Virtual local target method for avoiding local minimum in potential field based robot navigation*

ZOU Xi-yong(邹细勇)†, ZHU Jing(诸 静)

(*College of Electrical Engineering, National Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China*)

†E-mail: zouxiyong@163.net

Received June 3, 2002; revision accepted Aug. 10, 2002

Abstract: A novel robot navigation algorithm with global path generation capability is presented. Local minimum is a most intractable but is an encountered frequently problem in potential field based robot navigation. Through appointing appropriately some virtual local targets on the journey, it can be solved effectively. The key concept employed in this algorithm are the rules that govern when and how to appoint these virtual local targets. When the robot finds itself in danger of local minimum, a virtual local target is appointed to replace the global goal temporarily according to the rules. After the virtual target is reached, the robot continues on its journey by heading towards the global goal. The algorithm prevents the robot from running into local minima anymore. Simulation results showed that it is very effective in complex obstacle environments.

Key words: Local minimum, Virtual local target, Rules, Potential field based robot navigation
Document code: A **CLC number:** TP13

INTRODUCTION

Autonomous navigation is one of the most important topics in the mobile robot area and can be categorized into two parts: reactive navigation (Haddad *et al.*, 1998) and path planning. The first one is local path planning based and the second one plans a path in the global workspace. Local path planning is on-line obstacles avoidance strategy using the environmental information from its perceptual system and does not need a prior model of the workspace, whereas global path planning generates the overall path with much prior information on the workspace. In the global path planning area, A^* algorithm (Podsedkowski *et al.*, 1999) and potential-field method (Wu *et al.*, 1995; Wang *et al.*, 2000) are the most popular approaches used, wherein the environment is usually divided into a number of cells or grids. If the number of the grids is small, the map will lose much information and it may result in a poor path. If the number is too big, the computational cost will increase fast and real-time navigation is less efficient. Considering these drawbacks, we presented a new vector field based path planning algorithm, which in fact is an improved potential field method. In this algorithm, an analytical rather than numerical solution was obtained for the field model, thus the algorithm is computationally simple and can navigate the robot in real time.

The potential field method is performed in an iterative fashion; a robot moves in the direction of the resultant of the attraction force pulling the robot toward the goal, and the repulsive force pushing the robot away from the obstacles. As expected, the robot stops moving after reaching the goal position. But unfortunately, it always suffers from local minima where if trapped, the robot will oscillate in the potential valley and no finished path can be generated. This limits the applicability of the potential-field approach. Looking for a local-minimum-free solution has become a central concern in this approach.

Various efforts have been made to overcoming the local minimum problem. The two main methods include: (1) establishing new potential functions with a few or even no local minima; (2) use of certain techniques to escape from local minima. Previous papers in the first category

* Project (No. 69904009) supported by the National Natural Science Foundation of China

introduced some special potential functions to reduce the number of local minima. Rimon *et al*. (1992) introduced a potential function, which dealt with a sphere world. And a superquadratic potential function was presented by Volpe *et al*. (1990). Ge *et al*. (2000) proposed a new potential field function for the problem of goals not reachable with obstacles nearby, which ensures the goal position is the global minimum of the total potential. Harmonic functions do not have local minima were used as potential functions in robot navigation (Connolly, 1992), but are computationally expensive.

Previous works in the second category include random walk, wall following, and other heuristic methods. The random walk method requires walking in many directions before escaping from a local minimum. Janabi-Sharifi *et al*. (1993) introduced Simulated Annealing technique to search in random motion directions to escape from local minima when trapped. Wall following method steers the robot to follow the current obstacle contour when trapped in a local minimum (Borenstein *et al*., 1989; Yun *et al*., 1997). But the algorithm may generate false local minima and result in indefinite loops even in very simple environment.

Also some other heuristic methods (Chang, 1996; Liu *et al*., 2000; Singh *et al*., 1996) were developed to escape from the local minima. Liu integrated virtual obstacle concept with a potential field method. When the robot is trapped in undesired local minimum, a fuzzy tracking controller, which adds a fictitious force around the obstacle, is employed to escape the trapping.

Nevertheless, the solutions are usually limited to given or simple obstacle, and are difficult to be employed in complex environments. In this paper, we try to develop an algorithm that can navigate in indefinite and complex obstacle environments where obstacles are assumed to be rectangular.

