

A New Hybrid Method in Global Dynamic Path Planning of Mobile Robot

X.R. Song, S. Gao, C.B. Chen, K. Cao, J.R. Huang

Xiaoru Song*, Song Gao, Kai Cao, Jiaoru Huang

School of Electronic and Information Engineering,

Xi'an Technological University

No.2 of Xuefu Middle Road, Xi'an, Shaanxi, China

*Corresponding author: masha0422@163.com

545313193@qq.com, 602507619@qq.com, 674330497@qq.com

Abstract: Path planning and real-time obstacle avoidance is the key technologies of mobile robot intelligence. But the efficiency of the global path planning is not very high. It is not easy to avoid obstacles in real time. Aiming at these shortcomings it is proposed that a global dynamic path planning method based on improved A* algorithm and dynamic window method. At first the improved A* algorithm is put forward based on the traditional A* algorithm in the paper. Its optimized heuristic search function is designed. They can be eliminated that the redundant path points and unnecessary turning points. Simulation experiment 1 results show that the planned path length is reduced greatly. And the path transition points are less, too. And then it is focused on the global dynamic path planning of fusion improved A* Algorithm and Dynamic Window Method. The evaluation function is constructed taking into account the global optimal path. The real time dynamic path is planning. On the basis of ensuring the optimal global optimization of the planning path, it is improved that the smoothness of the planning path and the local real-time obstacle avoidance ability. The simulation experiments results show that the fusion algorithm is not only the shorter length, but also the smoother path compared the traditional path planning algorithms with the fusion algorithm in the paper. It is more fit to the dynamics of the robot control. And when a dynamic obstacle is added, the new path can be gained. The barrier can be bypass and the robot is to reach the target point. It can be guaranteed the global optimality of the path. Finally the Turtlebot mobile robot was used to experiment. The experimental results show that the global optimality of the proposed path can be guaranteed by the fusion algorithm. And the planned global path is smoother. When the random dynamic obstacle occurs in the experiment, the robot can be real-time dynamic obstacle avoidance. It can re-plan the path. It can bypass the random obstacle to reach the original target point. The outputting control parameters are more conducive to the robot's automatic control. The fusion method is used for global dynamic path planning of mobile robots in this paper. In summary the experimental results show that the method is good efficiency and real-time performance. It has great reference value for the dynamic path planning application of mobile robot.

Keywords: Mobile robot, path planning, DWM algorithm.

1 Introduction

Path planning is one of the most basic and important issues of mobile robots [2]. It is one of the independent acts to complete the task. The mobile robot path planning is a more difficult to solve the problem especially in the dynamic environment. It requires the robot process as far as possible not to deviate from the specified path, unless the obstacles encountered when the movement had to bypass the walk. But after avoiding the obstacle, the mobile robot must return to the original path until it reaches the destination [3, 8, 15, 18, 21].

Aim at this problem the dynamic window method is used in this paper. It has a good obstacle avoidance capability in the dynamic environment [6]. But this method does not satisfy the global optimal path planning. The node-based A* algorithm is suitable for global path planning. It has the advantages of simple data, small calculation compared to the biological inspired GA, PSO, etc [7] [9]. But when the environmental space increases, it needs large storage space, low efficiency, and poor real-time decision-making [11]. So the path curvature of the traditional A* algorithm is discontinuous. It will cause the movement parameters jump at the turning point. These are not conducive to the robot control problems [1]. Aim at these problems the algorithm is improved. It makes the path smoother on the basis of the overall optimization of the planning path [12]. so a global dynamic path planning method is proposed that it is combining improved A* algorithm and dynamic window method in the paper. It is proposed the evaluation function taking into account the global optimal path. And the real-time path planning is carried out by using dynamic window method based on the evaluation function. It would make the path smoother on the basis of ensuring the overall optimization of the planning path. At the same time it improves the local obstacle avoidance.

2 The construction of advanced A* algorithm in mobile robot global path planning

The traditional A* algorithm is a kind of global path planning algorithm [9]. It is a typical heuristic search algorithm in artificial intelligence. Aim at solving the problem of a large number of invalid search paths often generated in path planning [24]; the heuristic function is used to improve the search efficiency.

A* algorithm design core is its evaluation function:

$$f(n) = g(n) + h(n) \quad (1)$$

Where, $f(n)$ is the evaluation value of each node. It may be searched in the path plan [14]. The cost evaluation value of the starting point goes through the node n to the target point. $g(n)$ is the actual evaluation value from the starting node to the current node. $h(n)$ is the evaluation value from the current node to the target node (Heuristic function). The $h(n)$ can directly affect the performance of the algorithm. So it tries to approach the actual assessment $h^*(n)$ of the path planning problem. It is needed [20]. But it is also less than $h^*(n)$.

2.1 The design of the optimization heuristic function

Assuming that the $H(n)$ is actual value of the current node to the target node. When $h(n) < H(n)$, more nodes is low efficiency, but it can search for the optimal path. When $h(n) = H(n)$, it is ideal, searching along the shortest path and being the most efficient. When $h(n) > H(n)$, less nodes, high efficiency, but it is often difficult to search for the optimal path [23].

