A comparative study of A-star algorithms for search and rescue in perfect maze

Article · April 2011

DOI: 10.1109/ICEICE.2011.5777723

2 authors, including:

Daoxiong Gong
Beijing University of Technology
45 PUBLICATIONS **232** CITATIONS

SEE PROFILE

A Comparative Study of A-star Algorithms for Search and rescue in Perfect Maze

Xiang Liu
School of Electronic Information and Control Engineering
Beijing university of Technology
Beijing 100124, China
newshine.soon@yahoo.com.cn

Daoxiong Gong
School of Electronic Information and Control Engineering
Beijing university of Technology
Beijing 100124,China
gongdx@bjut.edu.cn

Abstract—**Robots can be widely used to fulfill the task of search and rescue trapped persons in some dangerous situations, which can be abstracted as a maze. Three A-star algorithms are studied in this paper to compare the maze searching capacity and efficiency of their different heuristic functions, and the depth-first search algorithm, which has no heuristic information, is also adopted as a benchmark to judge the usefulness of the 3 heuristic functions of A-star algorithms. Experiments validated the usefulness of heuristic function with the results that the A-star algorithms outperform the depth-first search algorithm in most cases, and the A-star algorithm with the Euclidean distance from the father point of current point to the target point included in the heuristic function shows the best performance.**

Keywords- Maze search, the depth-first search algorithm, A-star algorithm

I. INTRODUCTION

Robots can be widely used to fulfill the task of search and rescue persons trapped in some dangerous situations, e.g., the conflagrant building, the exploded house, the collapsed mine, the ruins of earthquake, and so on. Searching efficiency or search time is the first concern in the applications of search and rescue, because it is urgent to detect the position of the wounded or trapped persons in the ruins and save them as fast as possible. These complex situations of accident locale can be abstracted as a maze, and therefore the maze searching algorithm should be thoroughly studied and can be equipped on a rescue robot so as to enhance its capacity and efficiency of search and rescue.

The flood fill algorithm [1] is one of the most popular algorithms used for maze solving. The flood fill algorithm is based on the modified depth-first algorithm, it is purposed to find the shortest path from the start point to the target point instead of finding the target point as fast as possible, thus it is not suit the application of maze searching for rescue purpose.

The D* algorithm and A* algorithm are the classical path finding algorithms and both of them can be used to search the maze. Takayuki Goto & Takeshi [2] had studied the application of both the D* algorithm and the A*algorithm in the Intelligent Transfer System as well as the Robotic Path Planning, and they ascertained the advantages of A* algorithm over D* algorithm. Many variations of A* algorithm with different heuristic function have been proposed, and different heuristic functions may lead to different maze searching performances. In this paper, we select 3 typical A* algorithms and compare their rescue-oriented maze searching efficiency. Since the depth-first search algorithm is basically the A* path finding algorithm without a heuristic so all movement is given equal weight and in a fix sequence, we also adopt the depth-first search algorithm as a benchmark to judge the usefulness of the heuristic functions of A* algorithms. The purpose of our study is to find an A* algorithm that outperforms the others in the rescue-oriented maze searching application, and as a result we can employ this A* algorithm in the physical robot to fulfill the search and rescue task in a real maze.

II. EXPERIMENT SETUP

A. Characteristics of Rescue-oriented Maze Searching

Maze is a typical abstract of the complex situations of accident locale. In this paper, we adopt the "perfect" maze as the environment a robot should search, i.e., the mazes have neither any loops or closed circuit nor any inaccessible area [3].

In the rescue-oriented maze-searching application, the algorithm should focuses on the robot instead of the Maze. This is to say, there is only a robot (a single point) that should search and move through the maze from start point to the target point, and then it should bring the target out of the maze (i.e., the robot should return to the start point from the target point) as fast as possible. In this process, the robot can only get the local information of the maze near the location where it is at current time, because the robot can't look over the whole maze and can only sense a limited local area in the maze using its sensors.

When a real robot be sent on mission of search and rescue in a real situation, it may equipped some instruments that can detect some signals of rescuee, for example, the life detecting radar may be equipped on rescue robot to detect the victims and survivors in the debris of earthquake. A the same time, the survivors be trapped in the ruins may also send some signals to ask for help, for example, miners be trapped in a collapsed mine may ask for help by knocking the steel pipe or the wall. These signals are very helpful heuristic information that can guide the robot rescuer to find the location of rescuee. But

