacle avoidance control strategies that can complete various driving tasks, and transplanted to practical applications, so that they have better performance in automatic driving obstacle avoidance control. To sum up, DRL has great research and application value for obstacle avoidance control of self-driving electric vehicles, which can greatly promote the level of intelligent transportation system.

Keywords: DRL, self driving electric vehicle, obstacle avoidance

1. Introduction

In recent years, with the rapid development of scientific and technological innovation and artificial intelligence, autonomous driving technology has become a new focus in the industry, while electric vehicles have the advantages of simple operation and sustainable development. The autonomous electric vehicle technology has attracted more and more attention, so it is widely used [1]. Obstacle avoidance is a very important part of self-driving electric vehicles in the driving process. Therefore, it is of great significance for the development of autonomous vehicles to study the obstacle avoidance path planning method of autonomous electric vehicles, select an optimal path and improve the adaptability of autonomous driving technology in real scenes [2].Aiming at the practical application problem of obstacle avoidance control strategy of autonomous electric vehicle, researchers integrate deep learning (DL) and reinforcement learning (RL) and propose deep reinforcement learning(DRL).

Among them, the emergence of DRL provides a theoretical basis for the realization of this goal [3]. On the one hand, DRL has strong representation ability for strategy and state, and can be used to simulate complex decision-making process; On the other hand, RL endows agents with the ability of self supervised learning, enabling them to interact with the environment independently and make continuous progress in trial and error [4].

RL is a typical experience-driven, autonomous learning method, which has achieved good results in the fields of robots, unmanned aerial vehicles and vehicles [5]. The driving decision controller of DL selects actions through reinforcement learning, and the added control obstacle function can not only improve vehicle driving safety, but also optimize the strategy exploration process of RL controller and improve exploration efficiency [6].Through continuous interaction with the environment, DRL can learn how to conduct autonomous perception and decision-making control in complex driving scenes, perceive the obtained obstacle information and vehicle location, select the optimal obstacle avoidance method and route, flexibly control the speed and steering of vehicles, and realize smooth and safe driving [7]. In a deeper level, automatic driving technology is bound to effectively reduce the probability of traffic accidents and ensure driving safety. This makes it one of the feasible control schemes to realize obstacle avoidance control of automatic driving electric vehicle.

Through the integration of developing emerging technologies, the autonomous driving industry is expected to enter a golden development period. In order to improve the learning efficiency of autonomous vehicle control algorithm based on reinforcement learning, this paper combines deep neural network and Deep-QNetworks (DQN) algorithm of reinforcement learning, which combines convolution neural network with Q learning algorithm, and adopts DDPG-based RL framework to study DRL. Make use of the advantages of DL and RL to deal with the high-dimensional state space and discrete and continuous action space in decision-making and control problems, and then realize automatic driving obstacle avoidance control.

2. Deep learning theory

2.1. Function value method, strategy-based algorithm, actor-critic algorithm

Value function method is a RL method based on value function. It is a typical RL method. In a special state, each behavior will be given an evaluation value. When a behavior is executed, the higher the expected value, the greater the value of the behavior. Value method is widely used in reinforcement learning. Traditional numerical function methods include Q-learning, DQN, different depth reinforcement models based on DQN, and so on. In addition, many RL algorithms based on actor-critic structure adopt value function.

Corresponding to the value function method is the strategy based method, which is the so-called RL of strategy search. Compared with the value function method, the strategy based method does not evaluate the advantages and disadvantages of the strategy, but directly optimizes the strategy through sampling, so as to achieve the expected cumulative income.

Both methods have their own advantages. The strategy-based method adopts the direct parameterization method, which is simple and intuitive. Compared with the value function method, it has better convergence performance and can better solve large state space problems. However, when the action space is continuous or the dimension of the action space is too high, the value function method can't be effectively solved. Therefore, there are many RL methods that combine strategy search with value function, such as actor-critic algorithm, DDPG depth deterministic strategy gradient algorithm and so on. The strategy gradient integrates the solution idea of the value function approximation method, including actors and critics. The structure is shown in Figure 1.

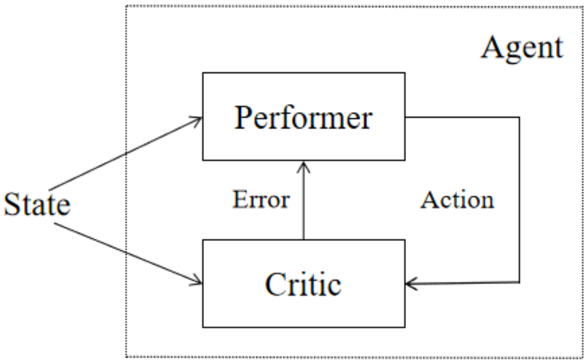

Figure 1. Schematic diagram of actor critic algorithm.