The traditional A* algorithm uses the Manhattan distance or Euclidean distance to define the heuristic function [16], namely

$$h_M(n) = |n_x - g_x| + |n_y - g_y| \quad (2)$$

$$h_E(n) = \sqrt{(n_x - g_x)^2 + (n_y - g_y)^2} \quad (3)$$

Where, n_x, n_y, g_x, g_y is the current node and target point.

A* algorithm searches the 8-neighborhood grid of the current node each time [25]. So these two heuristic function designs are not the best choice taking into account the two-wheeled robot control. Therefore, a heuristic function that is closer to the actual value $H(n)$ is designed with Manhattan distance and Euclidean distance in the paper.

$$h(n) = \begin{cases} \sqrt{2}d_x(n) + d_y(n) - d_x(n) & d_y(n) > d_x(n) \\ \sqrt{2}d_y(n) + d_x(n) - d_y(n) & d_x(n) > d_y(n) \end{cases} \quad (4)$$

Where, $d_x(n) = |n_x - g_x|$ $d_y(n) = |n_y - g_y|$, the rest parameters are the same as in the above formulas (2) and (3).

2.2 The proposition of key point selection strategy

A* algorithm planning optimal path contains a lot of redundant and unnecessary turning points. It is not conducive to the control of the robot [25]. In order to solve this problem, a key node selection strategy is proposed in the paper.

(1) Extract key turning points.

Starting from the second node of the planning path, if the current node is on the same line as the previous node and its parent node, the previous node is redundant. The redundant point is deleted. The path would update. All the path nodes would traverse. A path sequence will eventually get that contains only the starting point, the turning point, and the end point.

(2) Delete the redundant turning point.

After extracting the turning point there may still be redundant points in the trajectory. Assuming that path point after extracting the key turning point is $\{P_k | k = 1, 2, L, n\}$, connect P_1P_3 , if P_1P_3 do not pass the obstacle, namely the line distance P_1P_3 from the obstacle is greater than the preset threshold, then continue to connect P_1P_4 , until the $P_1P_k (k = 3, 4, L, n)$ through obstacles, then connect P_{k-1} and P_1 , while delete the middle redundant points, update the path; repeat the above procedure from the node P_2 until there is no redundant turning point in the path. The experimental results of the key point selection strategy are shown in Fig.1. It can be seen that the application of key selection strategy algorithm is better than traditional A* algorithm.

Fig.1 (a) is the original path sequence of the traditional A* algorithm. Fig.1 (b) is the path sequence obtained by advanced A* algorithm to extract the critical turning point. Fig.1 (c) is the final path sequence by advanced A* algorithm removing the redundancy turning point. Simulation experiment 1 results show that the planned path length is reduced greatly. The path length is from 11.455m to 10.946m. And the path transition points are less greatly.

3 The establishment of mobile robot motion model in dynamic window method (DWA)

The dynamic window method has the dynamic obstacle avoidance capability of the robot in the dynamic environment [17]. The multi-group velocity can be sampled in the velocity space (linear velocity v and angular velocity w) [26]. The trajectory of the robot is simulated at these speeds for fixed time interval. After acquiring multiple sets of trajectories the corresponding speed of the optimal trajectory is selected. It is driving the robot movement according to the evaluation index.

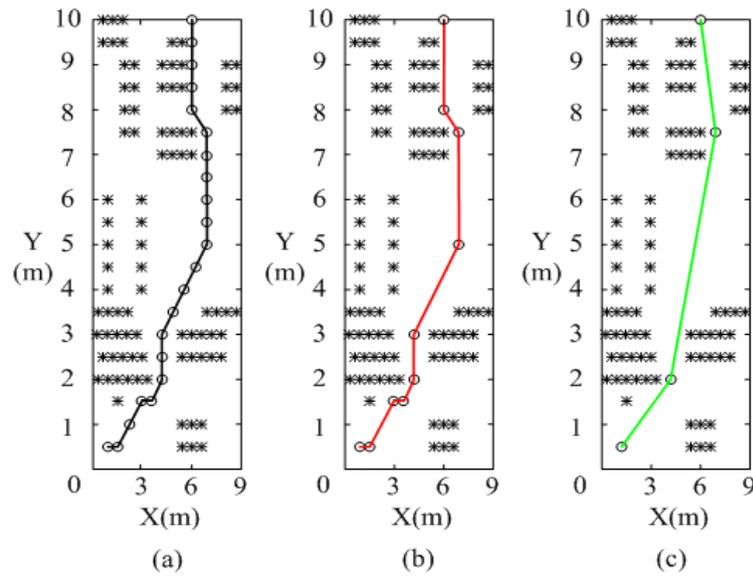


Figure 1: The experimental result of the key point selection strategy (where "-" means the planning path, "***" means the obstacle. "o" means the node.)

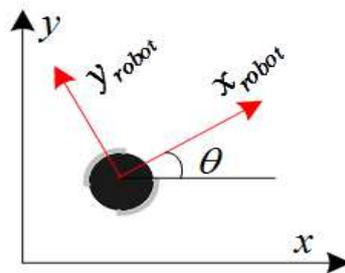


Figure 2: The robot motion diagram

3.1 The establishment of the robot movement model

In the dynamic window algorithm it is need that simulates the movement of the robot trajectory. So it is necessary that the robot movement model is known. It is assumed that the trajectory of the move robot is a circular arc. The arc trajectory is represented by (v_t, w_t) . When the robot trajectory is calculated, consider two adjacent moments first. Due to the short distance of motion between adjacent moments (general chute sampling periods ms), the trajectories of the two adjacent points can be viewed as a straight line. That is, the robot moves along the axis. The motion of the robot is shown in Fig. 2.

