Path planning for terrestrial platform based on A-star algorithm

Bochuan Jiang[1,2,a], Yueping Wang[1,2,b] and Zhiqiang Zhao[1,2,c]

[1] Science and Technology on Complex Land Systems Simulation Laboratory,

Beijing 100012, China

2. Army 63961, Beijing 100012, China;

[a]bcjiang05@163.com, [b]ypwang010@163.com, [c]13901304418@139.com

Keywords: military simulation, terrestrial platform, optimal path planning, A-star algorithm.

Abstract. Terrestrial platform mobility is different from that of aerial platform, for the former involves complex geographical elements such as mountainous and potamic terrain. It is also different from intelligent robot mobility on a given plane, as safety factor in simulated military countermining environment should be calculated in terrestrial platform. Concerning path planning problems for terrestrial platform, this paper combines A-star algorithm with factors of safety and passableness, designs a heuristic function and sets up models, in order to propose an optimal path planning method for land armies under practical circumstances. Thus with known antagonistic situation and geographical mapping data in a large-scale military simulation, land military units are able to plan the optimal path to reach the target point through real-time computation. Processes of the algorithm are provided, as well as simulated experiments that validate the performance of the proposed algorithm.

Introduction

In the military simulation community, terrestrial platform requires mobility to reach the target point on the map in performing projected mission or coping with pop-up combat threats. During these processes, complex terrestrial environments and safety factors in mobile area should be taken into consideration. Thus land armies need to search and find the optimal path according to combat situation and geographical information.

Multiple algorithms have been applied to path planning problems. Latombe [1] points out that traditional approaches mainly divide into two categories: the graphic method and the analytic method. The graphic approach includes "connectivity graph" method and "cell decomposition" method. Generic Algorithm (GA), Ant Colony Optimization (Ant), Taboo search intelligent algorithm, and their combinations, are also explored to solve path planning problems [2]. However, all previous algorithms could not eliminate the problem of computational inefficiency, or enlarge its computational capacity. In addition, Dijkstra Algorithm [3] searches directly the panoramic space, and spends too much time on calculation because of denying the information of the target, which fails to meet the real-time demand for simulation.

Despite computational inefficiency, bad performance in real-time practice, and limited capacity as mentioned above, existing models have two other deficiencies concerning path planning for terrestrial platform. The first one is over-emphasizing the safety factor by equating area existing certain dangers with absolute obstacles. It transforms path planning into 2D obstacles avoidance of mobile robot [4], which will drastically limit the range of terrestrial platform mobility. The other is neglecting the safety factor, that is to say, considering only the passableness of the ground, which does not conform to the rivalry environment in military simulations.

In this paper, we combine a heuristic A-star algorithm [5] with practical elements in terrestrial platform mobility, and propose a path planning model for terrestrial platform. As A-star algorithm is introduced with a heuristic function, which makes it possible to make full use of information about target points, the proposed algorithm is a real-time planner. The heuristic function of A-star algorithm could be devised in a flexible way as to improve the computational efficiency. As a result, this paper devises a proper heuristic function, which not only includes geographic conditions and safety factor,

but adjusts the model to the ground passableness and safety conditions in an optimal proportion; moreover, the search efficiency could be greatly improved.

Path planning algorithm for terrestrial platform

The theory of A-star algorithm. A-star algorithm is the most effective method to compute the shortest path from the initial node to the target node on a given static graph. Supposing X as the current node, A-star algorithm generates a heuristic function $f(X)$, which combines the sum of the cost for the current path $g(X)$ and the predicted heuristic cost for the remaining path $h(X)$, to evaluate node X:

$$f(X) = h(X) + g(X). \tag{1}$$

Here, $g(X)$ is the cost value from the initial node to node X, and $h(X)$ is the predicted cost of the optimal path (the path with the minimum cost) from node X to the target node, thus $f(X)$ is the predicted total cost of the optimal path that passes node X. Assume $h*(X)$ as the real cost of the optimal path from node X to the target node, then

$$h(X) \leq h^*(X). \tag{2}$$

On this condition, when costs of all the paths are positive, A-star algorithm is applicable. That is to say, A-star algorithm will find out the optimal path with the minimum cost from the initial node to the target node.