978-1-4244-8039-5/11/$26.00 ©2011 IEEE

because of the complexity of the rescue environment as well as the precision of the equipments, there are no doubt a lot of noises in the signals the robot detected, and as a result, the robot rescuer can't get a precise but an approximate heuristic information of the orientation of the rescuee. Generally speaking, the far the distance between the robot and the target point, the much uncertainty the heuristic information is; and vice versa. Therefore, we introduce some deviations of the target point's orientation information in the heuristic function of A* in the simulations, and the deviations are graphically illuminated in Figure 1 and computed as follows:

$$\begin{cases} x_t' = x_t + \Delta x \\ y_t' = y_t + \Delta y \end{cases} \quad (1)$$

Where, (x_t, y_t) is the actual coordinate of the target in the maze, (x_t', y_t') is the detected coordinate of the target in the maze, and $(\Delta x, \Delta y)$ is the errors caused by both the maze environment and the equipments as stated before.

$$\begin{cases} \Delta x = k\dfrac{l}{L} rand(-1,1) \\ \Delta y = k\dfrac{l}{L} rand(-1,1) \end{cases} \quad (2)$$

k is a weight, L is the Euclidean distance from the starting point to the target point, l is the Euclidean distance from current point (x, y) to target point, $rand(-1,1)$ is a random number between (-1,1).

In this experiment, $k = 40$, the coordinate of start point is (0,0), and the coordinate of target point is (24,24). (x_s, y_s) is the start point (0,0)

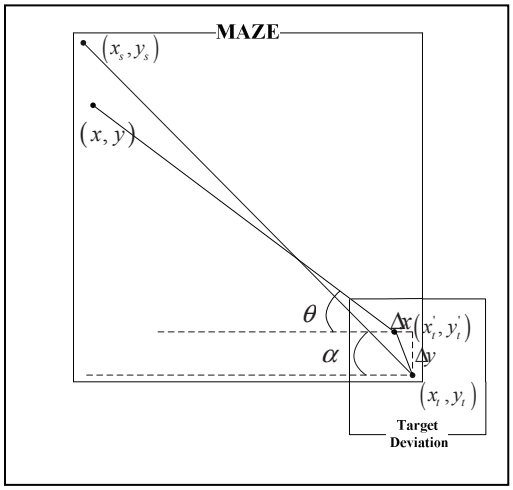

Figure 1. Scope of the target point deviation

B. A Algorithms*

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

Three A* Algorithms with different heuristic function are adopted in the experiment to compare their maze searching capacity and efficiency. We name them A*(1), A*(2) and A*(3) respectively.

The Evaluation Function of A*(1) [4] is:

$$f_1(i) = g_1(i) + h_1(i) \quad (3)$$

Where, i is the current position of robot in the maze, $g_1(i)$ denotes the path-cost function from the start point to the current position i, heuristic function $h_1(i)$ is the Euclidean distance from the current point i to the target point.

The Evaluation Function of A*(2) [5] is:

$$f_2(i) = g_2(i) + h_2(i) + h_2(j) \quad (4)$$

Where, j is the father point of current point, $h_2(j)$ is the Euclidean distance from the father point of current point to target point. This term is added to the father point for improving the search speed, because it reduces the number of nodes.

In A*(3) [6], the estimated value of the angle from current point to the target point is taken into consideration:

$$f_3(i) = g_3(i) + h_3(i) \quad (5)$$

Where, $h_3(i) = \omega_1 \dfrac{l}{L} + \omega_2 \dfrac{\theta}{\alpha}$,

ω_1 and ω_2 are weights, θ is the angle between the line from current point to the target point and the horizontal line (illuminated in figure 1), α is the angle between the line form the starting point to the target point and the horizontal line. They are graphically illuminated in figure 1. $\omega_1 = 0.55$ and $\omega_2 = 0.45$ in the experiment.

The depth-first search algorithm is adopted as a benchmark in the experiment, the depth-first search algorithm is basically the A* path finding algorithm without a heuristic so all movement is given equal weight and in a fix sequence, therefore we can judge the usefulness of the heuristic functions of A* algorithms in maze searching application. Because searching efficiency or search time is the first concern in the search and rescue applications, the depth-first search algorithm

will be stopped as soon as it hit the target point the first time in the experiment.

III. EXPERIMENT RESULTS

Ten perfect mazes of the same size are used in the experiment, and one of the mazes is shown in Figure 2. A maze has 24 ×24 grids, the red grid denotes obstacle and the white denotes the passageway. The start point's coordinates is (0,0), while the target point's coordinates is (24,24).