Therefore, when the robot moves in the Omni directional direction, the trajectory motion model of the two adjacent moments $t + 1$ relative to t is always as follows equation (5):

$$\begin{cases} x = x + v_x \Delta t \cos(\theta_t) - v_y \Delta t \sin(\theta_t) \\ y = y + v_x \Delta t \sin(\theta_t) - v_y \Delta t \cos(\theta_t) \\ \theta_t = \theta_t + w_t \Delta t \end{cases} \quad (5)$$

Where, x and y is the robot projection distance of axis x and y . v_x and v_y is the robot moving speed of axis x and y . θ is the angle between the robot and the axis x . Δt is the two adjacent moment.

3.2 Speed sampling

Based on the above-mentioned robot's trajectory motion model, the trajectory can be deduced according to the speed. So sample a lot of speed, calculate the trajectory, and evaluate the track is good or not is only needed [25]. How to sample speed is also a core of DWA. In the two-dimensional speed space (v, w) there exist endless groups of speed. According to the limitations of the robot itself and the environment, the robot sampling speed is controlled in a certain dynamic range.

The speed limit of mobile robot between its own maximum and minimum speed:

$$V_m = \{v \in [v_{\min}, v_{\max}], w \in [w_{\min}, w_{\max}]\} \quad (6)$$

The impact of mobile robot performance by the motor: There is a maximum acceleration and deceleration limit due to the limited motor torque. So there is a dynamic window in the period of sim-period of the mobile robot trajectory forward simulation. The speed is the robot can actually reach:

$$V_d = \{(v, w) \mid \begin{cases} v \in [v_c - \dot{v}_b \Delta t, v_c + \dot{v}_a \Delta t] \wedge \\ w \in [w_c - \dot{w}_b \Delta t, w_c + \dot{w}_a \Delta t] \end{cases} \} \quad (7)$$

Where v_c and w_c are the current speed of the robot. The other speeds are corresponding to the maximum acceleration and the maximum deceleration.

(3) About mobile robot security considerations

In order to allow the robot to stop before the obstacles encountered, so speed range is needed under the maximum deceleration:

$$V_a = \{(v, w) \mid v \leq \sqrt{2 \cdot \text{dist}(v, w) \cdot \dot{v}_b} \wedge \sqrt{2 \cdot \text{dist}(v, w) \cdot \dot{w}_b}\} \quad (8)$$

Where $\text{dist}(v, w)$ is the closest distance from the obstacle according to velocity (v, w) .

3.3 Design the evaluation function selection

In the velocity space, there are several groups of sampling speed is feasible, so design evaluation function selects the optimal trajectory. The criteria of the design evaluation function are the robot as far as possible to avoid obstacles during local navigation, and towards the target quickly forward. The design evaluation function is:

$$G(v, w) = \sigma \left(\begin{array}{l} \alpha \cdot heading(v, w) + \\ \beta \cdot dist(v, w) + \\ \gamma \cdot velocity(v, w) \end{array} \right) \quad (9)$$

Where $heading(v, w)$ is the azimuth angle evaluation function, it indicates the azimuth deviation (angular deviation) between the end direction of the simulated trajectory and the target at the current set sampling speed (v, w) . $dist(v, w)$ is the closest distance of the robot itself to the obstacle at speed (v, w) . If there is no obstacle on the track, it is set as a constant. $velocity(v, w)$ is the evaluation of the current speed size; $\sigma(\cdot)$ is the smoothing function, α, β, γ is the weight of the three factors. The deviation is smaller, the evaluation is higher value. The distance is greater, it is the higher evaluation value. The higher speed, and the higher evaluation value. These three indicators are normalized and then weighted.

4 The proposition of advanced A* algorithm fusion DWA algorithm path planning method

Dynamic window planning path method is online real-time, with good local dynamic barrier capability. However, the requirements of global optimal path planning are missed. It is a fatal problem; it will fall into the local optimum. So the advanced A* algorithm for global path planning, fusion of dynamic window method for local obstacle avoidance, it ensures the global optimization of the dynamic planning path.

Therefore, in order to avoid the dynamic window method into the local optimal, a dynamic window evaluation function to estimate the global optimal path is proposed:

$$G(v, w) = \sigma \left(\begin{array}{l} \alpha \cdot PHeading(v, w) + \\ \beta \cdot dist(v, w) + \\ \gamma \cdot velocity(v, w) \end{array} \right) \quad (10)$$

Where $PHeading(v, w)$ is the azimuth deviation between the simulated trajectory end point and the current target point at the current speed (v, w) . The current target point is the global optimal path sequence point; it is nearest to the current point in the robot forward direction. This evaluation function allows the local path planning to follow the global optimal path contour, thus ensuring the global optimal. This algorithm is shown in Fig. 3.

