2009 International Conference on Networks Security, Wireless Communications and Trusted Computing

A Hybrid Method for Dynamic Local Path Planning

Li Peng, Huang Xinhang, Wang Min
Department of Control Science and Engineering
Huazhong University of Science and Technology
Wuhan 430074, China
lipeng_hubei@126.com

Abstract— **A hybrid approach for efficiently planning smooth local paths for mobile robot in an unknown environment is presented. The single robot is treated as a multi-agent system, and the corresponding architecture with cooperative control is constructed. And then a new method of information fusion DSmT (Dezert-Smarandache Theory) is introduced to deal with the error laser readings. In order to make A* algorithm suitable for local path planning, safety guard district search method and optimizing approach for searched paths are proposed. Also, the parameters of internal Proportional-Integral-Derivative (PID) controller in the goto agent smoothes the path searched by evolutional A* algorithm. Finally, two kinds of experiments are carried out with Pioneer 2-DXe mobile robot: one uses the hybrid method proposed in this paper, the other uses artificial potential field (APF) which is the classical algorithm for local path planning. The experimental results reveal the validity and superiority of the hybrid method for dynamic local path planning. The approach presented in this paper provides an academic support for path planning in dynamic environment.**

Keywords- path planning, multi-agent, Dezert-Smarandache Theory, A* algorithm, mobile robot.

I. INTRODUCTION

Mobile robot path planning is one of the most important topics in robotics research. Point-to-point path planning of autonomous mobile robot, defined as finding a collision-free path linking a given start configuration to a goal configuration, has been extensively explored in last two decades. Many different methods achieving varying degrees of success in a variety of conditions/criteria of motion and environments have been developed.

Path planning for mobile robot is composed of two main parts: the global path planning and the local path planning. For global path planning, the entire environment is known for robot, the robot only needs to compute the path once at the beginning and then follow the planned path to the target point. Oppositely for local path planning, the robot only knows about the area which has been detected itself, it usually only casually decides the direction to move. There are many studies on robot path planning using various approaches, such as the grid-based A* algorithm [1], road maps [2], cell decomposition [3], and artificial potential field (APF) [4]. Some of the previous approaches use global methods to search the possible paths in the workspace, normally deal with static environments only, and are computationally expensive when the environment is complex. Some methods suffer from undesired local minima, the robots may be trapped in some cases such as with concave U-shaped barriers. But most studies mentioned above mainly focus on global path planning and there is few valid method for local path planning.

This paper presents an efficient hybrid approach for real time collision-free path planning and grid method is adopted to depict the environment map. In traditional path planning studies, the robot is always treated as a single unit. But here we make the robot a multi-agent system. In this system, each agent achieves its task and collaborates with other agents for the same purpose — to find a rational path. When the robot is commanded to reach an appointed target, in brief, the path planning agent calculates a temporary path with limited knowledge about the surroundings with the evolutional A* algorithm and the behavioral agent smoothes the planned path using its goto agent. The error readings are wiped off by filter based on DSmT [5]. And the path is revised once robot finds an obstacle block the path.

II. MULTI-AGENT ROBOT SYSTEM ARCHITECTURE

It is established that multiple cooperative control in a single robot involves cooperative control and multi-agent systems. Cooperative control has a very general meaning, any complex system developed with multi-agent architectures may be considered as a cooperative approach.

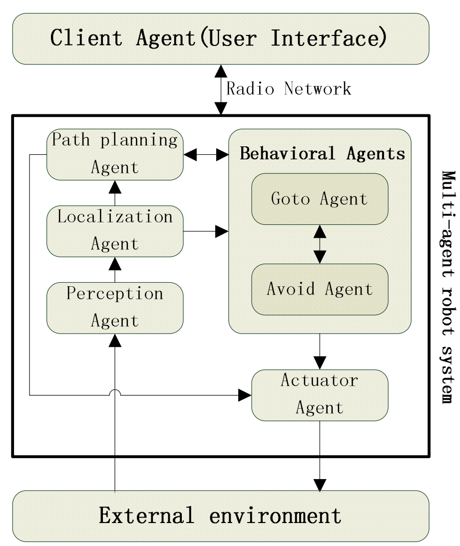


Figure 1. Multi-agent robot system.

