Proceedings of the 7th
World Congress on Intelligent Control and Automation
June 25 - 27, 2008, Chongqing, China

Combined Coverage Path Planning for Autonomous Cleaning Robots in Unstructured Environments

Yu Liu, Xiaoyong Lin
Automation Institute
Zhejiang Sci-Tech University
Hangzhou, China, 310018
liuyu@zstu.edu.cn

Shiqiang Zhu
State Key Lab of Fluid Power Transmission and Control
Zhejiang University
Hangzhou, China, 310027
sqzhu@sfp.zju.edu.cn

Abstract **– Complete coverage path planning is a key problem for autonomous cleaning robots, which concerns not only the cleaning efficiency but also the adaptability to unstructured environments. But the diversity of environments and limited perception ability of the robot make the problem still unsolved. In this paper, a novel strategy of combined coverage path planning is proposed, which combines the random path planning and local complete coverage path planning. The random planning lets the robot run straight until an obstacle is encountered. After turning a random angle, the robot continues the straight run. This mode is easy to implement and can provide the robot with the flexibility to environments. And local complete coverage path planning works out a comb-like path depending on dead reckoning. The comb-like path can cover every part in a relative small area. All these functions are just based on general hardware: ultrasonic sensors, infrared sensors, incremental encoders, DC motors, vacuum, etc. Finally the experiment shows that this strategy can work efficiently and robustly in common family environments.**

Index Terms - Autonomous cleaning robots. Ccomplete coverage path planning. Self-localization. Environment modeling.

I. Introduction

Recently autonomous cleaning robots have emerged in average household. It helps people to do tedious everyday cleaning job. For example, iRobot's "Roomba" and Electrolux's "Trilobite" are successful pioneer. But they are still not perfect, left lots of work to do. Path planning (CCPP) was not done well. But it is a key problem for autonomous cleaning robots (ACR). It decides both the system's efficiency and practicability.

Theoretically the global path planning, which means planing the cleaning path on the entire floor area, can make out optimal path for the ACR and show great intelligence. This method needs the robot to analyze the shape, size and topological characteristics of the environment, and to decompose it into separate sections. Finally, the robot cleans these sections in an optimal sequence. So global self-localization and accurate environment modeling are essential for this mode. Now there are already some methods proposed, such as line sweep deposition [1], scanning path [2], motion template [3], cost function [4], neural network [5,6], etc. Reference [2] carried out an experiment on a real robot platform in an area of 10 square meters with a large obstacle inside. Scanning path was used to fill the cleaning area and the orientation error was corrected by referring to straight-line features on the ceiling. The experiment was still far from practical application. All these methods have the assumption that environment modeling and global self-localization are available. But both of the problems are still not well solved. References [7, 8] use laser rangefinder and CCD image sensor to map the environment and references [9, 10] locate the mobile robots by Markov approach under the priori that the environment map is known. These methods are based on expensive hardware and complicated algorithms, which are hard to apply to the ACR.

Now there are two practical methods in use: random path planning and fixed motion mode based on random path, for example, rounding on an involute path. Random path planning has the advantage of good flexibility to environments, but low efficiency limits its wide use. The second strategy has higher efficiency, but still cannot implement complete coverage path planning.

In this paper, a novel strategy of combined coverage path planning is proposed, which combines the random path planning and local CCPP. The local CCPP adopts a comb-like path in a small area, and periodically a random path is inserted. The random path is easy to implement and can provide the ACR with the flexibility to environments. The comb-like path has high coverage rate and is independent on global localization and accurate environment map; it utilizes the reliable self-localization ability of dead reckoning in a relative small area, and carries out coverage path like combing the hair. And, it requires only general hardware, for example, ultrasonic sensors, infrared sensors, incremental encoders, etc. The ultrasonic sensors are used to perceive the environment, and incremental encoders mounted on the wheels to locate itself in a small area.

Firstly, the ACR learns the size and complexity about the environment using the strategy of "groping and exploring"[10]. This information can be used to control the cleaning time and ration between planning cleaning time and random cleaning time. Then the system carries out the combined strategy. The random planning lets the robot run straight until any obstacle is encountered. After turning a random angle, the robot continues the straight run. And local CCPP works out parallel paths like a comb covering every part of small areas.