5 Algorithm simulation verification

In order to verify the validity of the fusion algorithm designed in this paper, two experiments were carried out in the MATLAB simulation environment, the grid map scene (10m \times 10m, grid spacing 0.5m), and the coverage of the obstacle was 23%. In the simulation verification experiment, the starting point coordinates are (1.25 m, 0.75 m), the target point coordinates are (5.75 m, 9.75 m). Simulation experiment 2 adds dynamic obstacles based on the experiment 1 to verify the dynamic obstacle avoidance performance of this algorithm.

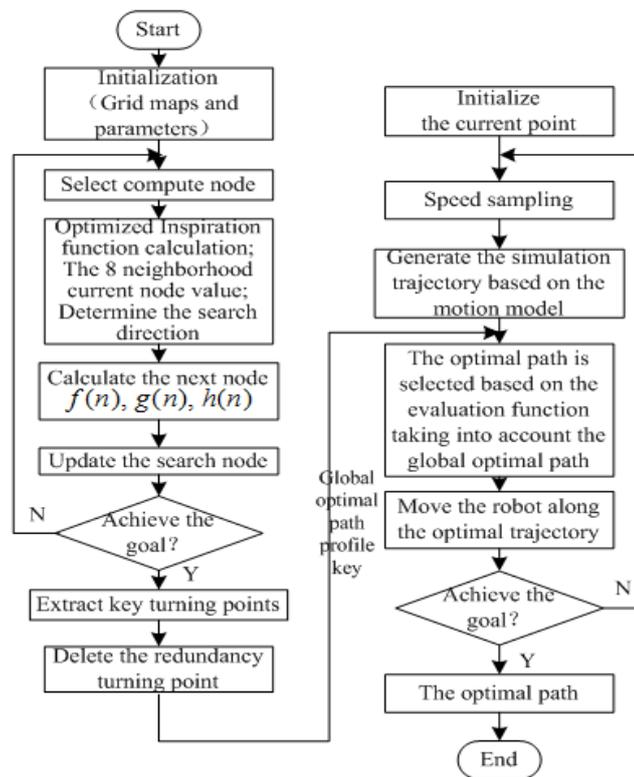


Figure 3: Fusion algorithm for mobile robot global dynamic path planning

Simulation 1: In the above mentioned simulation environment, the mobile robot uses the traditional A* algorithm, the advanced A* algorithm adding a key extraction strategy, the dynamic window algorithm and the fusion algorithm proposed in this paper are used to validate the path planning. The simulation results of the algorithm are shown in Fig.4.

In Fig.4, the blue box is the static obstacle, the black origin is the experimental mobile robot, and green solid line is the robot simulation path. It can be seen from Fig. 4 (a) that the traditional A* algorithm can plan a global optimal path. The length is 11.455m. In Fig. 4 (b), the advanced A* algorithm of key selection strategy can improve the path obviously, not only the redundant points in the traditional A* algorithm path are deleted, but the path length is better than the traditional A* algorithm. The path length is 10.946m. In Fig. 4 (c), the dynamic path method is more smooth and suitable for automatic control of the robot than the A* algorithm, but the path of the dynamic window method is non-global and its path length is about 14.439 m. In Fig. 4 (d), the fusion algorithm uses the evaluation function, the global optimal path is considered, so it can guarantee global optimization compared with the dynamic window algorithm. Compared with the traditional A* algorithm, the fusion algorithm makes the path length greatly optimized, the path length is about 11.126m. The planning path is smoother, so the curvature change of the path is continuous. These are more in line with the dynamic control of the robot.

Simulation 2: Based on the simulation experiment 1, the dynamic obstacle (5.75, 6.0) is added, the real-time dynamic obstacle avoidance simulation results of the mobile robot are shown in Fig. 5.

In Fig. 5, the red point is the artificial dynamic obstacle, the purple solid line is the simulation

path of the robot, and the rest is consistent with the simulation in Figure 4. From Fig. 5 (a) and Fig. 5 (b), it can be seen that the traditional A* algorithm and the advanced A* algorithm cannot avoid the sudden dynamic obstacle. In Fig. 5 (c), the path delimited by the dynamic window method, it can avoid the dynamic obstacle, but it is not global optimal. In Fig.5 (d), the fusion algorithm in this paper can avoid the dynamic obstacle, bypassing the obstacle to the target point, and the planned path to maintain global optimality; it achieves the expected purpose of global dynamic path planning for mobile robots.

