Mobile Robot Path Planning Based on Improved Reinforcement Learning Optimization

Yanshu Jing
Northeastern University
joyceyanshu@qq.com

Yukun Chen
Northeastern University
1784749212@qq.com

Minghai Jiao
Northeastern University
mhjiao@cc.neu.edu.cn

Jie Huang
Northeastern University
1824865762@qq.com

Bowen Niu
Northeastern University
916222110@qq.com

WenboZheng
Northeastern University
1214781521@qq.com

ABSTRACT

The constant parameter is usually set in adaptive function with traditional mobile robot path planning problem. Q-learning, a type of reinforcement learning, has gained increasing popularity in autonomous mobile robot path recently. In order to effectively solve mobile robot path planning problem in obstacle avoidance environment, a path planning model and search algorithm based on improved reinforcement learning are proposed. The incentive model of reinforcement learning mechanism is introduced with search selection strategy, modifying dynamic reward function parameter setting. The group intelligent search iterative process of global position selection and local position selection is exploited to combine particle behavior with reinforcement learning algorithm, dynamically adjusting the empirical parameter of the reward function by strengthening the data training experiment of Q-learning. to determine the constant parameters for simulation experiment, once the distance between the robot and the obstacle is less than a certain thresholds value, the 0-1 random number is used to randomly adjust the moving direction, avoiding the occurrence of mobile robot path matching deadlock. The study case shows that the proposed algorithm is proved to be better efficient and effective, thereby improving the search intensity and accuracy of the mobile robot path planning problem. And the experimental simulation shows that the proposed model and algorithm effectively solve mobile robot path planning problem that the parameter selection and the actual scene cannot be adapted in real time in traditional path planning problem.

CCS Concepts

• Information systems→ Information systems applications→ Decision support systems.

Keywords

Mobile robot; Path planning; Reinforcement learning; Evolutionary algorithm.

RSVT '19, October 16–18, 2019, Wuhan, China

ACM ISBN 978-1-4503-6242-9/19/03…$15.00

DOI: https://doi.org/10.1145/3366715.3366717

1. INTRODUCTION

With the development of science and technology, mobile robot is widely used in the fields of industry and life. Path planning is one of the key technologies in mobile robot path planning research and development. Robot path planning problem refers to a certain search solution optimization, such as the less work cost, the shortest choice of path, or the minimum operation time. And it also chooses an optimal or sub-optimal obstacle avoidance path from starting point to target point in the working zone. The mobile robot path planning essence is to obtain the optimal solution or feasible solution under several constraints [1]. Among the existing robot path planning algorithms, artificial potential field method, A* algorithm and ant colony algorithm are widely advanced for hot research. By using the principle of force and repulsion in electric field, the repulsion and attraction of target points and obstacles to robots are realized, and the path planning of complex terrain is realized by simulating the foraging behavior of actual ants to improve the search efficiency. Reinforcement learning is a learning algorithm between supervised learning and unsupervised learning [2]. The idea of reinforcement learning is to design the reward function so that the mobile robot can move to the targeted point more efficiently and avoid obstacles by learning from the external environment.

The mobile robot path planning optimization problem is different from the traditional path planning optimization problem. It is focus on building the optimization path model considering the environment obstacle condition. And so, main path planning algorithms in mobile robot moving environment are usually with intelligent optimization algorithms, especially as heuristic algorithm, intelligent optimization algorithm [3]. The intelligent optimization algorithms are classified as the swarm optimization algorithms, which also include bee colony algorithm [4, 5], ant colony algorithm [6], and particle swarm optimization algorithm [7]. These above algorithms have their own faults, which can solve the result of local optimal solution. In order to obtain the global optimal path planning, another intelligence swarm algorithms [8, 9] are applied for path planning optimization of mobile robot, which can obtain global solution.