Our extensive experimentation showed that the local minima usually occur at positions close to the obstacles. If the obstacles are too close, the possibility that they form a local minimum will increase rapidly. We introduce the concept of virtual local target, which replaces the global target as the current goal. In the navigation, when a new obstacle is detected to come into the check-range (an area around the robot), a possible local minimum exists. Then a virtual local target will be appointed according to the rules to replace the global goal. After this local target is reached, the global goal resumes and this process will be repeated until the robot reaches the global goal. Eight models were summarized for possible local minimum checking. And rules for appointing the virtual local target have been set up, which ensures that the new generated path is free from local minima and is as short as possible.

LOCAL MINIMUM PROBLEM

1. Local minimum example

A simulation result using the standard potential field method in an environment with two obstacles is shown in Fig. 1, which can illustrate the local minimum problem. In this figure, S is the current position and G is the global goal position, the two obstacles are zero distance to each other; the curve labeled with number are contours of the magnitude of the force in the potential field. We can find that the robot is trapped in a local minimum, which exists on the zero-force position. The path oscillates in this potential valley around the local minimum. A finished path cannot be generated, i.e., the standard potential field method fails in such an environment.

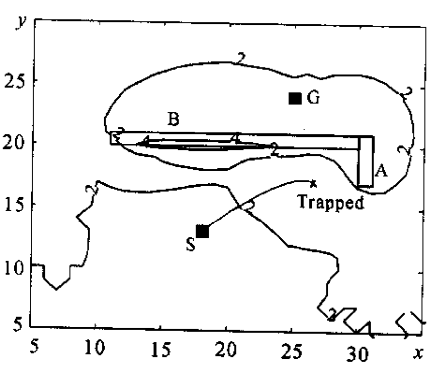

Fig.1 A local minimum example

■ S, the current position; ■ G, the global goal position; A, B: obstacles

2. Obstacle environment with local minimum

Extensive experimentations were made in various environments. As shown in Fig.2, eight models, which are local minimum susceptible are summarized for successional operation. There are two rectangular obstacles (H and V) in every environment, the distance between them is zero, and they together make up of another right-angle-shaped obstacle. The line between the current position S and the global goal G (SG) intersects an obstacle H or V, which is close to S. When two obstacles are too close to each other, there will be no path through them. Thus if the distance between two obstacles is smaller than twice the diameter of the robot, it is regarded to be zero and then the obstacles are matched with one of those models in the figure.

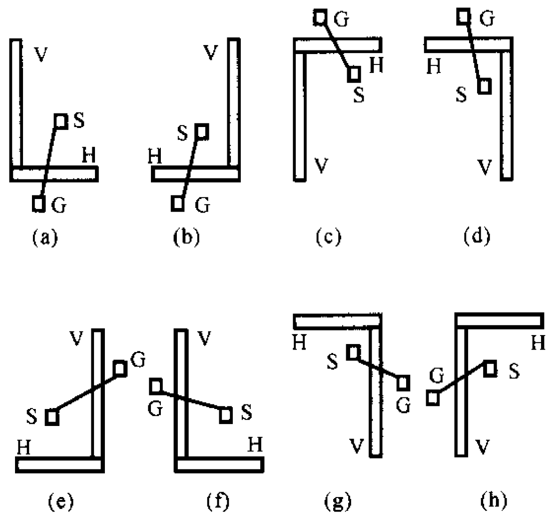

Fig.2 Eight right-angle-shaped obstacle models for local minimum detection

(a)(b)(c)(d): line SG intersect with H
(e)(f)(g)(h): line SG interesct with V

If the robot runs into such environments as shown in the figure, a local minimum is very likely to occur. A novel algorithm integrating the concept of virtual local target and the potential field navigation method is presented to solve this intractable problem. The concept will be illustrated in the next section.

Notice that usually there are three types of local-minimum-generating obstacles: bench, corner (right-angle-shaped), and dead-end (Podsedkowski *et al.*, 1999). The first one is easy to be solved and the third one can be decomposed into two corner ones.

RULES BASED VIRTUAL LOCAL TARGET METHOD

In this section, we explain the concept of virtual local target and the rules governing when and how to appoint the targets.

For checking the position whether it is in danger of local minimum, the distance of the robot to the obstacle and the distance between the obstacles need to be calculated.

1. Distance between obstacles

The distance between two obstacles is calculated as follows:

First find the obstacle with bigger coordinate y (y_{A+} in Fig. 3), label it A, the other B; calculate the distance in the Y-axis direction:

$d_1 = \max(0, y_{A-} - y_{B+})$

Also, calculate the distance in the X-axis direction:

$d_2 = \max(0, x_{A-} - x_{B+})$

$d = \min(d_1, d_2)$

d is defined to be the distance between these two obstacles.