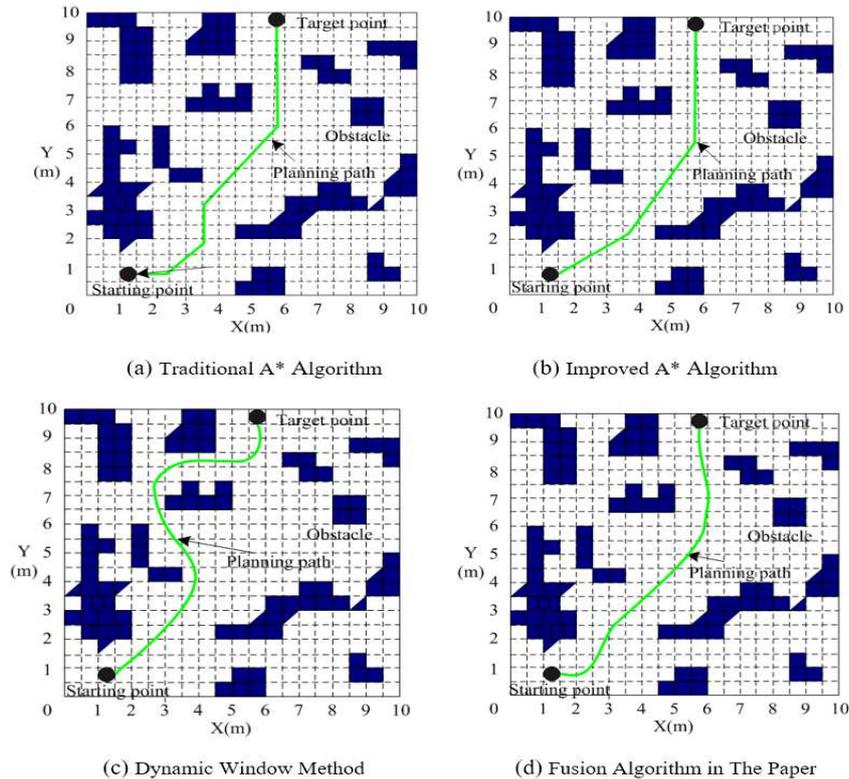


Figure 4: Various algorithmic path planning results

6 Implementation and experimental results

The experimental platform used in this article is the TurtleBot 2 two-wheel-drive robot produced by iRobot. The robot is equipped with the Kinect depth sensor developed by Microsoft as the visual sensor, Korea's Yujin Kobuki mobile base, it is shown in Fig. 6.

In this experiment, we choose our school international science and technology cooperation base in Shaanxi Province, autonomous system and intelligent control international joint research center of unmanned system laboratory. The experimental environment of this project was about $10m \times 10m$. The experimental environment was randomly, it is shown in Fig.7.

Robot uses Kinect sensor to obtain the actual scene of the three-dimensional point cloud data, combined with the odometer and other information through the SLAM (...) algorithm, it can obtain grid map in the interior scene. After the robot receives map of SLAM constructed, the navigation algorithm is invoked, that is, the advanced A* path planning algorithm for global path walking. In the process of obstacle avoidance, when the robot encounters the dynamic obstacle, calls the local DWA algorithm, the real-time dynamic obstacle avoidance is performed,

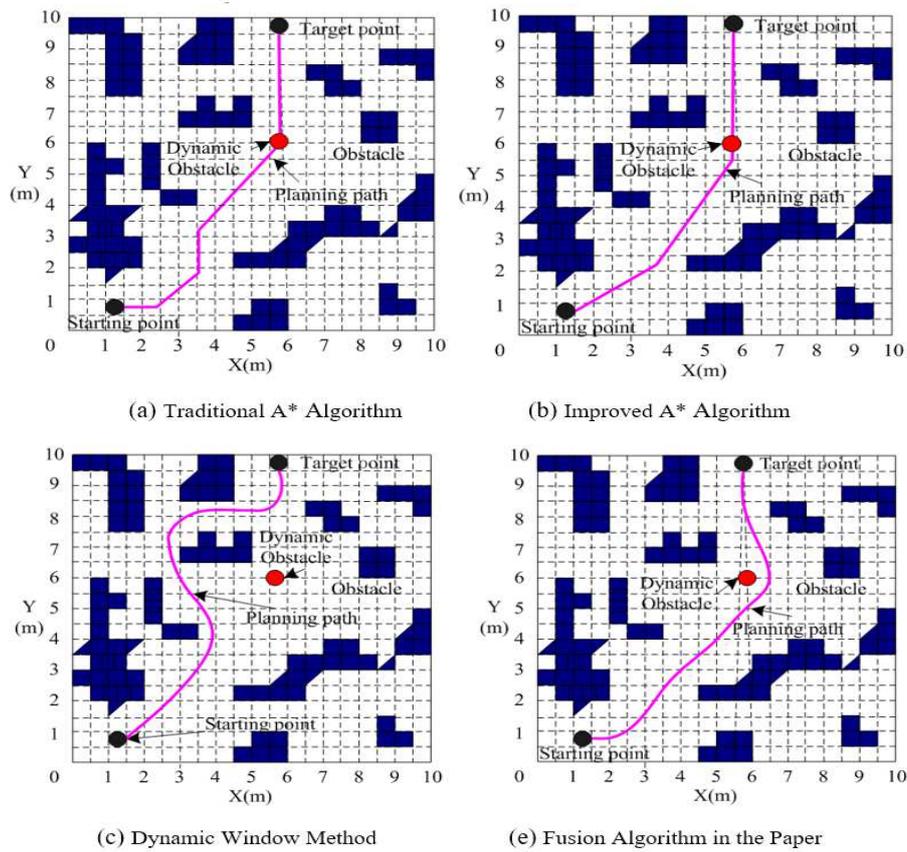


Figure 5: The simulation results of robot dynamic obstacle avoidance



Figure 6: The TurtleBot 2 mobile robot used in the experiment

the mobile robot global dynamic path plan is realized. Fig.8 shows a block diagram of the data processing for robot path planning.