As for the research of path planning of mobile robot, the bottleneck problem is how to find a global optimal path while avoiding the obstacles in the work environment, which is including building, tree, signboard, and so on. The research is focus on the potential field methodology. By assigning a potential function for each individual obstacle, the interaction of all scattered obstacles are integrated in a scalar potential surface (SPS) which strongly depends on the physical features of the mobile robot and obstacles [10]. The improved artificial potential field method put forward setting intermediate target. An external force is given to the robot to avoid robots stopping or wandering in the local minimum point [11]. And the robot path is always flaw with large path length and turning angle. The A* algorithm split path by tiny step to obtain a series of path point [12]. And multi-robot model can cope with the complexity of the problem [13]. The genetic algorithm (GA) and the particle swarm optimization algorithm (PSO) are also combined to improve result of path planning. And new variant of genetic algorithm using the binary codes through matrix for mobile robot navigation (MRN) in static and dynamic environment is presented, which path planning strategy is established using the trace theory as the optimum controller, Sylvester Law of Inertia (SLI) and matrix simulation [14]. An innovative Artificial Potential Field (APF) algorithm is presented to find all feasible paths between the start and destination locations in a discrete gridded environment. Then, an enhanced Genetic Algorithm (EGA) is developed to find the optimal path [15]. Another three level structure is proposed to obtain a feasible, safe path, which is a global optimal path in a cluttered environment [16].

The above learning mode is highly dependent on the location of obstacles in the environment. When the obstacles position is changed, the mobile robot cannot adapt to change for a short time to make new decision. We propose a mobile robot path planning method to improve the dynamic reward function parameter setting, which can solve this problem well. In addition, the global position selection and local position selection search iteration idea is applied to the model. In the process of the mobile robot gradually advances the target point, the discount coefficient is gradually reduced due to "fatigue", the learning degree is reduced, thus the adaptability is improved.

2. MOBILE ROBOT PATH PLANNING PROBLEM

2.1. A Mobile Robot Grid Graph Model

The mobile robot can "walk" in a working space, which is completely blend for the ways and obstacles. And the robot is designed to move by flexible legs or wheels, which can walk freely on the ground. So the mobile robot can move in eight directions, although it must move along with uncertain ways avoiding obstacle from begin point to destination point. The working space of mobile robot can be defined as grid graph, shown as Figure 1.

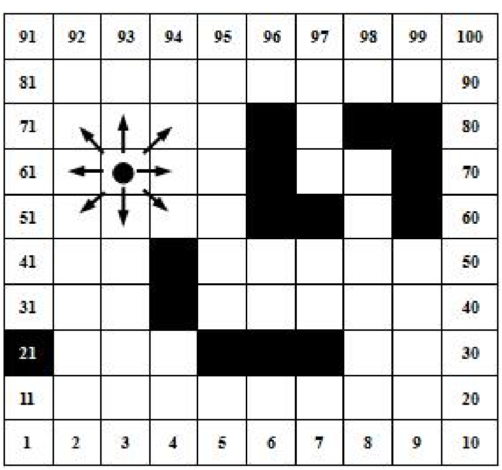


Figure 1. Grid graph model of mobile robot moving.

The grid graph is described on the following assume definition:

a) A mobile robot is defined as a particle which is no quality or size, called mobile particle robot;
b) Robot working space is defined as two dimensional grid graph model;
c) In the robot working space, the height of the obstacle is ignored. The obstacle occupy some grids in grid model;
d) Particle robot moves a grid step each time, which has an eight kinds of direction step to move along;
e) Particle robot collides the obstacle several times, it must avoid obstacle in working space.

Definition1 The mobile robot working space is described as a two-dimensional grid which is divided as row grid and column grid. The grid is encoded as sorted numbers. The numbers of grid are described as the follows.

$$\begin{cases} N_x = \dfrac{X_{\max}}{R} \\ N_y = \dfrac{Y_{\max}}{R} \end{cases} \quad (1)$$

The grid graph is a two-dimensional working map, built as $n \times n$ blocks. And the map is defined as mathematic model including robot moving parameter R, $X_{\max}$, and $Y_{\max}$. The parameter R is robot moving one step size. The N_x and N_y are separately the length and the width of working space map. The $X_{\max}$ and $Y_{\max}$ are separately the maximum length and the maximum width. For the optimization research, the parameter value of R, $X_{\max}$, $Y_{\max}$ is respectively set as 1, 10, 10 to focus on solution method. So the robot working space grid graph of 10×10 size is divided as 100 grid sizes. The sorted numbers are marked in the figure 1.

2.2. Path Planning Optimization Model

The goal of the mobile robot path planning optimization is not only to find the most optimal path from starting point to target point, but also to avoid collision with the obstacle blocks. The optimization problem has two parts of object function: the minimum path function and minimum collision punishment function. The path function is defined as total minimum distance path from start point to destination point. And the collision punishment is also defined as collision coefficient, which is indicated as the collision number that the mobile robot encounters the obstacle.