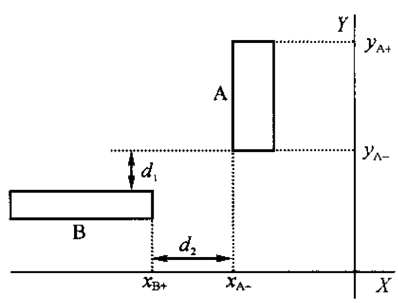

Fig.3 Distance between obstacles

2. Distance of the robot to obstacle

As have been mentioned, the local minimum often occurs at the location near the obstacles. When the robot moves to the location, which is less than a threshold D_s away from the obstacle, checking must be done to see if it is in danger of local minimum according to the models stated in Section 2. To make the computation simple, we define the distance of the robot to an obstacle as the smaller one of the distances along the X-axis and Y-axis:

a. If $x_{O-} < x_R < x_{O+}$, $d_x = 0$; else $d_x = \min(|x_R - x_{O+}|, |x_R - x_{O-}|)$

b. If $y_{O-} < y_R < y_{O+}$, $d_x = 0$; else $d_y = \min(|y_R - y_{O+}|, |y_R - y_{O-}|)$

c. $d_{RO} = \max(d_x, d_y)$

3. Concept of virtual local target

Fig. 4 shows an environment similar to the one in Fig. 1. There exists a local minimum near the intersection of the two obstacles. In this figure, D_R is the diameter of the robot. VT is the virtual local target we introduce, and is on the left end of obstacle ABCD, which intersects line SG. And the other obstacle is on the direction opposite to the virtual local target. Point VT is D_R to border CD in the direction of the Y-axis and $3D_R$ to border AD in the direction of the X-axis.

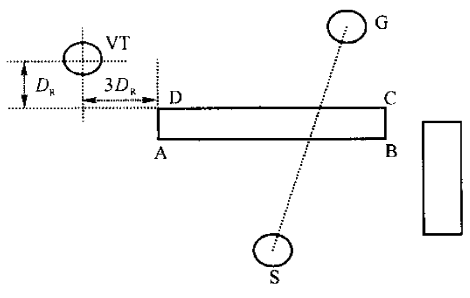

Fig. 4 Vivtual local target concept

When the robot finds itself in such an environment, the current goal G will be replaced by the new virtual local target VT. A path from the current position to the new temporary goal will be generated, using the potential field navigation algorithm. When it reaches this goal, the robot will continue on its journey by heading towards the global goal G.

4. Rules for appointing the virtual local target

(1) Local minimum detection

When the distance of the robot to a new detected obstacle is smaller than a threshold Ds, assumed here to be five times the diameter of the robot (which is correlated to the field model parameters; if there is no wide gap between the coefficients of the attractive and repulsive fields, this selection is appropriate), a possible local minimum exists, then the nearby obstacles environment should be checked. New means that the obstacle has not been checked and that it must be in the direction of the motion. Calculate the distance of the obstacle to "past-ones", if the distance is smaller than twice the diameter of the robot, match the composite obstacle, which is composed by the new detected obstacle and the past-one, with one right-angle-shaped obstacle model in Fig. 2.

(2) Look-back strategy

Here we illustrate the concept of "past-ones" in Part (1). For example, if the robot moves southeast, "past-ones" are the obstacles in the west and north directions. This concept we call look-back strategy, which ensures that the robot does not navigate back to past journeys. When more than one right-angle-shaped obstacle are formed, the one near the current position should be selected.

(3) How to appoint the position of virtual local target

Below are the rules for appointing the virtual local target in the eight models in Fig. 2:

1) If line SG intersects obstacle H, and if obstacle V is on the right of H (model Fig. 2b and Fig. 2d), appoint the virtual local target VT on the left of H (as shown in Fig. 4), else if V is on the left of H (model Fig. 2a and Fig. 2c), locate the target on the right of H.

2) If line SG intersects obstacle V, and if obstacle H is on top of V (model Fig. 2g and Fig. 2h), appoint the virtual local target below V, else if H is below V (model Fig. 2e and Fig. 2f), locate the target on the top of V.

3) If line SG intersects both H and V, no virtual local target need be appointed.

4) Avoid-past rule: If the virtual local target appointed according to the above rules is in the opposite direction of motion (on the end of obstacle H/V), it should be changed to the end of the other obstacle (on the end of obstacle V/H).

5. Nesting of virtual local target

One of the dead-end obstacles is like the one shown in Fig. 5, which has only a narrow channel at one end. It cannot be dealt with in traditional way (Podsedkowski *et al.*, 1999). Virtual local target concept is adopted to solve this problem, but the operation should be carried out in a nesting way.