After determining the proper heuristic function $f(X)$, A-star algorithm maintains two tables during computation, denominating the open list and the closed list. The closed list records nodes which have been visited, and the open list records nodes in the current searching area. Nodes in the open list are listed in the order from low to high by their cost values generating through A-star algorithm. The flow chart is presented in Fig. 1.

Generating the heuristic function $f(X)$. Assume A and B are two nodes on the graph that directly connected to each other. There are no other nodes to be investigated between A and B, which means segment AB does not intersect with other paths. Before generating the heuristic function, definitions are given as follows.

$t(AB)$—the index for the passableness of the ground between A and B. $t(AB) \in [0,1]$. Smaller $t(AB)$ denotes worse passableness, and $t(AB) = 0$ means unpasssableness.

$s(AB)$—the index for safety measurement between A and B. $s(AB) \in [0,1]$. Smaller $s(AB)$ denotes worse safety, and means the greatest danger.

$c(AB)$—the cost value of the movement from A to B. It can be checked that $c(AB) = c(BA)$. In order to include the passableness of the ground and the path safety, we define $c(AB)$ as

$$c(AB) = \begin{cases} \dfrac{|AB|}{[t(AB)^\alpha \cdot s(AB)^\beta]}, & \text{if } t(AB) \neq 0 \text{ and } s(AB) \neq 0 \\ \infty, & \text{otherwise} \end{cases} \tag{3}$$

Where, α and β are weight constants. The definition above represents that during the movement along segment AB, the cost value is in direct proportion to the Euclidean distance between A and B ($|AB|$); when the path is unpassable ($t(AB) = 0$)) or the path is very dangerous ($s(AB) = 0$), the cost value for AB are infinite, which means segment AB will not be included in the optimal path. Parameters α and β control $t(AB)$'s and $s(AB)$'s respective influences to the cost function $c(AB)$. If $\alpha < \beta$, $t(AB)$'s influence to $c(AB)$ will be weaker than that of $s(AB)$, and the function will generate the optimal path that emphasizes more on the safety factor. If $\alpha > \beta$, $t(AB)$'s influence to $c(AB)$ will be stronger than that of $s(AB)$, and the generated optimal result prefers to the shorter path to the target point. When $\alpha = 1$ and $\beta = 0$, the safety factor is left out of the problem.
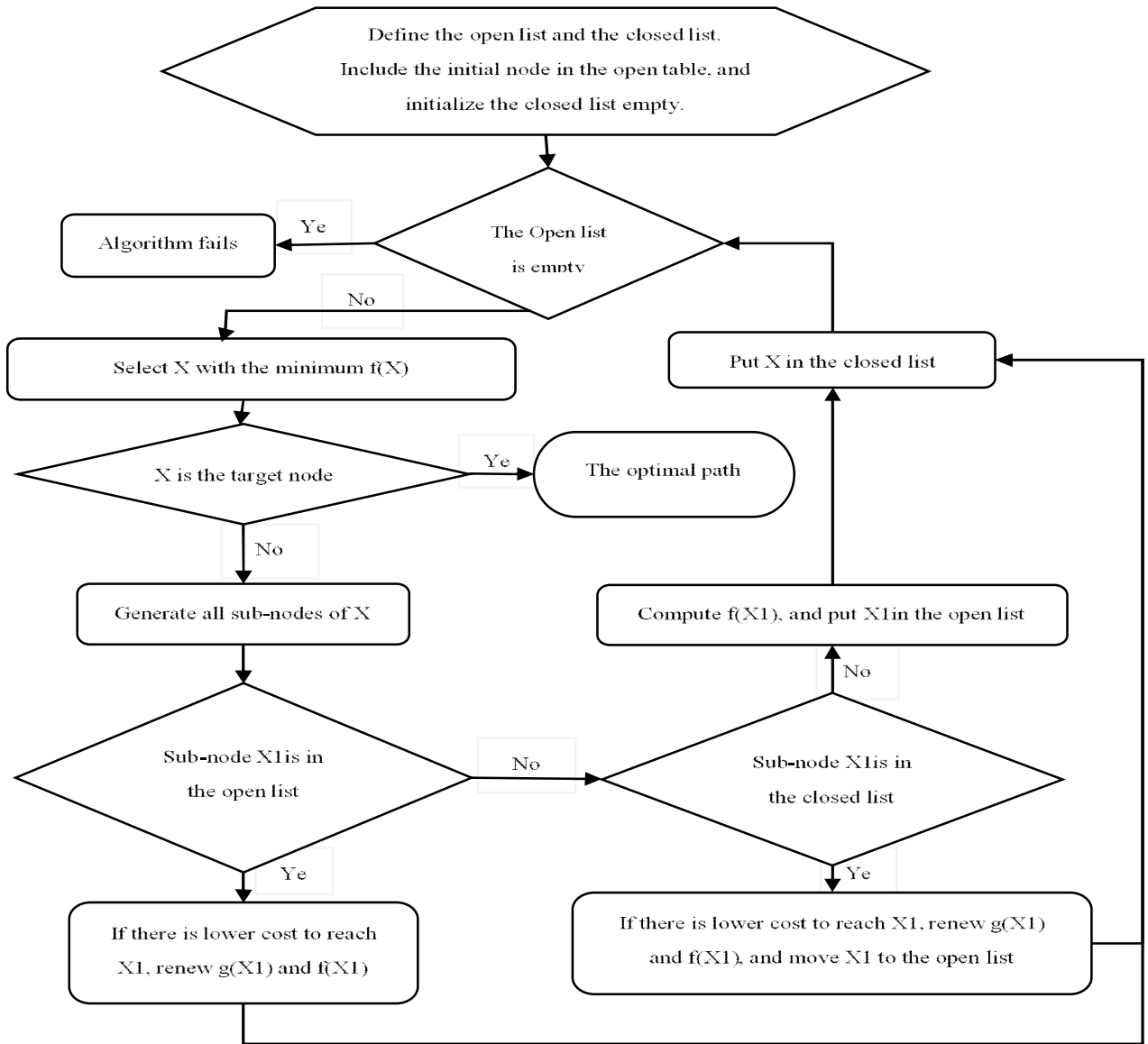
Fig. 1 The flow chart of A-star algorithm