The model of path planning optimization for mobile robot in working space is as the following:

$$\min f_{path}(X),\ X \in R^n \quad (2)$$

$$\min\ f_{punish}(X), X \in R^n \quad (3)$$

The optimization function f_{path} is the total path distance in grid working space map. Shown as the following:

$$f_{path}(X) = \sum_{i=1}^{n} D_i(X) \ ,i=1,2,\ldots n \quad (4)$$

D_i is the path distance of mobile robot step i which is described as the one grid step, and n is the total number of grid steps. The optimization function f_{punish} is given as the following:

$$f_{punish}(X) = \sum_{i=1}^{n} \sum_{k=1}^{m} H_{ik}(X) \ ,i=1,2,\ldots n,\ k=1,2,\ldots m \quad (5)$$

H_{ik} is the punishment coefficient of obstacle k collided by robot step grid i. The parameter n is the total numbers of grid, and m is the total numbers of obstacle. Considering the consistency of solving problem, the mobile robot path planning is built as a constraint optimization model. Shown as the following:

$$\min f_{path}(X),\ X \in R^n \tag{6}$$

s.t.

$$\min f_{punish}(X) \leq N \tag{7}$$

Parameter N is the number of mobile robot colliding with the obstacle, which is given by real empirical value.

3. IMPROVED REINFORCEMENT LEARNING OPTIMIZATION ALGORITHM FOR SOLVING PATH PLANNING MODEL

3.1. Improved Reinforcement Learning Optimization Algorithm

The mobile robot can walk in the working space of $N_x \times N_y$ grid map. The model of shortest path optimization from start location to the target location is resolved by random search method. The search process is that robot can travel in shortest time in working space. But the larger the number n is, the longer the problem solving time is. So the improved reinforcement learning algorithm is introduced for accelerating search process.

The reinforcement learning is the reward and punishment mechanism. When the robot particle searches a "good" direction, it gets a reward score. On the contrary, the robot particle gets a negative score for the "bad" search punishment. The search reward function is formulated as the following:

$$Q(s, a) = Q(s, a) + \alpha[R(s, a) + \gamma max Q'(s', a') - Q(s, a)] \tag{8}$$

The above equation is the Bellman reward function [17] avoiding the obstacles. The function Q is an action value function. The function R is a reward function. The parameter α is the learning rate which is a training value in [0, 1]. And the parameter γ is the discount rate which is also a training value in [0, 1], indicating the extent to which future remuneration will be obtained. The function Q' is the highest Q value between possible actions from the new state of s'.

The particles robot seeks the optimal solution in target search space. The improved reinforcement robot swarm optimization algorithm is formulated as the following:

$$C_i^t = r_1 {P_{local}}_i^t + r_2 {P_g}^t \tag{9}$$

$$X_i^{t+1} = Q_i(s,a) + \beta^t \cdot \left| C_i^t - X_i^t \right| \cdot \ln\left(\frac{1}{L_i^t}\right) \tag{10}$$

The equation (9) is the idea of the swarm decision making mechanism, that the parameter ${P_{local}}_i^t$ is the local best position with single particle i after t iteration, and P_g^t is the global best position of all particles after t iteration. The parameter r_1 and r_2 are the random number in the interval [0, 1]. The equation (10) is the mobile robot searching process of path planning with reinforcement learning swarm intelligent policy. The parameter L_i is random number in the interval [0, 1]. And the β^t is the contraction expansion coefficient in iteration process, which is formulated as the following:

$$\beta^t = u - \frac{(u-v) \cdot t}{t_{\max}} \tag{11}$$

The formula is a dynamic punishment function for iteration, which value is changed from u to v with the increasing t times. According to experience setting, the u value is set as 1, and v value is set as 0.5. The $t_{\max}$ is the maximal generation.

3.2. Algorithm Flow for Robot Path Optimization

The algorithm flow of improved reinforcement learning optimization for solving mobile robot path planning problem is described as the follows. Pseudo-code is sketched in Algorithm 1.

Algorithm 1 Improved Reinforcement Learning Optimization Algorithm

Input: Q: initialized with Q table value to zero, α: initialized learning rate in [0, 1], γ: initialized discount rate in [0, 1], N_x, N_y : initialized sizes with x and y dimension for robot grid model.