In the experiment the environment local map in the movement is shown in Fig. 9. It is



Figure 7: Laboratory to build mobile robot navigation and obstacle avoidance experimental

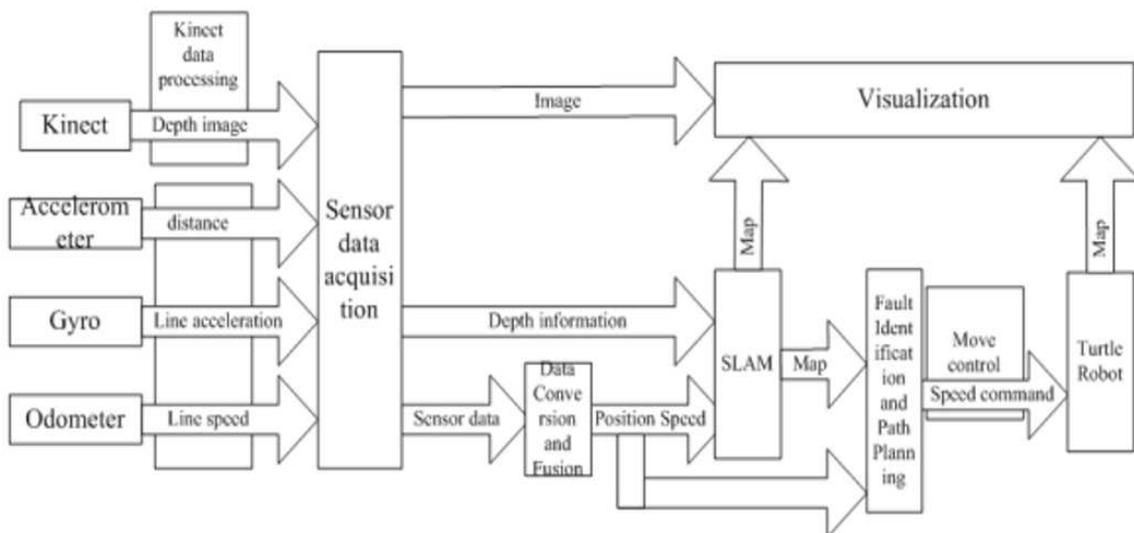


Figure 8: The block diagram of data processing for robot path planning

established by the Kinect visual sensor of the robot.

In the experiment, the initial scan was done using a speed of 20cm/s forward and rotational speed of 20cm/s, this is considered an average speed, as the robot is capable of 50cm/s forward and rotational speeds. The speed of the robot in the course of the road is appropriately slow, it ensures that the environmental data is fully analyzed and the environmental maps are established. As shown in Fig. 9, the surrounding dark area is Kinect has not yet obtained the environmental information, the blue-green area is the obstacle obtained by the expansion operation during the movement, the black part is known as an obstacle after detecting and recognizing. If a dynamic obstacle is encountered (such as a dynamic pedestrian or man-made obstacle) during the course of the process, an optimal line is re-planned and the robot smoothly bypasses the obstacle to the target point.

This experiment was carried out in unmanned system laboratory, about 10m * 10m labo-



Figure 9: The map of the robot in the indoor scene

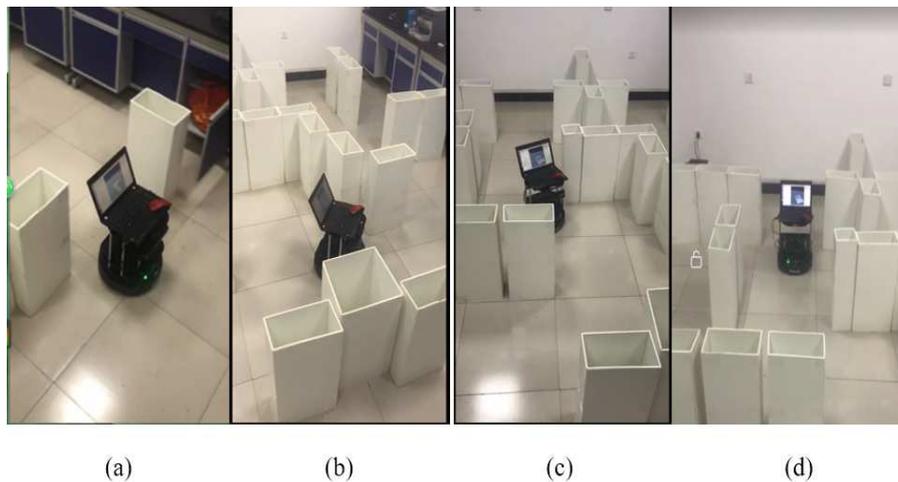


Figure 10: Indoor automatic dynamic obstacle avoidance path planning

ratory, a number of robot autonomous global navigation obstacle avoidance experiment is performed, the experimental results shown in Fig. 10 and Fig. 11. In Fig. 10, the sequence diagram of the global path planning experiment results is shown, it carried out by the mobile robot in the artificially set static obstacle experiment environment. Fig. (a) is the initial entry of the test



