# Mobile Robot Path Planning Based on Improved DDPG Reinforcement Learning Algorithm

Yuansheng Dong
*Huazhong University of Science & Technology*
*Luoyu Road 1037, Wuhan, China*
ysdhust @hotmail.com

Xingjie Zou
*Huazhong University of Science & Technology*
*Luoyu Road 1037, Wuhan, China*
zxjhust@hotmail.com

*Abstract*—: **Mobile robotics has a wide range of applications and path planning is key to its realization. Mobile robots need to explore the environment autonomously to find their destinations. The Deep Deterministic Policy Gradient (DDPG) algorithm, a classical algorithm in deep reinforcement learning, has a large advantage in continuous control problems. However, the DDPG algorithm suffers from the problems of low training efficiency and slow convergence caused by the high proportion of illegal policies due to the lack of policy action filtering. In this paper, we propose a mobile robot path planning method based on an improved DDPG reinforcement learning algorithm, which uses a small amount of a priori knowledge to accelerate the training of deep reinforcement learning, reduce the number of trial and error, and adopt an adaptive exploration method based on the ε-greedy algorithm. Dynamically adjust the exploration factor to rationally allocate the probability of exploration and exploitation. The adaptive exploration method can improve the exploration efficiency, reduce the exploration duration and speed up the convergence of the algorithm. Simulation experiments are conducted in a grid environment, and the results show that the proposed algorithm can successfully find the optimal path. Moreover, the comparison experiments between Q-learning, SARSA and the proposed algorithm demonstrate that the proposed algorithm has better path planning performance, spends the least computation time and converges the fastest.**

***Keywords-mobile robot, path planning, improved DDPG algorithm, a priori knowledge, adaptive search***

## I. Introduction

Mobile intelligent robot is a kind of robot system which can sense the environment and its own state through sensors, realize the goal-oriented autonomous movement in the environment with obstacles, so as to complete certain operation functions. In an environment full of obstacles, finding a collision free route from the starting position to the target position for mobile robots is a problem to be solved in path planning. However, in general, mobile robots may have found more than one route. At this time, setting some conditions can help the mobile robot to select the most suitable route, such as not touching obstacles, the smoothest route, the shortest time spent, etc. The most commonly used criterion is to see which route has the shortest length. As an important part of intelligent robot research, path planning aims to select an optimal or suboptimal collision free path from the starting point to the end point in the environment where the robot is located. The result of path planning will directly determine whether the robot can complete the task efficiently and accurately, and the design of algorithm is the core content of robot path planning.

Path planning algorithm can be divided into two kinds: classical algorithm and artificial intelligence algorithm. The early classical path planning algorithms include simulated annealing (SA), artificial potential field method, rapid exploring random trees method (RRT) and probabilistic roadmap method. The artificial potential field method was proposed by Khatib in 1986 to solve the problems related to obstacle avoidance. Because of its simple and beautiful mathematical description, it is widely used in many fields [1].S. Akishita et al. used Laplacian potential energy to deal with the motion planning of robot in dynamic environment[2], and D. Alvare et al. added optimization criteria on the basis of Laplace potential energy to achieve better motion effect[3]. Chen Wenbai et al. Introduced chaos optimization method to improve the traditional artificial potential field method, which solved the problems of local minimum and target unreachable in the artificial potential field method, and also made the planned route more smooth [4].Rantanen improved the probabilistic roadmap method to enhance the sampling ability by detecting which areas in the configuration space are easy to implement, and biased the sampling objects to those difficult areas that may contain narrow channels, so as to alleviate the sampling difficulties of narrow channels [5].M. H. Overmars and others proposed a probabilistic roadmap algorithm. The main idea of this algorithm is to randomly place the set of vertices in the free space of the environment. If the straight-line segment connecting them does not pass through any obstacles, the two vertices can form a path that can bypass the obstacles through the straight-line segment connection [6,7]. Therefore, the path planning problem of probabilistic roadmap can be solved by using basic graph search algorithms, such as depth first search [8], breadth first search [9] or Dijkstra algorithm [10], or heuristic search algorithms, such as a * algorithm [11] or D * algorithm [12].