The initial node is denoted by S_0, and the target node D_0. The heuristic function fits the Eq. 1. For a leg on the graph $\Omega_0 = S_0 A_1 A_2 \ldots A_n$, the cost value from S_0 to A_n passing leg Ω_0 is defined as

$$g(A_n) = c(S_0 A) + \sum_{i=1}^{n-1} c(A_i A_{i+1}) \qquad (4)$$

To generate the heuristic function, the predicted cost from the current node to the target node is defined as

$$h(A_n) = |A_n D_0| \qquad (5)$$

Here, the definition is in accordance with Eq. 2, ensuring the proposed A-star Algorithm to find out the optimal path. Assuming the shortest path from A_n to D_0 is $\Omega_1 = A_n A_{n+1} \ldots A_{n+k} D_0$, then

$$h^*(A_n) = c(A_n A_{n+1}) + \cdots + c(A_{n+k} D_0) \geq |A_n A_{n+1}| + \cdots + |A_{n+k} D_0| \geq |A_n D_0| = h(A_n)$$

Therefore, with the heuristic function $f(X)$ generated from Eq. 2, the A-star algorithm provided above is adoptable.

The realization of terrestrial platform mobility model

This section explains the process of practical modeling. Firstly a comprehensive knowledge of the map is given beforehand, which includes the passableness of each position, the distribution of dangerous areas, and respective safety indexes for dangerous areas. These assumptions are made because the passableness and safety indexes are various concerning different situations and in the eyes of different commands-in-chief, thus practical instances should be analyzed according to detailed conditions. The processes are as following:

Step 1. Divide the given map into congruent rectangles, thus transform the problem of the shortest path between nodes into the problem of the shortest path between rectangles. Define the distance between adjacent (neighborhood) rectangles as the distance between the midpoints of these two rectangles. Then the problem of the shortest path between rectangles is transformed to that of rectangles' midpoints.