Output: Optimizaton path solution f^* with robot grid model from start to end.

```
Select initial R, random α, and γ; //initialization
Calculate initial Q(s,a) using equation (8);
Define objective function f(x1,x2,...,xn) using equation (4);
Define punishment coefficient H_ik in equation (5);
Do initialization of population P_i of n particles in random positions;
Calculate C_0, X_0;
for t=1:iterNum do
    X_i←P_i // encoding X with ith particle position
    Calculate Q_i^t(s,a) using equation (8);
    Calculate C_i^t using equation (9);
    Calculate X_i^{t+1} using equation (10);
    Calculate f_i^{t+1} using equation (6) and (7);
    if f(X_i^{t+1})<f(X_i^t) then
        P_local_i^t=P_i^t;
    end if
    if f(X_i^t)<f(X_j^t) and i ≠ j, j =1, 2, ..., n then
        P_gi^t=P_i^t;
    end if
    P_i^{t+1}←X_i; // decoding X_i to ith particle position
    Q_i^{t+1}(s,a)←Q_i^t(s,a); // updated reinforcement learning Q value
end for
f* = Σ_{i=1}^n D(X_i*);
Return f*;
```

3.3. Local Search and Global Search for Update Q Values

The proposed improved Q-learning intends to initialize the Q values search local "good" and global "best" mechanism. Therefore, following from the population initialization, the fitness of each particle, representing the Q value, is evaluated using equation (8). The highest Q value is the P_g search position, and the highest Q value is also the P_{local} search position. Subsequently, the population will undergo the global and local search process, integrated the particle global position and particle local position, shown in equation (9) in order to search for the best feasible random path.

From the current search position, the selection of next position relies on equation (10). The fitness value of each particle in a new position, which corresponds to the Q value, utilized as the equation (6) for object function optimization and equation (7) for constraint function. The Q value is assigned to a negative values as a penalty when the particles search interaction with an obstacle occurs. Hence, this position will be marked for negative scores and new position will be generated for Q value. Therefore, the fitness values of the entire population are evaluated again. When a better free space is found, the position will be the particle global position P_g. The process of search is repeated for maximum iteration. Through the repetitive steps, the highest Q value along the path can be found within a shorter computational time.

3.4. Exploration and Exploitation Process by the Q-learning

The Q values are initialized in the Q-table when the particle population search begin. And then, the Q-learning will take place subsequently. The exploration and exploitation processes at this stage are similar to the classical Q-learning, except that the particle starts with an updated Q-table with different Q values. The agent will start from the starting position and continue its search by select the P_g and P_{local} position status. Since all the next possible search will have different Q values, the selection of next state will not be completely random as compared to classical Q-learning. After proceeded to the next state, Q value of the previous state will be updated based on equation (8) and equation (10). The search process continues until the target position is found.

4. EXPERIMENT DESIGN AND RESULT ANALYSIS

4.1. Experiment Design

The test environment is a robot working space with the real obstacles. The mobile robot can walk on the indoor and outdoor ground, avoiding the obstacles. The robot can get an optimization path from start position to target location.

The test experiment is designed just for grid graph model, which model is solved by improved reinforcement learning swarm search optimization algorithm. The test experiment environment of model is set as the follows.

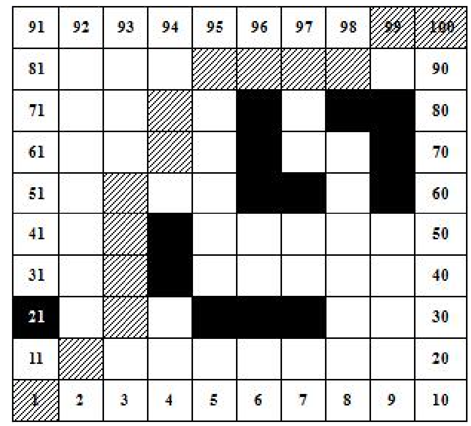
Table 1. Experiment environment set.

Item	Parameter
Grid size	10╳10
Obstacle numbers	5
Moving direction	8
Moving step size	1
Start block index	1
Target block index	100

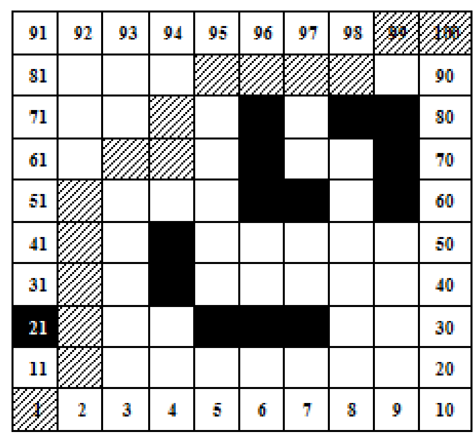
The algorithm is implemented on the windows operation system. The test experiment on above platform is described as the following: The CPU is 4.3GHz of Intel core i7, the memory size is 32G, the hard disk storage size is 2T, and the program development language is C++.