The multi-agent robot system can be divided into five subsystems of agents: perception, behavior, path planning, localization and actuator (Fig.1). The behavioral agent

978-0-7695-3610-1/09 $25.00 © 2009 IEEE
DOI 10.1109/NSWCTC.2009.135

subsystem includes goto agent and avoid agent. In addition to all of the above, there is a client agent acting as user interface. Fig. 1 also depicts the flow of information among different agents.

The perception agent obtains information about the environment and about the internal conditions of the robot. Of course, it includes an error reading filter based on DSmT. It collects data from the sensors and after getting rid of error readings adapts them to provide the information requested by the other agents of the system.

The localization agent locates the robot on the global map, the path planning agent searches for obstacle free paths.

The behavioral agent subsystem carries out specific actions, such as avoiding obstacles, going to a point, etc. The information coming from the localization agent and path planning agent are used to react or respond to the changes produced in the robot itself or in the environment. The following agents have been defined: the goto agent, which is in charge of taking the robot from the initial to the final coordinates without considering obstacles; the avoid agent, which must go around the obstacles when they are found in the path of the robot.

The actuator agent is responsible for directly using the robot's various performance motion components, such as linear and angular velocity controllers, etc.

Once all the agents are running, the user can request a task through the client agent which sends the new robot goal to the path planning agent. Then the path planning agent decomposes the task into a series of turning point goal and sends the first target position to the goto agent. Based on this information and the actual position (obtained from the localization agent), the goto agent calculates the best linear and angular speeds to reach the target. On the other hand, based on the information provided by the localization agent and laser agent, the avoid agent calculates the linear and angular speeds needed to dodge the obstacle. At this point goto agent and avoid agent negotiate in order to decide who uses the motors. But usually the avoid agent does not need to work because the path planning has find a collision free path for robot except accidents. If the target position sent by the path planning agent is reached, another target position will be sent to the goto agent.

III. READING FILTER BASE ON DSMT

The DSmT of plausible and paradoxical reasoning proposed by the authors in recent years allows to formally combine any types of independent sources of information represented in term of belief functions [6,7]. And it is able to solve complex static or dynamic fusion problems, especially when conflicts between sources become large and when the refinement of the frame of the problem under consideration, denoted Θ, becomes inaccessible because of the vague, relative and imprecise nature of elements of Θ.

Here the Θ is defined as the status of each grid on the map constructed by the robot. Suppose there are two elements θ_1 and θ_2 in the frame of discernment Θ. θ_1 means the reading is wrong and θ_2 is defined as right. The hyper-power set is $D^{\Theta} = \{\phi, \theta_1\cap\theta_2, \theta_1, \theta_2, \theta_1\cup\theta_2\}$. Then define $m(\theta_1)$ as the general basic belief assignment function (gbbaf) for right status; $m(\theta_2)$ is the gbbaf for the wrong status; $m(\theta_1\cap\theta_2)$ is the gbbaf of conflict mass, it is generated during the fusion; and $m(\theta_1\cup\theta_2)$ is the gbbaf of unknown status (it mainly refers to those areas that still not be scanned at present).

There are two evidence sources for the filter. The first source is from the readings themselves. When the robot get a new group of laser readings, each reading is compared with its two neighbor readings (left reading and right reading) except the first and the last reading. For example, if a reading R is being checking, the left reading is defined as R_L and the right reading is R_R, if $R \gg R_L$ & $R \gg R_R$ or $R \ll R_R$ & $R \ll R_L$, these two situations (shown as Fig.2) means that the reading R probably is a wrong reading and it need to be fused with the second source mentioned later. Obviously other situations mean that R is a right reading and does not need further fusion.

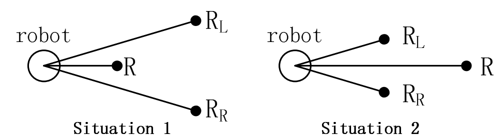


Figure 2. Two situations of error readings.