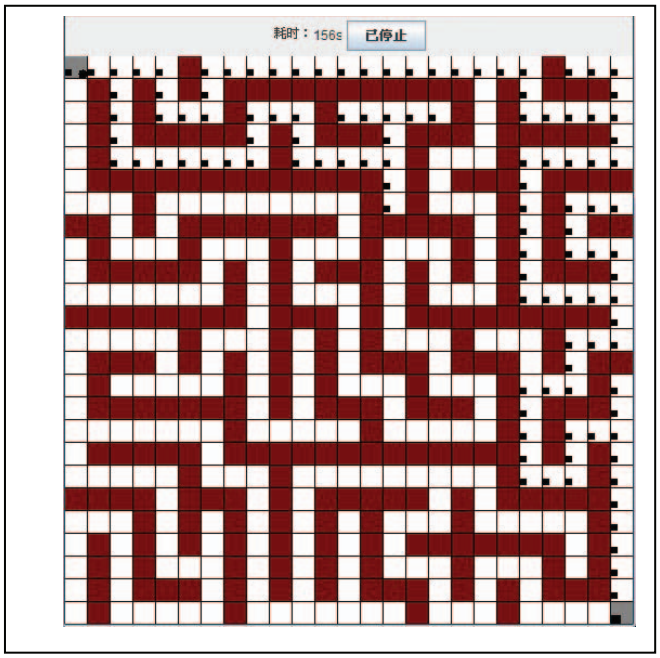

Figure 2. Maze environment

We run the 3 A* algorithms and the depth-first algorithm 10 times on every maze, and record the average time that these algorithms consumed to search and find the target in table 1. The average time of the depth-first algorithm is an integer, because the depth-first algorithm search the maze without any heuristic information, and the errors of target's position detected by the rescue robot has no influence on the search result.

TABLE I. COMPARISON OF ALGORITHMS

average time / algs / maze	$A^*(1)$	$A^*(2)$	$A^*(3)$	Depth-first
Maze1	171	158.4	174.8	236
Maze2	282.8	250.8	297.4	140
Maze3	163.2	148.2	161.6	170
Maze4	201.8	199	206.2	334
Maze5	251	229.4	252.6	454
Maze6	181.6	180	181.2	212
Maze7	193.6	179.2	192.4	188
Maze8	249.8	224.4	240.4	224
Maze9	220	179.2	220.4	324
Maze10	210.4	208	208.4	212

From table 1, we can draw the following conclusions:

(1) The A* algorithms outperform the depth-first algorithm in most mazes. This result validated that the heuristic information is very helpful in maze searching, and an algorithm with heuristic information will usually perform better than those without heuristic information;

(2) A*(3) that employed a heuristic function with angle and distance has not been demonstrated well in this experiment, the reason is: in this experiment, we have added deviations not only on distance but also on angle, so the A*(3) algorithm has no advantage in this searching.

IV. CONCLUSIONS

Through experiment, this article compared searching applications in unknown mazes of A* algorithm and depth-first search algorithm , and applications of three types of A* algorithms with different heuristic functions, 10 mazes are all Perfect mazes, the results, through the experiments and their comparisons, show that: generally, in unknown maze, while only proximate location of target point is known, A*algorithm is better than depth-first search algorithm in searching, however when the heuristic functions are different, searching results are also different. From the above results, A*(2) gets the best searching path, then A*(1) and A*(3). Depth-first search algorithm is of blind when it searches in an unknown maze. The A* algorithms can, during searching, judge the movement of target point by referring heuristic information, it does not need to thumb through the map, so that the calculating complexity is relative simple, and effective fast searching can be achieved. However this way may shielded many points, and the substitution functions who introduced enlighten information can not ensure complete correctness due to complexity of experiences and real issues, the shielded points may be one of the best routed point.

In further work, we will employ this A* algorithm in the physical robot to fulfill the search and rescue task in a real maze, and as stated in the introduction, for searching purpose, to let the robot return back by the route it just passed is not a wise idea, therefore optimization to the return route will be a key part of further work.

REFERENCES

[1] A.Francy Golda, S.Aridha, and D.Elakkiya, "Algorithmic Agent for Effective Mobile Robot Navigation in an Unknown Environment." Intelligent Agent & Multi-Agent Systems, 2009. IAMA 2009.

[2] Takayuki Goto, Takeshi Kosaka, and Hiroshi Noborio, "On the Heuristics of A* or A Algorithm in ITS and RobotPath-Planning, " Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1159-1166, Oct, 2003

[3] http://www.astrolog.org/labyrnth.htm.

[4] Hiroshi Noborio, Keiichi Fhjimura, Yohei Horiuchi, “A Comparative Study of Sensor-Based Path-Planning Algorithms in an Unknown Maze,” Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 909-916,2000.

[5] Zhang renping, Zhou qingzhong, “Updated A* Algorithm and its Application,” Computer Systems & Applications, pp. 98-100, 2009.

[6] Shi hui, Cao wen, “A* algorithm Improvement and Its Application In Path Planning, ” Geomatics & Spatial Information Technology, Vol.32, No.6, pp.208-211, Dec, 2009