In this environment, after the robot starts from S, when it finds obstacle 1 coming into the local minimum detection area, virtual local target VT1 is appointed according to the rules stated above. But on the journey to this target, the robot finds another new obstacle 2. VT1 obviously cannot be reached currently. A nesting strategy is adopted to avoid local minima in such a scenario.

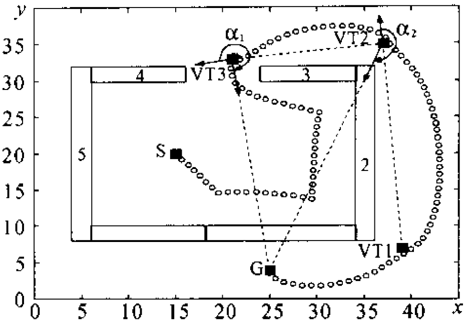

Fig.5 Avoiding local minima in a dead-end type obstacle

A stack is defined to preserve the un-reached virtual local minima. In this example, VT1 is pushed into the stack for obstacle 2, and then VT2 is pushed into the stack for obstacle 3. But VT3 can be reached directly and need not be pushed into the stack.

Define two directions at the currently reached virtual target: one is from the un-reached virtual target on the stack top to this reached target, and the other is from this reached target to the global target. The corner from the first direction turning to the second one according to the rule stated below is defined as the target corner.

Turning rule: If the nearest obstacle is on the left of robot, turning direction is counter-clockwise. And if it is on the right of the robot, turning direction is clockwise.

When VT3 is reached, such target corner is defined α_1 which is from line (VT2→VT3) to line (VT3→G) as shown in Fig.5.

Stack operation rule: If a virtual local target cannot be reached for the moment, it is pushed into the target stack. And if there is any target in the stack, when a virtual local target is reached, pop the target on the stack top and check the target corner α at current position. If α is greater than π, the popped virtual local target should be resumed, otherwise the global goal should be resumed. In this example, VT2 is resumed at location VT3, and VT1 is resumed at location VT2.

This nesting method navigates the robot in such a dead-end obstacle environment, free from any local minimum. The path generated using the method is shown in Fig. 5.

6. Local minimum avoidance for bench obstacle

When the encountered obstacle is bench type one (as mentioned in Podsedkowski *et al.*, 1999), whose distance to the robot is smaller than the threshold, appoint one virtual local target on one of the two ends of the obstacle similar to operating in Fig.4.

APPLICATION

In order to verify the algorithm's ability to tackle the local minimum problem in potential field robot navigation, many simulations have been made in environments with local minima; here are two of them. The navigation simulations are carried out using Matlab 5.3 software package on PC with Pentium III 550 CPU and 128M RAM.

Fig.6 and Fig.7 are the robot paths generated using different algorithms in the same envir-

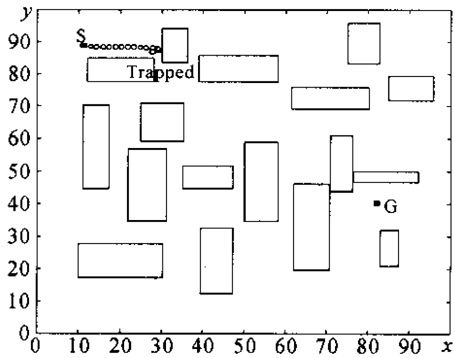

Fig.6 Path generated using standard potential field algorithm

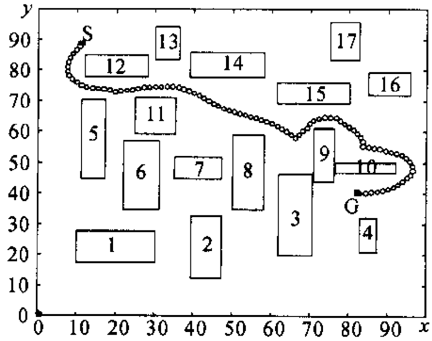

Fig.7 Path generated using virtual local target concept (1 – 17, obstacles)

onment with the same field model parameters. In

Fig.6, standard potential field navigation algorithm is used. It can be seen that the robot gets rapidly trapped in a local minimum. In Fig.7, the navigation algorithm presented in this paper is applied. Some virtual local targets are appointed to generate the finished path. When the robot starts at position S, it finds itself in a right-angle-shaped obstacle environment, which is formed by obstacle 12 and obstacle 13. According to the virtual local target concept and the rules, a temporary goal on the left of obstacle 12 is appointed to replace the global goal. When it arrives at this target, the global goal resumes and the robot continues on its journey by heading towards the global goal. This process is repeated until the robot reaches goal G.