The belief assignments of the first source $m_1(\cdot)$: $D^{\Theta}\rightarrow[0,1]$ are constructed by authors as follows:

$$R_e = |R_L+R_R-2R|/2; \quad (1)$$

$$R_{max} = Max\{R, R_L, R_R\}; \quad (2)$$

$$m_1(\theta_1) = \exp[-5\times(R_e/R_{max})^2]; \quad (3)$$

$$m_1(\theta_2) = 1- m_1(\theta_1); \quad (4)$$

The second evidence source is from the map which has been built by the robot. If a laser reading is in one of the two above mentioned error reading situations, the $m_1(\cdot)$ calculated from the first source will be fused with the $m_2(\cdot)$ of second source. The robot checks the area around of the reading point which is marked as the potential error reading. The area is a rectangle with 5×5 grids, and the reading point is the center grid. The $m_2(\cdot)$ of the reading point is calculated according to the amount of occupied grid in this area. Suppose N is the amount of occupied grid in this area, the $m_2(\cdot)$ of the second source is computed as:

$$m_2(\theta_1) = \begin{cases} N/10, & N<10, \\ 1 \quad , & N\geq 10; \end{cases} \quad (5)$$

$$m_2(\theta_2) = 1- m_2(\theta_1); \quad (6)$$

The fusion between $m_1(\cdot)$ and $m_2(\cdot)$ follows the combining rules of Proportional Conflict Redistribution rule 2 (PCR2) [7] under the framework of DSmT. The PCR2 formula for $k\geqslant 2$ sources is:

$$\forall(X\neq\phi)\in D^{\Theta}, \ m_{PCR2}(X) = [\sum_{\substack{X_1,X_2,\ldots,X_s\in D^{\Theta}\\ X_1\cap X_2\cap\ldots\cap X_s=X}} \prod_{i=1}^{s} m_1(X_i)] + C(X)\frac{c_{12\ldots s}(X)}{e_{12\ldots s}}\bullet k_{12\ldots s} \quad (7)$$

$$k_{12\ldots s} = \sum_{\substack{X_1,\ldots X_s\in D^{\Theta}\\ X_1\cap\ldots\cap Xs=\phi}} \prod_{i=1}^{s} m_i(X_i) \quad (8)$$

In this formula, $C(X)$ equals 1 if X involved in the conflict, or it equals 0. $c_{12\ldots s}(X)$ is the non-zero sum of the column of X in the mass matrix, $k_{12\ldots s}$ is the total conflicting mass, and $e_{12\ldots s}$ is the sum of all non-zero column sums of all non-empty sets involved in the conflict.

After fusion, the final gbba of θ_1 can be calculated and if $m_1(\theta_1)\geq 0.8$ that means the laser reading which is being checked is a right reading instead of an error reading.

IV. PATH PLANNING METHOD

A. Path planning agent

1) Method of path planning agent: The robot plans the path through path planning agent. After updating the environmental information, the robot firstly makes use of evolutional A* algorithm to re-calculate the path if need. The *path from current location to the target point is* decomposed into a series of turning goal point. And then the real path between every two goal points is smoothed by internal PID controller.

The A* algorithm is usually used for static global path planning. There are few applications for dynamic local path planning in real time because of its large amount of calculations. In this paper, a simple but very effectual method is proposed to make the A* suitable for dynamic local path planning in real time. In most path planning studies, the robot is abstracted as a point without acreage. But actually the robot's radius must be considered in practical applications. The area near the planned path is the safety guard district. That means this area must be empty or the robot cannot pass. The safety guard district is as Fig.3.

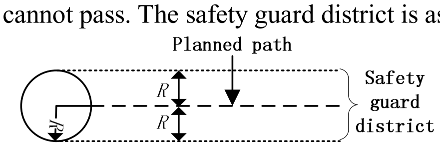
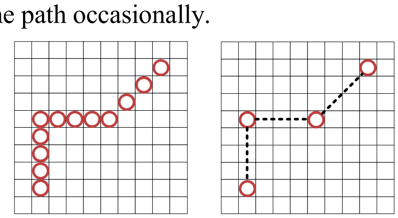


Figure 3. Safety guard district.

Once the robot gets a new group of laser readings, each reading is checked to make sure whether it is in the safety guard district. If a reading is in this area, it means an obstacle blocks the way and the path must be re-calculated. Oppositely, if there is none in the area, well then the path does not need to be re-calculated. This method reduces the computations of path planning. The robot only needs to calculate the path occasionally.