4.2. Experiment Result and Performance Analysis

The experiment parameter of α and γ is an empirical value which can train for modifying the set. We give the parameter value of α=0.8, γ=0.7 and α=0.5, γ=0.5 for the 10╳10 robot grid model. The optimization result is as the follows.



(a) α=0.8, γ=0.7



(b) α=0.5, γ=0.5

Figure 2. Mobile robot optimization result with α and γ.

The result is the comparison with Q-learning parameter α and γ. When The learning rate parameter α=0.8 or α=0.5 is individually set, the discount rate parameter γ=0.7 or γ=0.5 is individually set too. And the path search result of (a) is shorter than the search result of (b). We adjust the random value in interval [0, 1] as the new iterative search direction once the mobile robot collide the obstacles. For example, the random number is generated in [0, 0.25], the robot particle up moves in the grid map model. At the same time, the robot particle down moves with the number in [0.25, 0.5], the robot particle left moves with the number in [0.5, 0.75], and then the robot particle right moves with the right moves with the number in [0.75, 1]. Other direction, such as left-up, right-up, left-down, right-down, is adjusted according the formulate (8) as the reward or punishment score.

We propose the Improved Reinforcement Learning Optimization (IRLO)algorithm as the performance analysis. The experiment performance analysis is implemented for performance comparison with Quantum Particle Swarm Optimization (QPSO) algorithm [18], and basic Particle Swarm Optimization (PSO) algorithm in mobile robot path planning problem. All three algorithms are also initialized as 20 group particles population sizes, which are carried out for 1000 times running iteration in same experiment environment.

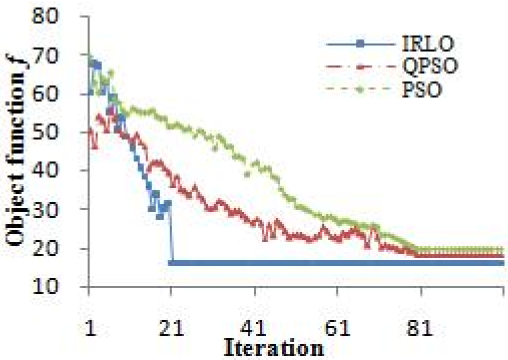


Figure 3. The convergence behavior of three algorithms.

Figure3 presents the graph of average total steps used over iterations in order to observe the convergence behavior of all algorithms. As shown in this figure, the IRLO algorithm is superior to the QPSO and PSO algorithm in terms of convergence rate. This is due to the Q-table of IRLO is based on Q-learning action between the position of the next action state and the target position. This has somehow guided the mobile robot to move from initial position to target position more efficiently since the information about the distance to the target position is available.

The experiment results of optimization algorithms are shown as the following table.

Table 2. Optimization results with three algorithms.

Algorithm	f	f_{path}	T(ms)
PSO	22.628	19.932	132
QPSO	20.398	18.273	77
IRLO	18.545	16.235	38

From the experiment result, the improved reinforcement learning swarm optimization algorithm overcomes other effective artificial intelligent algorithms. The running time is great less than the other algorithms. The main reason is that the IRLO is excellent performance for solving constrained continuous optimization problem.

5. CONCLUSION

In order to satisfy the robot path planning needs, how to choose the shortest path and avoid the obstacles becomes the most important solution for proposed path planning problem. The improved reinforcement learning swarm intelligent optimization algorithm is proposed to solve the path planning problem. The incentive model of reinforcement learning is introduced with planning problem, improving the dynamic parameter setting of the reward function. The group intelligent search iterative process is exploited to combine particle behavior with reinforcement learning algorithm, dynamically adjusting the empirical parameters of the reward function by strengthening the data training experiment of learning. The experiment platform is built on the hardware device of mobile robot and test environment. The results show that the improved IRLO algorithm is effective and efficient for mobile escort robot moving path planning problem.

6. ACKNOWLEDGMENTS

This work is supported by the thirteenth batch of College Students' Innovation Training Program in Northeastern University (2019), by “the Fundamental Research Funds for the Central Universities” (project 201910145161), by Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry under Grant No. 49-1.