The path planning algorithm based on artificial intelligence includes genetic algorithm and neural network. Mohamed elhoseny et al. Proposed an improved genetic algorithm based on Bezier curve [13]. The algorithm can search the most suitable points as control points of Bezier curve, and optimize the distance between the starting point and the end point. The test results show that the algorithm reduces the energy consumption of the robot in the harsh environment. Mihai Duguleana et al. used Q-learning algorithm and neural network to solve the path planning problem in uncertain workspace [14]. The experimental results show that the algorithm can

**978-1-7281-6579-0/20/$31.00©2020 IEEE**

successfully navigate, and shorten the calculation time, even in the time limited application is feasible. Xue Yang uses non dominated sorting genetic algorithm to solve the multi-agent path planning problem in static environment [15].

As an artificial intelligence technology, reinforcement learning has been widely concerned by scholars at home and abroad because of its ability to learn strategies through autonomous interaction between system and environment in unknown environment, and has become a research hotspot in the field of robot and control. However, there are still many problems and challenges in the application of reinforcement learning algorithm to mobile robot path planning.

(1) Dimension disaster. Most reinforcement learning algorithms consider discrete state space, which is prone to dimensional disaster for robot path planning.

(2) The efficiency of learning and training is not high. Reinforcement learning requires agents and environment to constantly try and error to obtain a large number of state, action and return data, which are sent to the neural network to solve the parameters iteratively.

(3) It is difficult to design a good reward function for different control tasks.

(4) The existing reinforcement learning algorithms have poor stability, difficult convergence and easy to be affected by hyper parameters.

In order to solve the problems mentioned above, we propose a mobile robot path planning based on the improved DDPG reinforcement learning algorithm. The system uses the improved DDPG reinforcement learning algorithm. The content of this paper is arranged as follows: Section 2 introduces the relevant technical background; section 3 proposes the overall structure of the system and describes the system functions; section 4 proposes the experimental analysis of the system; section 5 describes the work of this paper. Finally, the paper summarizes and prospects the future work.

## II. Relation Works

### A. DDPG Algorithm

The classical DQN algorithm and other algorithms evolved from DQN algorithm based on state action value function have made great progress in the control problem of discrete action. Deepmind proposed DPG (deterministic policy gradient) algorithm in deterministic policy gradient algorithms in January 2014 [16].In September 2015, Deepmind used deep neural network to calculate actor and critical values on the basis of DPG algorithm, and used normalization mechanism in deep learning for reference, and proposed DDPG algorithm. The core of DDPG algorithm is actor critical method [17-18], DQN algorithm and deterministic strategy gradient (DPG). DQN algorithm uses the powerful function fitting ability of deep neural network to map environmental state into action strategy and state action pair to value function, avoiding the problem of huge storage space of Q table. The actor critical method is to establish actor and critical networks. The actor network is used to generate the current strategy, and the critical network is used to evaluate the advantages and disadvantages of the current strategy [19]. In order to improve the stability of training, target actor network and target critical network are introduced.

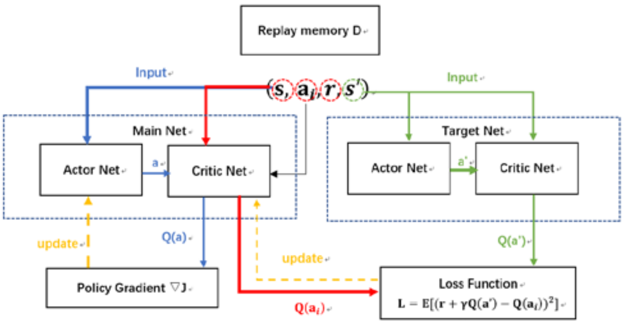

Figure 1 DDPG algorithm

### B. Adaptive exploration method

Greedy algorithm and ϵ - greedy algorithm are the most commonly used strategies of adaptive exploration algorithm.

(1) Greedy algorithm refers to the action that can maximize the value for each state.