II. Combined Path Planning

We start with two definitions used in this paper:

Cleaning unit: The straight path while the robot runs free of obstacles is defined as the basic cleaning component, cleaning unit.

Cleaning coverage rate: In specific time, the rate between the floor area that the robot has cleaned and the whole floor area is defined as cleaning coverage rate:

$$\eta = s_{\text{covered}} / s_{\text{total}} \, . \tag{1}$$

2.1 Random path planning

Random path planning is shown as fig.1. Before the robot encounteres obstacles, it goes straight; Once the the ultrasonic sensors see the obstacle, the robot slows down until the robot almost touches the obstacle. Then the robot selects anothere direction to go at random. This mode works well even in complex environment and needs less hardware. But what about the efficiency?

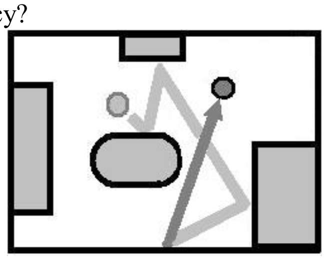


Fig.1 Random path planning

Let $f_{\mathrm{r}}(x_i)$ be single cleaning unit's coverage rate in random path planning, where x_i is cleaning unit's length. And let η_{r} be the whole random cleaning process's coverage rate.

So $$f_{\mathrm{r}}(x_i) = x_i \mathrm{W} / \mathrm{S}, \tag{2}$$

then $$\eta_{\mathrm{r}} = \sum_i \alpha_i f_r(x_i) \, . \tag{3}$$

where S is the room's total reachable area for the robot, and W is the robot's width. And α_i is the effective factor which presents the overlap between the cleaning path. With the increase of i, α_i decreases. When $i = 0$, $\alpha_i = 1$, because having no path to overlap then; When i reaches infinite, α_i is almost zero. Just for this reason, this path planning method has low coverage rate.

2.2 Local complete coverage path planning

Theoretically, Comb-like path shown as fig.2 has highest coverage rate. But this path planning method needs accurate localization data with no accumulative errors. But this kind of localization system is still hard to build because of expensive hardware and complex algorith.

And back-reckoning system with two increment encoders is easy to complement. The system is composed of two incremental encoders mounted on the robot wheels. The position ($x(t)$, $y(t)$) and heading $\theta(t)$ represent the state of the robot. The following nonlinear state equation describes the time-domain kinematic model of the ACR:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} v \cdot \cos\theta(t) \\ v \cdot \sin\theta(t) \\ \dfrac{\mathrm{R} \cdot (\omega_l - \omega_r)}{\mathrm{W}} \end{bmatrix} \tag{4}$$

Where, ω_l , ω_r is the angular velocities of the right and left wheel; R denotes the radius of the wheels; W represents the distance between the two wheels. So the linear velocity is

$$v = \mathrm{R}(\omega_l + \omega_r) / 2 \tag{5}$$

But there are many uncertainties in position and heading estimation, including unequal wheel diameters, misalignment of wheels, finite encoder resolution and sample rate, uneven floors and slippage of wheels. But within a short period, the pose estimation is accurate enough to carry out local coverage path planning and obstacle rounding action. Localization result in an office is shown in fig. 5.

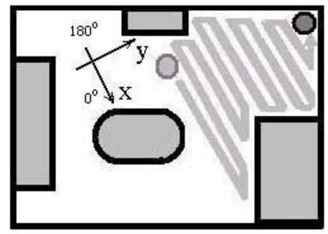


Fig.2 Local complete coverage path

In comb-like path planning shown as fig.2, localization system works only to decide cleaning direction and to control shift length. In order to leave least overlap and most coverage, shift length is set equal to the robot's width. So in this planning, even the localization data are not adequate, they can not affect the path planning badly.

When entering a narrow area, the complete coverage planning path is not effective. Due to the limit perception ability, the robot cannot realize the bad situation and alternate the work mode. A special algorithm is designed to deal this difficulty. When the length of cleaning unit is shorter than a predefined threshold, the robot ends the planning mode.