7. REFERENCES

[1] Huo, F. C., Chi, J., Huang, Z. J., Ren, L., Sun, Q. J., Chen, J. L. 2018.Review of path planning for mobile robots. *Journal of Jilin University* (*Information Science Edition*) 36(6), 639-647.

[2] Peters, J., Schaal, S. 2008.Reinforcementlearningofmotorskillswithpolicygradients. *NeuralNetworks*21(4), 682-697.DOI=https://doi.org/10.1016/j.neunet.2008.02.003.

[3] Mac,T. T., Copot,C., Tran,D. T., Keyser,R. D. 2016. Heuristic approaches in robot path planning: A survey.*Robotics and Autonomous Systems*86, 13-28.DOI= https://doi.org/10.1016/j.robot.2016.08.001.

[4] Li,Z. J. 2012. Application of artificial bee colony algorithm in path planning of mobile robot.*ComputerSimulation*29(12),247-250.

[5] Contreras-Cruz,M. A., Ayala-Ramirez, V. U., Hernandez-Belmonte, H. 2015. Mobile robot path planning using artificial bee colony and evolutionary programming.*Applied Soft Computing* 30, 319-328.DOI=https://doi.org/10.1016/j.asoc.2015.01.067.

[6] Qu,H.,Huang,L. W.,Ke,X. 2015. Research of improved ant colony based robot path planning under dynamic environment.*Journal of University of Electronic Science and Technology of China* 44(2), 260-265.

[7] Weng,L. G.,Ji,Z. Z.,Xia,M.,Wang,A. 2014. Robot path planning based on improved multi-objective particle swarm.*Journal of System Simulation*26(12), 2892-2898.

[8] Feng, X., Ma, M. Y., Shi, Y.,Yu,H. Q. 2013. Path planning for mobile robots based on social group search algorithm.*Journal of Computer Research and Development* 50(12), 2543-2553.

[9] Liang, X. D., Lin, N.,Chen, H. N. 2016. Mobile robot dynamic path planning based on bacterial foraging behavior.*Chinese Journal of Scientific Instrument* 37, 1316-1324.

[10] Bayat, F.,Najafinia, S.,Aliyari, M. 2018. Mobile robots path planning: Electrostatic potential field approach.*Expert Systems with Applications* 100, 68-78.DOI=https://doi.org/10.1016/j.eswa.2018.01.050.

[11] Xu, F. 2016. Research on robot obstacle avoidance and path planning based on improved artificial potential field method.*Computer Science* 43(12), 293-296.

[12] Sun, W., lv, Y. F., Tang, H. W.,Xue, M. 2017. Mobile robot path planning based on an improved A* algorithm.*Journal of Hunan University*(*Natural Sciences*) 44(4), 94-101.

[13] Ma, Y., Wang, H.,Xie, Y.,Guo, M. 2014. Path planning for multiple mobile robots under double-warehouse.*Information*

Sciences 278, 357-379.DOI= https://doi.org/10.1016/j.ins.2014.03.058.

[14] Patle, B. K.,Parhi, D. R. K.,Jagadeesh, A.,Kashyap, S. K. 2017. Matrix-Binary Codes based Genetic Algorithm for path planning of mobile robot. *Computers and Electrical Engineering*, 1-22.DOI= https://doi.org/10.1016/j.compeleceng.2017.12.011.

[15] Nazarahari, M.,Khanmirza, E.,Doostie, S. 2019. Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm.*Expert Systems with Applications* 115, 106-120.DOI= https://doi.org/10.1016/j.eswa.2018.08.008.

[16] Mac, T. T.,Copot,C., Tran, D. T., Keyser, R. D. 2017. A hierarchical global path planning approach for mobile robots based on multi-objective particle swarm optimization.*Applied Soft Computing* 59, 68-76.DOI= https://doi.org/10.1016/j.asoc.2017.05.012.

[17] Hester, T., Vecerik, M., Pietquin, O., Lanctot, M., etc. 2018.Deep Q-learning from Demonstrations. In: *Proceedingsof the Thirty-Second AAAI Conference on Artificial Intelligence*, pp. 1–12. AAAI Press, New Orleans, Louisiana, USA.

[18] Jiao, M. H., Chen, X. B., Liu, H. Q., Cheng, Y. R., Zhang, H. 2018.Research on quantum particle swarm optimization in mobile robot path planning for aged service.In: *Proceedingsof the30st Chinese Control and Decision Conference*, pp. 2034-2039. IEEE, Shenyang, China.DOI:10.1109/CCDC.2018.8407460.