The expression is as follows:

$$\pi(s_t) = argmax_a Q\ (\ s_t\ ,\ a\ ) \quad (1)$$

Among them, $\pi(s_t)$ is used to inform the agent of the current status What's the next choice.Value itself is an estimate.If the expected value of a state action pair is greater than its actual value, and the greedy algorithm tends to choose the action with the maximum value, the overestimated state action pair will deviate from the actual value more and more in the iteration process, and its Q value will be larger and larger, and it has a higher probability to become a part of the final strategy. This will undoubtedly affect the algorithm to determine the optimal path, which may take the suboptimal path as the optimal path.

(2) ϵ-greedy algorithm

ϵ-greedy algorithm essentially balances the exploration utilization dilemma in Q-learning algorithm. It is to ensure that the agent can make maximum use of the acquired knowledge under the premise of fully exploring the environment. Exploration refers to the agent groping for the environment without enough prior knowledge. Exploitation means that the agent accumulates certain experience knowledge in the continuous exploration, and uses the experience knowledge to find the way to complete the target task.

## III. Path Planning of Mobile Robot Based on Improved DDPG Reinforcement Learning Algorithm

### A. Experience playback based on weight

The mechanism of experience playback is as follows: it uses a fixed size memory space (also known as replay buffer) to store previous experience, and randomly selects a fixed number of experiences to update the network each time. Obviously, the temporal correlation between the experience of playback is greatly weakened, because the experience playback mechanism mixes the old experience with the new experience to update the network. However, the experience playback mechanism is based on the introduction that all experiences in the experience pool are equally important, so it evenly selects a certain number of samples from the experience pool to train the network. But obviously, this kind of introduction is against the common sense. We propose the value of distinguishing different experiences in the experience pool, that is, the original random experience playback in DDPG is replaced by the experience playback based on weight.

In order to find the optimal action strategy, reinforcement learning needs to set up exploration strategy.In continuous control, one of the main challenges is the exploration of control strategy.Like the original ddpg algorithm, we add the exploration noise to the output strategy of the actor, so the exploration strategy can be written as follows:

$$\mu'(s_t) = \mu\left(s_t|\theta_t^{\mu}\right) + N(\delta,\sigma,u) \qquad (2)$$

Among them, $N(\delta,\sigma,u)$is the Ornstein Uhlenbeck process. Here, Represents the speed at which a variable changes to its average value; Is the average value; It&apos;s the magnitude of the change.

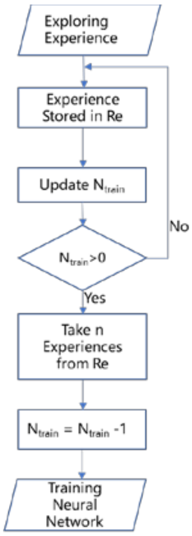

Figure 2 Experience playback based on weight

### B. Adaptive exploration method

Exploration refers to the agent groping for the environment without enough prior knowledge. Exploitation means that the agent accumulates certain experience knowledge in the continuous exploration, and uses the experience knowledge to find the way to complete the target task. ε - greedy algorithm is proposed to solve the above-mentioned exploration utilization dilemma. Based on the greedy algorithm, the idea of the algorithm is to determine an exploration factor ε, and then when selecting an action, the agent has the probability of ε to choose the action that is most conducive to the completion of the final task, and has the probability of 1 - ε to select an action randomly. The expression for this selection process is as follows:

$$\pi(s_t) = \begin{cases} \arg\max Q(s_t,a) & x \le \varepsilon \\ \qquad a\epsilon A & x > \varepsilon \end{cases}$$

Where, ε is the exploration factor, ε (1,0) when ε approaches 0, the agent only explores the environment; when ε approaches 1, the agent only uses the environment. x is a random variable, x (1,0) .If x is less than or equal to ε, the agent chooses the action that maximizes the Q value; if x is greater than ε, the agent chooses the action randomly.

### C. Algorithm Design

The algorithm consists of the following steps:

S1, initialization of strategy network, evaluation network, target strategy network, target evaluation network and network parameters, initialization of experience buffer pool, initialization of cleaning robot;

S2. The cleaning robot perceives the surrounding environment through gyroscope, lidar, camera, ultrasonic, infrared and other sensors, fuses the sensor data to judge whether there are obstacles around, the ground conditions, the surrounding garbage distribution and the status of the cleaning robot itself;

S3, the strategy neural network receives the state data of the surrounding environment, and takes the state as the input strategy neural network. The strategy neural network selects the execution strategy through calculation. Behavior strategy is based on the random process generated by current policy and random noise, and the value of behavior strategy is obtained by sampling from the random process.