Let $f_{\mathrm{p}}(\mathrm{y}_i)$ be single cleaning unit' coverage rate in local complete coverage path planning, where y_i is the cleaning path length. Cleaning units seldom overlap each other in this planning. So

$$f_{\mathrm{p}}(y_i) = y_i \cdot \mathrm{W/S} \tag{6}$$

This path planning cannot completely finish the floor because of localization data error. It doesn't know where the wall is or where obstacles are, or where the start is or where the end is. So it should be combined with random path planning.

2.3 Combined path planning

Combined path planning builds together the random path planing and complete coverage path planning, shown as Fig.3. The two modes take the duty by turn. So one cleaning unit consists of one set of local complete coverage path and several random paths. So the coverage rate is:

$$f_{\mathrm{c}}(y_i + \alpha_j \beta_j x_j) = f_{\mathrm{p}}(y_i) + \sum_j \alpha_j \beta_j f_r(x_j) \tag{7}$$

where β_j is the effective factor. It presents the overlap between the random cleaning path and complete coverage path. The whole process's coverage rate is:

$$\eta_{\mathrm{c}} = \sum_i \gamma_i f_c(y_i + \alpha_j \beta_j x_j) \tag{8}$$

where γ_i is the effective factor. It presents the overlap between the combined paths.

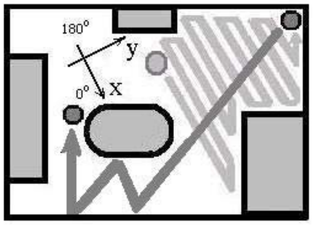


Fig.3 Combined path planning

Lemma 1: The coverage rate of combined path is higher than that of random path.

Proof: Given the speed of local complete coverage path equels that of random path, local complete coverage path is reguarded as same length random path. So the coverage rate is:

$$\eta_c = \sum_i \theta_i f_r(x_i) \tag{9}$$

For planned path x_i, the effective factor θ_i is bigger than effective factor α_i of random path, because planned path does not overlap that in the same set of planning path. This completes the proof.

The remainder of this paper is organized as follows: in Section 3, we introduce the system structure. Then in Section 4 environment leaning result is shown. In Section 5, implement of combined path planning is dicussed in details. Finally, Section 6 contains experimental results illustrating the combined path planning.

III. System Overview

The combined coverage path planning algorithm has been tested in our office environment on the real cleaning robot. It is round shape, which makes the robot easy to run about, shown as Fig. 4. Two DC motors can drive the ACR run straight, and turn at any radius. 11 pairs of ultrasonic sensors (a sender and a receiver for each pair) make up the perception system, 7 pairs being placed on the front side and the rest on the left and right sides. A bumper covers the whole front side, which gives out alarm when all the ultrasonic sensors fail. Two incremental encoders mounted on the robot wheels work as the self-localization system. And a battery pack is used which makes the robot move freely in the room. Finally, a small vacuum is fixed tightly in the ACR. During moving, the covered path is cleaned by this system.

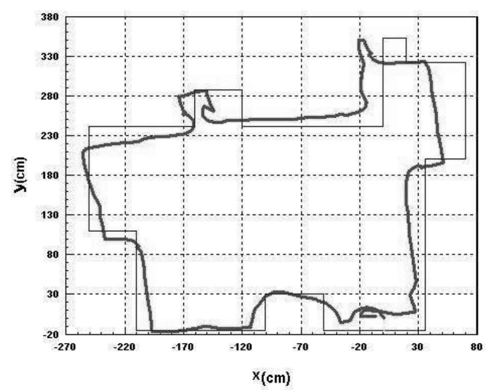
Fig.4 Experiment setup

IV. Environment Leaning

To decide how the ACR works and how much time it needs, the ACR should know the size and complexity of the environment. Because of short sight, the ACR adopts the strategy of “groping and exploring” to learn the environment like a blind man. Firstly the ACR runs out of the dock, and walk alone the wall. It keeps going until its side infrared receiver hears infrared signal from the dock. Then it stops beside the dock and approximately knows the size and shape of the environment.

Fig. 2 shows the shapshot of the experiment environment and the result of environment learning. The result explains that self-localization based on dead reckoning accumulates pose errors. But it keeps the main feature of the room, so the self-localization method can support local CCPP in a relative small area and also obstacle rounding action.