2.2. DRL

At present, the DRL algorithm has roughly two kinds of solutions: value function approximation and strategy gradient method. The DRL based on value function approaches the value function by using deep neural network; Using the deep neural network to approach the strategy and then obtaining the optimal strategy through the strategy gradient method is called DRL based on the strategy gradient.

RL is a typical experience driven and autonomous learning method, which has achieved good results in various fields, such as vehicle and vehicle [8]. RL completes the action strategy through an agent. The goal of the agent is to maximize the cumulative rewards received during its life cycle. As shown in Figure 2, agents can gradually increase their long-term rewards by using knowledge of the expected utility of different state action pairs (i.e. the discount of expected future rewards).

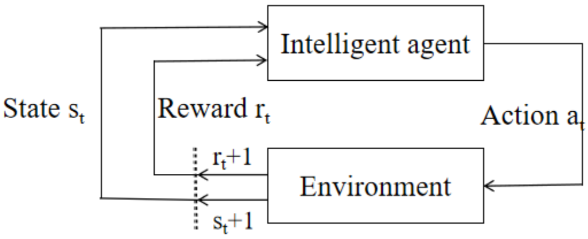

Figure 2. RL framework.

The learning agent interacts with the environment with time. In each time T, the agent obtains the current state st from the state space S, then selects the agent's action at from the action space A according to the current strategy $\pi(a_t|s_t)$, and obtains the scalar reward rt based on the reward function R(s,a), and updates the next one from the state space S according to the state transition probability function $(s_{t+1}|s_t,a)$ of the environment dynamic model. By defining the cumulative reward discount coefficient, the total return that an agent can obtain in a stage is defined as:

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_t + k \tag{1}$$

Markov decision process (MDP) is the most popular solution when formalizing sequential decision problems involving a single RL agent. A Markov decision process is composed of four tuples: state space s, action space a, state transition probability P and reward function R. Namely：

$$MDP = (S, A, P_{\mathrm{sa}}, R) \tag{2}$$

The goal is to find the optimal strategy, so as to produce the highest total expected value of discount reward. The non-aftereffect of Markov simplifies the solution process of RL problem.

In multi-agent path planning, obstacle is a problem that almost all path planning methods have to solve, and it is ubiquitous in all kinds of problems. It and other agents are obstacles that their own agents need to avoid. Obstacle has the characteristics of mechanical trajectory, independence and non-interactivity, which makes it a special existence to be considered in obstacle avoidance. Figure 3 shows an agent avoiding a fixed obstacle. Agent path planning, that is, the agent needs to avoid other agents and obstacles in the environment, so as to plan a whole track. Obstacles always move along their own trajectory and will not take the initiative to avoid objects in any environment, then obstacles can be regarded as an agent with priority $\rightarrow\infty$ and infinite priority, so that they only need to consider themselves without taking care of any situation around them.

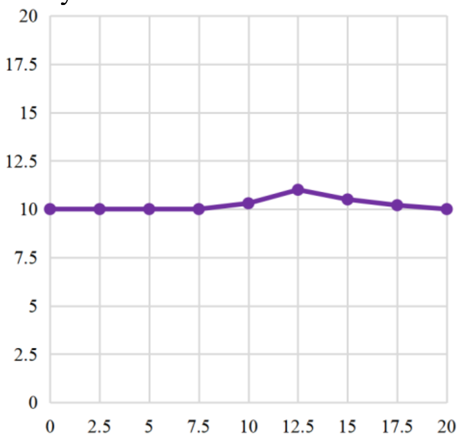

Figure 3. Schematic diagram of obstacle avoidance of agent.

RL is a machine learning algorithm based on Markov Decision Process (MDP). Its core idea is to imitate the trial-and-error and exploration mechanism in bionics to gain advantages and avoid disadvantages. According to the interaction between agents and the environment, it uses constant actions to conduct independent training and learning, and through the reward after interaction, it optimizes the strategy and learns the best behavior [9]. The RL framework is shown in Figure 4. The appearance of intensive learning is another great progress in the history of intensive learning, which broadens the application field of intensive learning again.

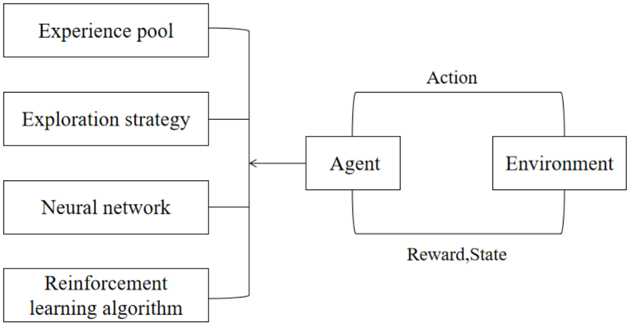

Figure 4. DRL framework.