Figure 11: Dynamic obstacle

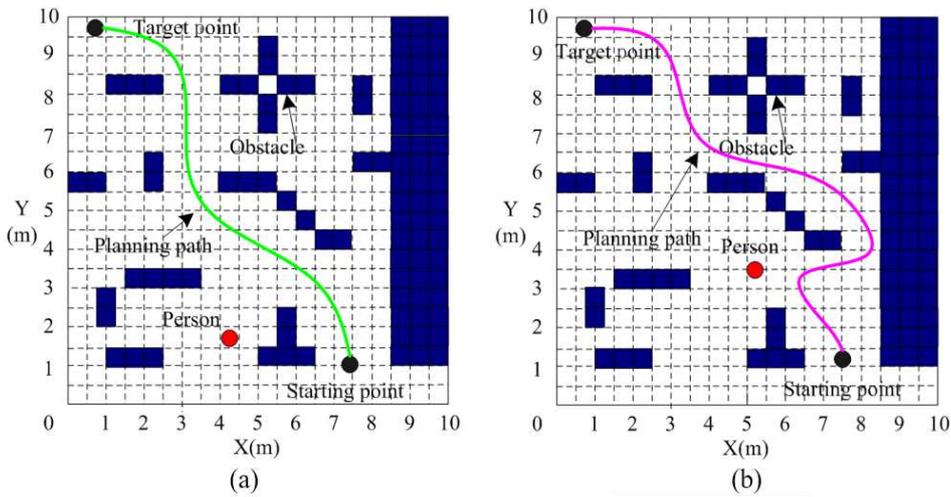


Figure 12: Dynamic obstacle avoidance experiment results

area, Fig. (b) and Fig. (c) are normal operations along a pre-defined path, Fig. (d) is near the end of the path.

Fig. 10 shows the normal operation of the mobile robot in the static obstacle of the experimental environment, the sudden emergence of human obstacle, as shown in Fig. 11 (a), mobile robot is running along the specified path, suddenly appeared in a red clothes (set the dynamic obstacle), the pre-planned path is blocked. Turble robot detects a fault, re-planning the path is re-planned; the final Fig. (b) Shows the path of the robot is re-planned, bypassing the obstacle, and returning to the original set end.

In Fig. 12, the blue is a static obstacle, the black circle is the mobile robot in the experiment, and the red is the dynamic obstacle (this experiment selects the person as a dynamic obstacle).In Fig.(a), the green solid line indicates the path planning of the robot when the dynamic obstacle person is initially immobile. In Fig. (b), the purple line indicates the re-planned path; it is obtained after the dynamic obstacle person randomly sets the obstacle point in the robot running. The results of the final experiment show that in the relatively complex environment, the robot can perform the global dynamic path planning more accurately, and it can dynamically bypass the obstacle and reach the target point in real time to realize the global path planning of real-time obstacle avoidance.

7 Conclusion

Aim at the problem of mobile robot path planning, a global dynamic path planning method based on advanced A* algorithm fusion dynamic window method is proposed. Based on the redundancy of the traditional A* algorithm, this paper designs a key selection strategy, the redundant points in the traditional A* algorithm can be removed, the performance of the global path planning is improved. Based on the dynamic window algorithm, an evaluation function considering the global optimal path is constructed; it can avoid the local optimal optimization of the dynamic window algorithm and ensure the global optimality of the path planning. Therefore, compared with the traditional A* algorithm, the path of this algorithm is more smooth, the curvature change is continuous, and the motion control parameters can be output. These are more conducive to the feedback control of the mobile robot and can be obstructed in real time. Compared with the dynamic window algorithm, the global optimization of the planned path can be ensured. Finally, the simulation and experimental results show that the improved A* algorithm and DWA combined method have good stability and real-time performance for the global real-time dynamic path planning of mobile robots, which provides a reference for the motion control of the mobile robot in the global dynamic environment.

Acknowledgment

This paper is grateful to the following projects: National Key Research and Development Program (2016YFE0111900), Shaanxi International Science and Technology Cooperation Project (2018KW-022 and 2017KW-009), Shaanxi Provincial Natural Science Fund (2017JM6041 and 2014 JM2-6093), also thanks to Xi'an Industrial University of International Science and Technology Cooperation Base Independent System and Intelligent Control International Joint Research Center of the unmanned system laboratory and autonomously intelligent control research and innovation team support for this subject.

Conflict of interest

The authors declare no conflict of interest.

Bibliography

- [1] Abd, M.E.; Ahmed, M.Y.; Amgad, M.B.; Yehia, Z.E. (2018). Fixed ground-target tracking control of satellites using a nonlinear model predictive control, *Mathematical Modelling of Engineering Problems*, 5(1), 11–20, 2018.
- [2] Bhattacharya, P.; Gavrilova, M.L. (2008). Roadmap-based path planning - using the Voronoi diagram for a clearance-based shortest path, *IEEE Robotics & Automation Magazine*, 15(2), 58–66, 2008.
- [3] Dai, Y.; Zhu, X.; Chen, L.S.; Liu, H.; Zhang, T.; Liu, S.J.A. (2015). New Multi-Body Dynamic Model of A Seafloor Miner And Its Trafficability Evaluation, *International Journal of Simulation Modelling*, 14(4), 732–743, 2015.
- [4] Dai, Y.; Pang, L.; Chen, L.; Zhu, X.; Zhang, T. (2016); A New Multi-Body Dynamic Model of a Deep Ocean Mining Vehicle-Pipeline-Ship System and Its Integrated Motion Simulation, *Journal of Mechanical Engineering*, 62(12): 757-763, 2016.