S4. The cleaning robot executes the behavior strategy and converts the behavior strategy into the motor recognizable command, and then controls the motor speed, speed direction, rotation time, etc.;

S5. After the upper computer sends the instructions, the lower computer receives and executes the corresponding actions to complete the cleaning task and path planning, and gets the reward RT and the next state st + 1;

S6. Judge whether the cleaning robot has reached the garbage station and whether the action time is over. If it is terminated, turn to step S1. Otherwise, proceed to step S7;

S7. The experience storage is stored in the experience buffer pool, and the experience cache pool is used to make the States independent of each other to eliminate the strong correlation between input experiences;

S8. Randomly sampling n experiences from the experience buffer pool to calculate the loss function value of the policy value algorithm and the loss function value of the policy decision algorithm.

S9. The expected return of the current strategy is calculated by the objective evaluation neural network, and the cumulative return of each state strategy pair is estimated.

The gradient descent method is used to train the neural network. The weight coefficient of the target value network is updated by stochastic gradient descent algorithm to minimize the loss function, and the parameters of the target value neural network and the strategy neural network are calculated.

## IV. Experimental Simulation

The purpose of the experiment is to verify the path planning effect of the improved DDPG algorithm in the dynamic obstacle environment. The experimental environment is a grid map of 30 × 30, and the robot model is Turnlebot. In the experiment, the robot is used as a dynamic obstacle. The robot starts from the coordinates (2,2), and the coordinates of the target point are (26,2), the parameters of DDPG algorithm are: random action probability $\varepsilon = 0.2$, learning rate $\alpha = 0.85$, discount factor $\gamma = 0.8$, planning step number parameter $n = 25$. The premise is that the speed of dynamic robot is smaller than that of planning robot. Observe the effect of path planning of robot using DDPG algorithm strategy when there are dynamic obstacles in the environment. At the same time, compared with Q-learing and DDPG algorithm. In the process of experiment, Matlab platform is used to analyze the changes of iteration steps and average cumulative reward value of each scene in the process of robot and environment interaction. As shown in the figure is the relationship between the number of scenes and the average cumulative reward value. As shown in the figure, the convergence speed of the average cumulative reward value in the learning iteration process of this algorithm is significantly higher than that of Q-learing and DDPG algorithms, and the cumulative reward value is higher. To sum up, when the state space is large, the proposed algorithm shows good convergence.

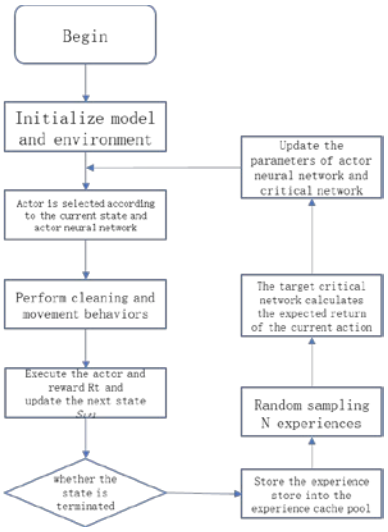
Begin
Initialize model and environment
Actor is selected according to the current state and actor neural network
Perform cleaning and movement behaviors
Execute the actor and reward Rt and update the next state $S_{t+1}$
whether the state is terminated
Store the experience store into the experience cache pool
Random sampling N experiences
The target critical network calculates the expected return of the current action
Update the parameters of actor neural network and critical network