(a) Real environment



(b) Experiment result: solid line is modeling result, and dash line is the real measurements

Fig.5 Environment modeling

V. Path Planning

This combined coverage path planning includes two parts: random path planning and local CCPP. The two parts work alternately according to time. The total cleaning time t is defined as follows:

$$t = \sum_{i=0 \ldots n} (t_{ri} + t_{pi}) = f\left((x_{max} - x_{min})(y_{max} - y_{min}), \omega\right) . \tag{10}$$

Where, t_{ri} is the i^{th} random cleaning time; t_{pi} is the i^{th} planning cleaning time; n is the alternation times; x_{max}, x_{min}, y_{max}, and y_{min} are the maximal and minimal coordinates of the room; ω is the complexity measurement of the environment denoting the shape of the room, the amount of the furniture and small obstacles(such as legs of tables and stools), etc. For easy computation, the relation between cleaning time and the environment's features is treated as linear:

$$t = (x_{\max} - x_{\min})(y_{\max} - y_{\min})\mathrm{T}_0 + \omega \mathrm{T}_1 \tag{11}$$

Because the complexity ω is hard to measure, make$\omega = 0$ at the first cleaning time and (3) is simplified as:

$$t = (x_{\max} - x_{\min})(y_{\max} - y_{\min})\mathrm{T}_0 \, . \tag{12}$$

And the random cleaning time is

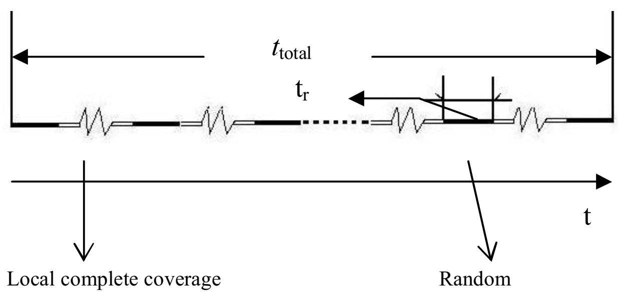
$$t_{\mathrm{ri}} = (x_{\max} - x_{\min})(y_{\max} - y_{\min})\mathrm{T}_\mathrm{r} \, . \tag{13}$$

T_0, T_1 and T_r are time factors obtained through experiments in a simple small room. The planned cleaning time t_{pi} is regulated dynamically according to the environment with an upper limit T_p. When the ACR enters a complex environment in which the planning path is non-effective, the planning cleaning is terminated and the random process is carried out.

$$t_{\mathrm{pi}} \le \mathrm{T}_\mathrm{p} \tag{14}$$

So the ration between the random cleaning time and planned cleaning time can be used to denote the complexity of the environment:

$$\omega = \sum_{i=0\ldots n} \mathrm{t}_{\mathrm{pi}} : \sum_{i=0\ldots n} t_{\mathrm{ri}} \, . \tag{15}$$



ning

A. Random Mode

The random mode lets the robot run straight until any obstacle is encountered. After turning a random angle, the robot continues the straight run. This kind of cleaning mode can provide the ACR with the flexibility to the environment, shown as Fig. 3.

M1 Algorithm for the random mode

Step 0 go straight until one of the two events occurs:

1) An obstacle is encountered. Then jump to **Step 1**;

2) The random cleaning mode times out. Jump to **Step 4**;

Step 1 Generate a turning angle $\Delta\theta$ at random.

Step 2 Turn around though the angle $\Delta\theta$;

Step 3 Jump to **Step 0**;

Step 4 The random mode ends**.**

The random mode is independent on self-localization and environment map, so it works well even in unstructured environments.

B. Complete Coverage Path Planning

The main cleaning task is taken by the local CCPP. This path is like a comb, which can clean the floor completely and leave little repeatedly cleaned area, shown as Fig. 4. The cleaning direction and shift distance (equal to the robot's width) is supported by the dead reckoning based on the two incremental encoders mounted on the wheels. In a single planning cleaning duty, the period is so short that the self-localization is relatively reliable with little accumulated error. So on general hardware, this strategy still can work.