(a) Before optimizing (b) After optimizing

Figure 4. Sketch map of optimizing point-path.

2) Evolutional A algorithm:* The path searched by A* algorithm is a group of continuous goal points. If the grid is small, it will spend a large of memory to store the point-path; and if the goal points are placed too closely, the robot yet cannot follow the path well because of the limit of its turning radius. So here the point-path is optimizes. The goal points on the same line are deleted and then the robot only needs to store the turning goal points. This method markedly reduces the memory for storing point-path. The sketch map of optimizing point-path is shown in Fig.4.

B. Goto agent

It is known that the turning angle of the path calculated by A* algorithm in the grid map is 45° or 90°. So the path is not smooth and sometimes the robot cannot follow the trajectory because of these stark turnings. The goto agent can solve this problem. It is as shown in Fig.5, the robot's walking between two neighbor goal points is under the charge of goto agent.

The input of goto agent is the target goal point (x_i,y_i,θ_i) and the output is a group of control parameters (v,ω). v is the robot's velocity and ω is angular velocity for turning. The variable d is the distance from current robot's position to the target goal point.

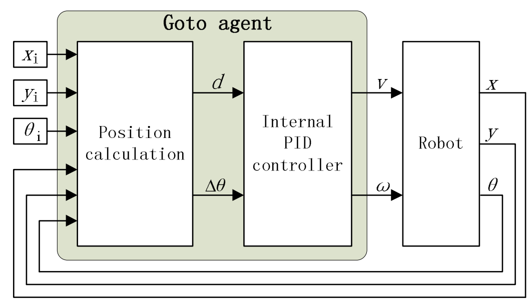


Figure 5. Architecture of goto agent.

V. EXPERIMENTS

A user interface platform for experiment is developed by authors. Pioneer 2-DXe mobile robot is used in experiments. Two kinds of experiments are carried out: one uses the hybrid method proposed in this paper, the other uses the classical algorithm artificial potential field (APF).

An experiment field (size: 4840×3100 mm) is created as Fig.6. The point of robot is treated as the coordinate origin of the global map. It is set to the pose of (0,0,0°). The third parameter is the deflection angle of robot. And the target goal position is placed near to the right top corner.

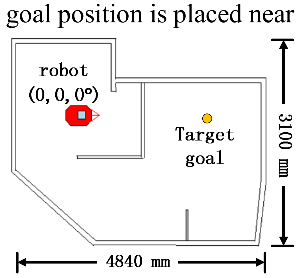
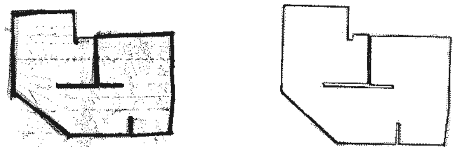


Figure 6. Experiment field.

Figure 9. APF experimental result

A. Hybrid method experiment

1) The effect of error reading filter: The effect of error reading filter is verified before the path planning experiment. The striking dissimilarity between mapping without filter and mapping with filter is revealed clearly in Fig.7. If there is no filter, the robot cannot find the way to the target goal because so many barrier-points block the way that there is not enough space to pass.



(a) Mapping without filter. (b) Mapping with filter.

Figure 7. Effect of error reading filter

2) Path planning experiment: In order to distinguish the planned path and the final real path, the real path is only displayed when the robot arrives at the target goal point. The experimental result is shown in Fig.8. The circles are the turning goal points calculated by A* algorithm.