Figure 3 Algorithm flow

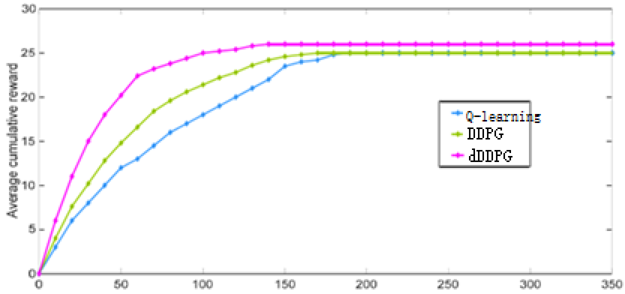

Figure 4 Comparison of algorithm experiments

## V. Conclusion

This paper presents a path planning method for mobile robot based on improved DDPG reinforcement learning algorithm is proposed. This method uses a small amount of prior knowledge to accelerate the training of deep reinforcement learning and reduce the number of trial and error. An adaptive exploration method based on the ε - greedy algorithm is used to dynamically adjust the exploration factor and reasonably allocate the exploration and utilization probability. The model can balance the problem of "exploration and utilization", and make the training process stable. Finally, the effectiveness of the algorithm is verified by experiments.

## References

[1] Khatib O . Real-Time Obstacle Avoidance System for Manipulators and Mobile Robots[J]. International Journal of Robotics Research, 1986, 5(1):90-98

[2] Akishita S, Hisanobu T, Kawamura S. Fast path planning available for moving obstacle avoidance by use of Laplace potential[C]//Proceedings of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'93). IEEE, 1993, 1: 673-678

[3] Alvarez D, Alvarez J C, Gonzalez R C. Online motion planning using Laplace potential fields[C]//2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422). IEEE, 2003, 3: 3347-3352

[4] Chen W B, Wu X B, Lu Y. An improved path planning method based on artificial potential field for a mobile robot[J]. Cybernetics and Information Technologies, 2015, 15(2):181-191.

[5] Rantanen M T. A connectivity-based method for enhancing sampling in probabilistic roadmap planners[J]. Journal of Intelligent and Robotic Systems, 2011, 64:161-178.

[6] Sánchez G, Latombe J C. A single-Query bi-directional probabilistic roadmap planner with lazy collision checking[J]. Robotics Research, 2003, 6:P403-417.

[7] Nash A, Daniel K Koenig S, Felner A. Theta: Any-angle path planning on grids[C]. Proceedings of the AAAI Conference on Artificial Intelligence; 2007. Vancouver (Canada): P1824-1830.

[8] Tarjan, Robert. "Depth-first search and linear graph algorithms." SIAM journal on computing 1.2 (1972): 146-160.

[9] Beamer, Scott, Krste Asanovic, and David Patterson. "Direction-optimizing breadth-first search." SC'12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. IEEE, 2012.

[10] Broumi, Said, et al. "Applying Dijkstra algorithm for solving neutrosophic shortest path problem." 2016 International conference on advanced mechatronic systems (ICAMechS). IEEE, 2016.

[11] Hart, Peter E., Nils J. Nilsson, and Bertram Raphael. "A formal basis for the heuristic determination of minimum cost paths." IEEE transactions on Systems Science and Cybernetics 4.2 (1968): 100-107.

[12] Stentz, Anthony. "Optimal and efficient path planning for partially known environments." Intelligent unmanned ground vehicles. Springer, Boston, MA, 1997. 203-220.

[13] Elhoseny M, Tharwat A, Hassanien A E. Bezier curve based path planning in a dynamic field using modified genetic algorithm[J]. Journal of Computational Science, 2018(25):339-350.

[14] Duguleana M, Mogan G. Neural networks based reinforcement learning for mobile robots obstacle avoidance[J]. Expert Systems with Applications, 2016, 62:104-115.

[15] Xue Y. Mobile Robot Path Planning with a Non-Dominated Sorting Genetic Algorithm[J]. Applied Sciences-Basel, 2018, 8(11).

[16] Thrun, Sebastian. "Simultaneous localization and mapping." Robotics and cognitive approaches to spatial mapping. Springer, Berlin, Heidelberg, 2007. 13-41.

[17] Hu, Gibson, et al. "A robust rgb-d slam algorithm." 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2012.

[18] Lillicrap T P, Hunt J J, Pritzel A, et al. Continuous control with deep reinforcement learning[J].Computer Science, 2015, 8(6):A187

[19] Mnih V, Badia A P, Mirza M, et al. Asynchronous methods for deep reinforcement learning[C]//International conference on machine learning. 2016: 1928-1937.