The complete progress is as follows:

M2 Algorithm for the planning mode (cleaning alone the x coordinate axis):

Step 0 Find the cleaning direction (the x coordinate axis direction), and record the start point: $x_0 \leftarrow x$, $y_0 \leftarrow y$ (x and y are real-time estimated position);

Step 1 Run and clean straight, until one of the following events occurs:

1) An obstacle is encountered. Then go to **Step 2**;

2) The planning cleaning times out. Then go to **Step 8**;

Step 2 Check the distance:

$$d = \sqrt{(x - x_0)^2 + (y - y_0)^2} \, . \tag{16}$$

If d < M, go to **Step 8;**

Step 3 Walk close by the obstacle toward the y coordinate axis direction, until the movement distance Δs exceeds the vacuum suction mouth's width L, that is:

$$\Delta s = \int_l dy \rangle \mathrm{L} \, . \tag{17}$$

Step 4 Find the x coordinate axis counter direction, record the start point: $x_0 \leftarrow x$，$y_0 \leftarrow y$;

Step 5 run and clean straight, until one of the following events occurs:

1) Encounter an obstacle. Then go to **Step 6**;

2) The planed cleaning times out. Then go to **Step 8**;

Step 6 Check the cleaning distance d, using (7);

if $d <$ M ， go to **Step 8;**

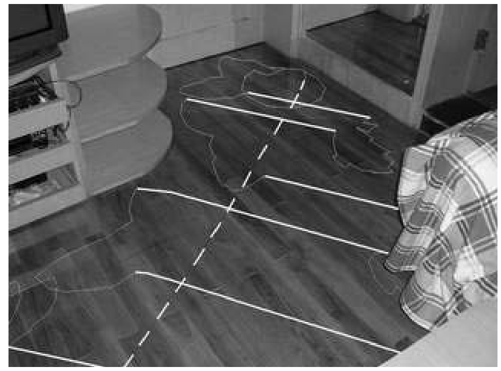
Step 7 Walk close by the obstacle toward the y coordinate axis direction, until the movement distance Δs exceeds the vacuum suction mouth's width L, and go to **Step 0;**

Step 8 The planning cleaning ends.

VI. EXPERIMENTS

An experiment was carried out in a common family room, which was unstructured with many small obstacles and tables and stools. A paintbrush with dark color was fixed on the back of the ACR. While going, the paintbrush left a clear trace showing the cleaning path. The experiment condition: the robot's speed was set at 0.2m/s, the total cleaning time was 10 minutes, and the total floor area was about 5.5*3.5 square meters.

After the experiment, the floor was shot by a digital camera. The pictures were processed, so that the planning and random paths were distinguished by different kind of lines, shown as Fig. 7. The results show the robot works well with no trap and the covering rate η is above 90%.



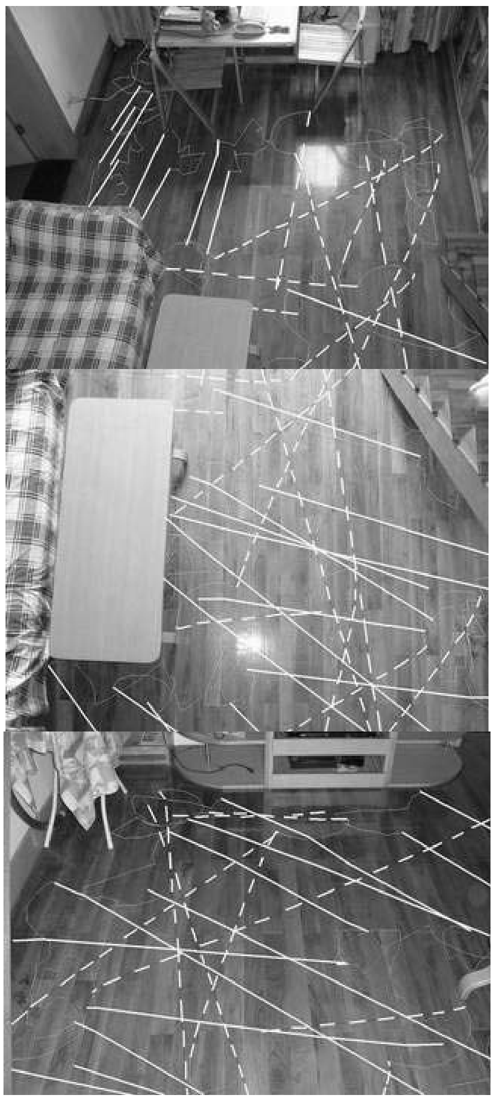


Fig 7 Combined covering path planning (Solid lines represent planning path; dash lines represent random path)

VII. CONCLUSION

In this paper, a novel strategy of combined coverage path planning is described, which works on general hardware and without any environment map and global self-localization. The random path planner makes out cleaning path, which is robust and flexible in unstructured environments; the local CCPP can make out comb-like path in a small area, which increases working efficiency and coverage rate remarkably. And the experiment results show the strategy works efficiently and robustly.