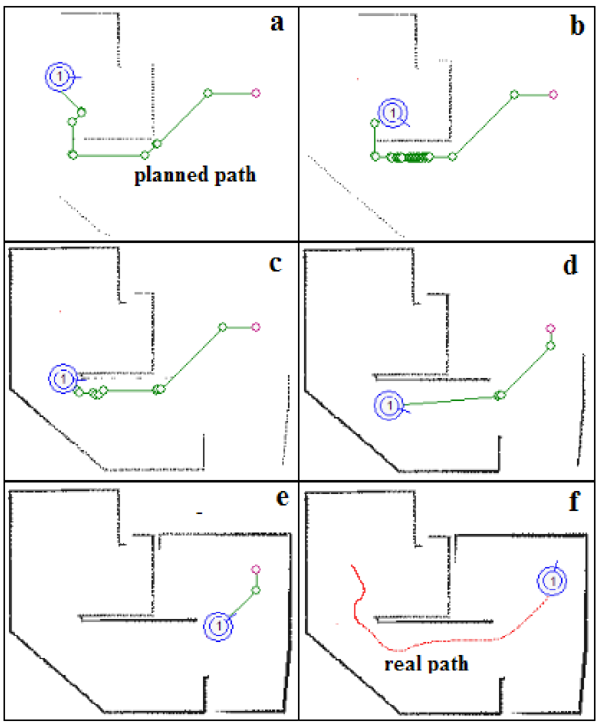


Figure 8. Hybrid method experimental result

B. *APF experiment*

The error reading filter is still used in this experiment for its important effect. The start position and the target goal position is the same as the hybrid method experiment. Of course in this experiment the robot is not a multi-agent system any more. For this experiment does not need the behavioral agent subsystem. And then the information flow becomes unilateral. There are no reciprocities between the parts in the robot. The APF experimental result is shown as Fig.9. The trajectory in the center is robot's real path.

C. *Analysis*

It is obviously that the hybrid method proposed in this paper is a very effectual method for dynamic local path planning. The experimental results show:

1) In hybrid method experiment, the robot only needs to store several turning goal points. The planned path is changed once the robot finds new barriers block the way; this is expressly shown in Fig.9(c~d).

2) The goto agent performs so perfectly that the real path of hybrid method experiment is very smooth instead of stark turnings which is the fault of A* algorithm.

3) Actually, the start position is a concave U-shaped trap. In hybrid method experiment, the robot can easily walk out this area and the path is rational. But in APF experiment, the robot is trapped in local minima; it repeatedly follows the same trajectory and cannot get out the repetition.

4) The effect of error reading filter based on DSmT is valid. The maps built with the laser readings are clean and accurate.

VI. CONCLUSIONS

This paper has proposed a hybrid approach for planning smooth paths satisfying dynamic local constraints for robot in an unknown environment. The single robot is divided into several synergic agents; among them the most important agents for path planning are path planning agent and behavioral agents since their cooperation influences directly the final result. This paper also presents a valid error reading filter based on DSmT. Safety guard district search method and optimizing approach for searched path are proposed. The results of experiments carried out with Pioneer 2-DXe mobile robot prove the validity and superiority of the hybrid method for dynamic local path planning. The application of the approach presented in this paper to path planning in completely dynamic unknown environment with moving objects around the robot will be investigated in the future.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China under grant No. 60675028.

REFERENCES

[1] Chia Hsun Chiang, Po Jui Chiang, Fei, J.C.-C., et al. A comparative study of implementing Fast Marching Method and A* SEARCH for mobile robot path planning in grid environment: Effect of map resolution, Proc. IEEE Workshop Adv. Rob. Soc. Impacts, ARSO. Hsinchu, Taiwan, Dec 9-11, 2007. pp. 1 – 6

[2] Jung Sungwon, Pramanik Sakti. An efficient path computation model for hierarchically structured topographical road maps, IEEE Trans Knowl Data Eng. 2002, 14 (5): 1029-1046.

[3] Rosell Jan, Iniguez Pedro. Path planning using harmonic functions and probabilistic cell decomposition, Proc IEEE Int Conf Rob Autom. Barcelona, Spain, Apr 18-22, 2005. pp. 1803-1808.

[4] Zhu Qidan, Yan Yongjie, Xing Zhuoyi. Robot path planning based on artificial potential field approach with simulated annealing, Proc. ISDA Sixth Int. Conf. Intelligent Syst. Design Applic., Jinan, China, Oct 16-18, 2006. pp. 622-627.

[5] Dezert J. Foundations for a new theory of plausible and paradoxical reasoning. Information and Security, 2002, 9: 13~57

[6] Dezert J, Smarandache F. Advances and Applications of DSmT for Information Fusion Vol.1. Rehoboth: American Research Press, 2004

[7] Dezert J, Smarandache F. Advances and Applications of DSmT for Information Fusion Vol.2. Rehoboth: American Research Press, 2006