3. Algorithm based on DRL

3.1. RL Algorithm

DQN is a DRL algorithm improved on the basis of classical RL Q-learning. Q-learning is a RL method based on value iteration ε − Green explores the strategy, replacing the state value function V with the state action value function Q, and the strategy update is also to select the action that can maximize the state Q value of the agent, so it is called Q learning. The Q-learning algorithm directly selects the maximum value when updating the action value function, which is independent of the strategy used by the current selected action. The update formula is as follows:

$$q(s_t,\ a_t) \leftarrow q(s_t,\ a_t) + \alpha\left[\underbrace{\underbrace{r_{t+1} + \lambda \underset{a}{max} q(s_{t+1}, a_t)}_{TDT\,arg\,et} - q(s_t, a_t)}_{TDError}\right] \tag{3}$$

In the deep Q network, the method of Q-learning using Q table to record the state-action value is changed into a method of deep neural network to fit the state-action value function, that is, the Q function. Thus, the discrete Q value is fitted into a continuous function, making it suitable for the infinite state space.

In DRL, the direct use of neural network instead of Q-value matrix will cause the oscillation and divergence of learning strategies. Dqn algorithm introduces empirical playback mechanism and fixed target value network to solve the above problems.

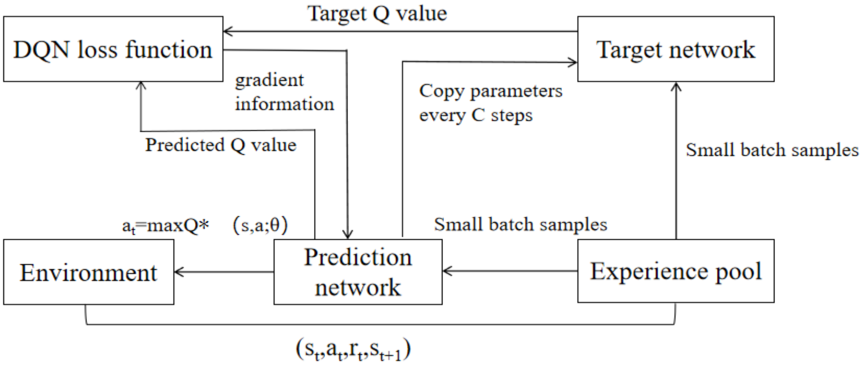

Figure 5. DQN algorithm flow chart.

The learning process of the whole dqn algorithm is shown in Figure 5. The samples obtained by the agent from each interaction with the environment are stored in an experience pool.

3.2. DPG deterministic strategy gradient algorithm

The input of DPG algorithm model is the state space, and the output changes from the probability of each action to the specific action, which is obtained by the deterministic strategy function μ_θ(s). The formula of DPG is as follows:

$$\nabla_\theta J(\theta) = E_{\mu\theta}\left[\nabla_\theta \mu_\theta(s) \nabla\theta q^{\mu\theta}(s,a) | a = \mu\theta(s)\right] \tag{4}$$

In DPG, the policy gradient is a deterministic policy function μ_θ (s). And action value function Q μθ (s, a) find the derivative product of its parameters, and then find the mean value. Compared with the value function approximation method, the strategic gradient method can effectively complete the continuous operation space task, but because of its infinite movement space, it often takes the local optimum as the goal, rather than the global optimum as the goal, and the gradient method's parameters are adjusted very little every time. Its advantage lies in the small fluctuation of the learning process, but the cost is too high.

3.3. DDPG depth deterministic strategy gradient algorithm

Depth deterministic strategy gradient (DDPG) algorithm combines the characteristics of value function approximation method and strategy gradient method. It is a model free algorithm based on actor critic algorithm and DPG algorithm. Compared with DPG algorithm, depth deterministic strategy gradient (DDPG) algorithm combines the advantages of DQN algorithm, effectively removes the correlation and dependence between samples, and speeds up the convergence speed of the algorithm.

The final output can be understood that even under the same strategy π_θ(s, a), the action at output at the same state st is not the same. Compared with stochastic strategy, the solution formula of deterministic strategy gradient is as follows:

$$\nabla_\theta J(\mu_\theta) = E_s{\sim}\rho^\mu\left[\nabla_\theta \mu_\theta(s) \nabla_a Q^\mu(s,a) \middle| a = \mu_\theta(s|\theta^\mu)\right] \tag{5}$$

According to the different convergence requirements between RL and DDPG algorithm, after following the convergence evaluation standard of the return function of inverse reinforcement learning, the convergent return function is extracted and applied to the training process of DDPG algorithm, and the cumulative return curve is obtained, as shown in Figure 6.