Robot navigation in a more complex environment with maze-like obstacles using this algorithm is shown in Fig. 8. It can be seen that the path is safe and short.

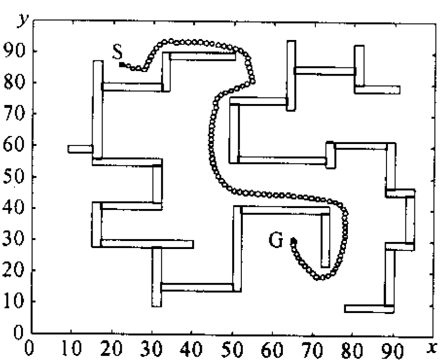

Fig.8 Path generated in a maze-like environment

CONCLUSIONS

In this work, a frequently encountered problem, local minimum in potential field robot navigation method was studied thoroughly. Eight environment models, which are local-minimum suspectable are summarized. The virtual local target concept is proposed to solve this intractable problem and rules are presented to direct appointing this target position appropriately. Simulation results proved the validity of this algorithm. But it can be seen in the results that there is a sharp turning at the virtual local target position. When steering the robot, these turnings can be completed in several steps.

Also the presented algorithm is computationally simple and appropriate for real-time navigation.

References

Borenstein, J. and Koren, Y., 1989. Real-time obstacle avoidance for fast mobile robots. *IEEE Trans. On Systems, Man and Cybernetics*, **19**(5):1179 – 1187.

Chang, H., 1996. A new technique to handle local minimum for imperfect potential field based motion planning. *In*: Proceedings of IEEE International Conference on Robotics and Automation, Minneapolis, USA, **1**: 108 – 112.

Connolly, C. I., 1992. Applications of harmonic functions to robotics. *In*: Proceedings of the IEEE International Symposium on Intelligent Control, p.498 – 502.

Ge, S. S. and Cui, Y. J., 2000. New potential functions for mobile robot path planning. *IEEE Trans. on Robotics And Automation*, **16**(5):615 – 620.

Haddad, H., Khatib, M., Lacroix S. and Chatla, R., 1998. Reactive navigation in outdoor environments using potential fields. *In*: Proceedings of IEEE International Conference on Robotics and Automation, Lewen, Belgium, **2**:1232 – 1237.

Janabi-Sharifi, F. and Vinke, D., 1993. Integration of the artificial potential field approach with simulated annealing for robot path planning, *In*: Proceedings of the IEEE International Symposium on Intelligent Control, Chicago, USA, p.536 – 541.

Liu, C. Q., Marcelo, H. Ang. Jr, Hariharan, K. and Lim, S. Y., 2000. Virtual obstacle concept for local-minimum-recovery in potential-field based navigation. *In*: Proceedings of IEEE International Conference on Robotics and Automation, San Francisco, USA, **2**:983 – 988.

Podsedkowski, L., Nowakowski, J. and Idzikowski, M., 1999. Modified A* algorithm suitable for online car-like mobile robot control. *In*: Proceedings of the First Workshop on Robot Motion and Control, Kiekrz, Poland, p. 235 – 240.

Rimon, E. and Koditschek, D. E., 1992. Exact robot navigation using artificial potential functions. *IEEE Trans. on Robotics And Automation*, **8**(5):501 – 518.

Singh, L., Stephanou, H. and Wen, J., 1996. Real-time robot motion control with circulatory fields. *In*: Proceedings of IEEE International Conference on Robotics and Automation. Minneapolis, USA, **3**: 2737 – 2742.

Volpe, R. and Khosla, P., 1990. Manipulator control with superquadric artificial potential functions: theory and experiments. *IEEE Trans. On Systems, Man and Cybernetics*, **20**(6):1423 – 1436.

Wang, Y. F. and Chirikjian, G. S., 2000. A new potential field method for robot path planning. *In*: Proceedings of IEEE International Conference on Robotics and Automation, San Francisco, USA, **2**:977 – 982.

Wu, K. H., Chen, C. H. and Lee, J. D., 1995. A fuzzy potential approach with the cache genetic learning algorithm for robot path planning. *In*: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, vancouver, CA, **1**:478 – 482.

Yun, X. P. and Tan, K. C., 1997. A wall-following method for escaping local minima in potential field based motion planning. *In*: Proceedings of International Conference on Advanced Robotics, Monterey, USA, p.421 – 426.