REFERENCES

[1] W.H. Huang, "Optical Line-sweep-based Decompositions for Coverage Algorithms", *Proc. IEEE Intl. Conf. on Robotics and Automation*, Seoul, Korea, May 21-26, 2001, pp. 27-32.

[2] Y.L. Fu and S.Y.T. Lang, "Fuzzy logic based mobile robot area filling with vision system for indoor environments", *Proc. IEEE Intl. Symp. on Computational Intelligence in Robotics and Automation,* Monterey, USA, 1999, 326-331.

[3] R. Neumann de Carvalho, H.A. Vidal, P. Vieira, and M.I. Ribeiro, "Complete coverage path planning and guidance for cleaning robots", *Proc. IEEE Intel. Symp. on Industrial Electronic*s, Guimaraes, Portugal, 1997, pp. 677-682.

[4] Pirzadeh, W. Snyder, "A unified solution to coverage and search in explored and unexplored terrains using indirect control", *Proc. IEEE Intl. Conf. on Robotics and Automation*, Cincinnati, OH, USA, 1990, pp. 2113 –2119.

[5] Luo, S.X. Yang, and X. Yuan, "Real-time area-covering operations with operations with obstacle avoidance for cleaning robots", *Proc. IEEE/RSJ Intl. Conf. on Intelligence Robots and Systems*, Lausanne, Switzerland, October 2002, pp. 2359-2364.

[6] C. Luo, S.X. Yang, D.A. Stacey, and J.C. Jofriet, "A solution to vicinity problem of obstacles in complete coverage path planning", *Proc. IEEE intel. Conf. on Robotics and Automation*, Washington, DC, USA, 2002, pp.612-617.

[7] L. Zhang,B.K. Ghosh, "Line segment based map building and localization using 2D laser rangefinder", *Proc. of IEEE Intl. Conf> on Robotics & Automation*, San Francisco.CA, 2000, pp.2538-2543.

[8] A.J. Davison, D.W. Murray, "Simultaneous localization and map-building using active vision", IEEE Transaction on Pattern Analysis and Machine Intelligence, vol.24, no.7, pp.865-880, 2002.

[9] Dieter Fox, Wolfram Burgard, Sebastian Thrun. Active Markov Localization for Mobile Robots[J]. Robotics and Autonomous Systems,1998,25:195-207.

[10] M. Mehrandezh, and K.K. Gupta, "Simultaneous path planning and freee space exploration with skin sensor", *Proceedings of IEEE International Conference on Robotics and Automation*, Washington, DC, USA, May 2002, pp. 3838-3843.

[11] Q.X. Wu, B. David, "A study on Markov localization for mobile robots", *Acta Automatica Sinica*，vol. 29, no. 1, pp.154-160, 2003 (In Chinese, with English abstract).

[12] H.B. Wu, S.Q. Zhu, X. Ma, "Obstacle avoidance and path planning for autonomous cleaning robot in unstructured environments", *Robot*, vol. 22, No.7, 2000(In Chinese, with English abstract).
[13] X. Ma, S.Q. Zhu, H.B. Wu, "Design of intelligent dust-collecting robot", *Application of Electronic Technique*, vol.26, No.8, pp.6-8, 2000(In Chinese).
[14] X. Ma, S.Q. Zhu, J. Fu, "Study on intelligent dust-collecting robot", *Proc. of the Fifth Intel. Conf. on Fluid Power Transmission and Control*, Hangzhou, China, 2001, pp.392-396.
[15] Y. Liu, S.Q. Zhu, B. Jin, S.S. Feng, H.F. Gong, "Sensory navigation of autonomous cleaning robots", *5th world congress on intelligent control and automation*, in press.