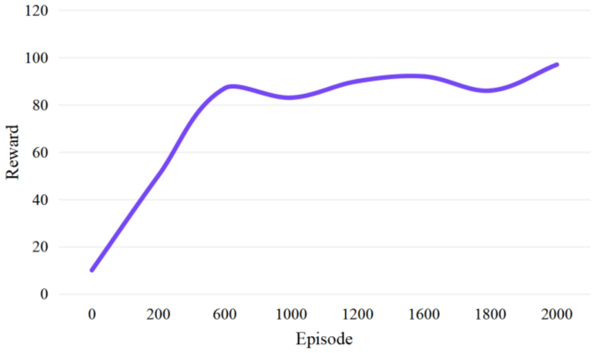

Figure 6. Cumulative return value of reinforcement learning process.

3.4. Obstacle avoidance control of self-driving electric vehicle based on DRL.

The goal of using the DRL algorithm to solve the obstacle avoidance problem is to obtain the joint state s_t^{jn} of autonomous electric vehicles and control the optimal strategy *π* * of output at. In the obstacle avoidance algorithm based on DRL, the agent gets the state s_t^{jn} of the self-driving electric vehicle and all the surrounding cars and pedestrians at the current moment through interaction with the environment, then discretizes the control signals of the self-driving electric vehicle into a certain size of action space, and predicts the state of each control signal in the action space after execution by a one-step prediction model to obtain the predicted state $s^{jn}_{t+\Delta t}$, Then, the predicted state $s^{jn}_{t+\Delta t}$ is input into the value function network, and the value of this state is obtained by combining the reward function. The action in the action space corresponding to the maximum value, that is, the optimal action, is selected as the final output action at of the self-driving electric vehicle. The formula is as follows.

$$a_t = argmax\, R\left(s_t^{jn}, a\right) + \gamma^{\nabla t\cdot v} pref V^*\left(S_{t+\nabla t}^{jn}\right) \tag{6}$$

During the interaction between the agent and the environment, the value function network V * is continuously optimized until it converges. In the value function network, pedestrian interaction information is introduced and the reward function of RL is modified. Based on DRL, an obstacle avoidance algorithm that meets the interaction requirements of autonomous electric vehicles is obtained.

4. Conclusions

DL is a breakthrough in the fields of computer vision and natural language processing through data-driven methods. In DRL, agents obtain data through interaction with the environment. In the process of interaction, agents continue to learn and make progress. DRL has gradually become a hot research topic for researchers in the field of artificial intelligence. The existence of DL and RL makes people have a stronger desire for the application of self-driving electric vehicles and even other artificial intelligence fields in the future. In the field of autonomous driving, DRL has become a hot spot. Firstly, aiming at the defects of shallow valued function network in complex vehicle and pedestrian environment, this method adopts DRL model and uses angle grid to interact. Secondly, the reward function method is used to introduce the comfort requirement of the vehicle into the obstacle avoidance control, so that its obstacle avoidance strategy can better meet the interactive needs of automatic driving, thus realizing the obstacle avoidance control of the vehicle.

References

[1] Wang Bingchen, Si huaiwei, Tan Guozhen Research on automatic driving vehicle control

algorithm based on deep reinforcement learning Journal of Zhengzhou University: Engineering Edition, vol. 41, no. 4, pp. 6, 2020.

[2] Xia Wei, Li Huiyun A study method of automatic driving strategy based on deep reinforcement learning Integration technology, vol. 6, no. 3, pp. 12, 2017.

[3] Dai Shanshan, Liu Quan Safe automatic driving method based on deep reinforcement learning of action constraints Computer science, vol. 48, no. 9, pp. 9, 2021.

[4] Yang Shun, Jiang Yuande, Wu Jian, et al Deep reinforcement learning method of automatic driving based on multi type sensor data Journal of Jilin University: Engineering Edition, no. 4, pp. 8, 2019.

[5] Jing Jie, Chen Tan, Du Wenli, et al Spatiotemporal feature extraction method in depth learning scene Computer science, vol. 46, no. 11, pp. 4, 2019.

[6] Yang Xiao, Li Xiaoting. Research on Autopilot Technology Based on Deep Reinforcement Learning. Network Security Technology and Application, no. 1, pp. 3, 2021.

[7] Zhang zhen. Design of autopilot control software based on deep reinforcement learning. Automation and Instrumentation, no. 10, pp. 4, 2021.

[8] Liu Xian. Research on Autopilot Based on Deep Reinforcement Learning. Automation Application, no. 5, pp. 3, 2020.

[9] Pan Feng, Bao Hong. Research progress of automatic driving control technology based on reinforcement learning. journal of image and graphics, vol. 26, no. 1, pp. 8, 2021.