-
- [5] Dragic, M.; Sorak, M. (2016). Simulation for Improving the Performance of Small and Medium Sized Enterprises, *Journal of Simulation modeling*, 15(4), 597-610, 2016.
- [6] Eele, A.J.; Richard A. (2015). Path-planning with avoidance using nonlinear branch-and-bound optimization, *Journal of Guidance Control & Dynamics*, 32(2), 384–394, 2015.
- [7] Endres, F.; Hess, J.; Sturm, J.; Cremers, D.; Burgard, W. (2014). 3D Mapping with an RGB-D Camera, *IEEE Transactions on Robotics*, 30(1), 177–187, 2014.
- [8] Gao, Y.; Wu, X.; Liu, Y.; Li, J.M.; Liu J.H.(2017). A Rapid Recognition of Impassable Terrain for Mobile Robots with Low Cost Range Finder Based on Hypotheses Testing Theory, *International Journal of Computers Communications & Control*, 12(6), 813-823, 2017.
- [9] Gao, X.; Zhang, T. (2015). Robust RGB-D simultaneous localization and mapping using planar point features, *Robotics and Autonomous Systems*, 72, 1–14, 2015.
- [10] Genco, A.; Viggiano, A.; Magi, V. (2018). How to enhance the energy efficiency of HVAC systems, *Mathematical Modelling of Engineering Problems*, 5(3), 153-160, 2018.
- [11] Glasius, R.; Komoda, A.; Gielen, S.C.A.M. (1995). Neural network dynamics for path planning and obstacle avoidance, *Neural Networks*, 8(1), 125–133, 1995.
- [12] Herrera, C.D.; Kannala, J.; Heikkila. J. (2012). Joint depth and color camera calibration with distortion correction, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(10), 2058–2064, 2014.
- [13] Kaess, M. (2015). Simultaneous localization and mapping with infinite planes, *IEEE Int. Conf. on Robotics and Automation*, 4605–4611, 2015.
- [14] Lei, W.J.; Cheng, X.S.; Dai, N. (2014). Multi-model machining path planning based on improved genetic algorithm, *Journal of Mechanical Engineering*, 50(11), 153–161, 2014.
- [15] Liu, C.M.; Liu, L.; Liu, C.B. (2018). Analysis of wind resistance of high-rise building structures based on computational fluid dynamics simulation technology, *International Journal of Heat and Technology*, 36(1), 376–380, 2018.
- [16] Liu, J.H.; Yang, J.G.; Liu H.P. (2015). Robot global path planning based on ant colony optimization with artificial potential field, *Transactions of the Chinese Society for Agricultural Machinery*, 46(9), 18–27, 2015.
- [17] Neerendra, K.; Zoltan, V. (2016). Heuristic Approaches in Robot Navigation, *IEEE International Conference on Intelligent Engineering Systems*, 219–212, 2015.
- [18] Sara, J.; Jumel, F.; Simonin, O. (2017). Dynamic multi Agent patrolling Robotic application for service delivery to mobile people, *Revue d'Intelligence Artificielle*, 31(4), 379–4001, 2017.
- [19] Saric, T.; Simunovic, G.; Simunovic, K.; Svalina, I. (2016). Estimation of Machining Time for CNC Manufacturing Using Neural Computing, *Journal of Simulation Modeling*, 15(4), p. 663-675, 2016.
- [20] Song, X.R.; Chen, H. (2015). Stabilization Precision Control Methods of Photoelectric Aim-Stabilized System, *Optics Communication*, 9(351), 115–120, 2015.

- [21] Widyotriatmo, A.; Joelianto, E.; Prasdianto, A.; Bahtiar, H.; Nazaruddin, Y.Y. (2017). Implementation of Leader-Follower Formation Control of a Team of Nonholonomic Mobile Robots, *International Journal of Computers Communications & Control*, 12(6), 871-885, 2017.
- [22] Wang, C.; Mao, Y. S.; Du, K. J.; Hu, B. Q.; Song, L.F. (2016). Simulation on Local Obstacle Avoidance Algorithm for Unmanned Surface Vehicle, *Journal of Simulation modeling*, 15(3), 460-472, 2016.
- [23] Xiao, Q.K.; Liu, S.Q. (2017). Motion retrieval based on Dynamic Bayesian Network and Canonical Time Warping, *Soft Computing*, 21(1), 267–280, 2017.
- [24] Xiao, Q.K.; Song, R. (2017). Motion retrieval based on Motion Semantic Dictionary and HMM inference, *Soft Computing*, 21(1), 255–265, 2017.
- [25] Zhang, H.; Hu, Y.L. (2016). Path planning of mobile robot based on improved D* algorithm, *Industrial Control Computer*, 29(11), 73–77, 2016.
- [26] Zhu, D.Q.; Sun, B.; Li L. (2015). Algorithm for AUV's 3-Dpath planning and safe obstacle avoidance based on biological inspired model, *Control and Decision*, 30(5), 798–806, 2015.