Step 2. Assume that the movement takes place from a midpoint of an arbitrary rectangle to eight different directions (to east, south, west, north, northeast, northwest, southeast, or southwest), which means to move from the midpoint to eight neighboring rectangles. The distance traveled is the distance between midpoints of two rectangles. For example, suppose l and s represent respectively lengths of the horizontal and vertical sides of the rectangle, then the distance between two horizontally neighboring rectangles (e.g. from east to west) will be l, while the distance between two vertically neighboring rectangles (e.g. from south to north) will be s. As for the rectangles adjacent to each other in the rest four directions, there will be a diagonal relationship $\sqrt{s^2 + l^2}$ to calculate the distance.

Step 3. Let $t(A)$ denote the passableness of the rectangle where node A is located, and $t(B)$ denote that of B. Define the passableness index of the segment between the midpoints of rectangles A and B is the arithmetic average of the passableness of the two rectangles, which can be expressed as $t(AB) = 0.5(t(A) + t(B))$. Define the safety index $s(AB)$ in the same way, i.e. $s(AB) = 0.5(s(A) + s(B))$.

Step 4. Generate the heuristic function based on the method mentioned above, and execute A-star algorithm

Simulations

Test 1. In computer-based experiments, set the map with an equal side length of 10 kilometers, and divide the map into 10,000 small rectangles, which have equal side length of 0.1 kilometer. The function of passableness can be formalized as:

$$t(x, y) = \begin{cases} \left(\sqrt{0.1x} + 0.1y\right)^{-1}, & x \neq 4 \text{ or } y = 4, \\ 0, & x = 4 \text{ and } y \neq 4 \end{cases}$$

Here $t(x, y)$ denotes the passableness of the rectangle which has the midpoint (x, y). Assume $\alpha = 1$ and $\beta = 0$, then the safety factor is left out. The second equation in the above represents that there is an impassable obstacle vertically existing in the position coordinating $x = 4$, where the only passable exit is in the small square with the midpoint $(4, 4)$. Suppose the initial node is $(0, 2.5)$, and the target node is $(10,1)$. Plot the results in Fig. 2 by computer.
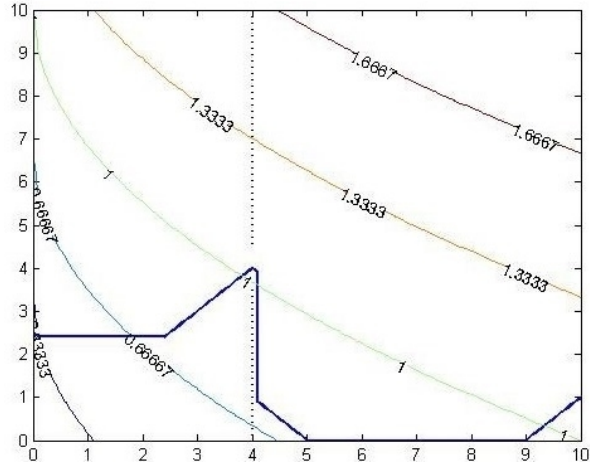
Fig. 2 The optimal path searched by the A-star algorithm

The numeric values of contour lines in Fig. 2 are the reciprocals of $t(x, y)$. Therefore, higher numeric values of contour lines corresponds to smaller $t(x, y)$, which means worse passableness. The dotted line represents the obstacle, which has the only passable exit in the small square (4,4). The thick real line represents the computed optimal path. From the Fig. 2 we can see that the calculated optimal path is feasible. The calculation time is 0.1s, which meet the real-time demand.

Test 2. The safety factor is taken into account in this experiment. Locate a dangerous center in coordinate (2.5,2.5). The function of safety can be formalized as

$$s(x, y) = \begin{cases} |x - 2.5| \cdot |y - 2.5|, & \text{if } |x - 2.5||y - 2.5| < 1 \\ 1, & \text{otherwise} \end{cases}$$

Where, higher numeric value of $s(x, y)$ indicate farther distance away from (2.5, 2.5), which means the location is safer. Apply the same function of passableness as for Test One. Assume $\alpha = 1$ and $\beta = 1$, meaning the safety factor and the passableness are of equal importance. Plot the results in Fig. 3. The circle in the graph is the dangerous area.
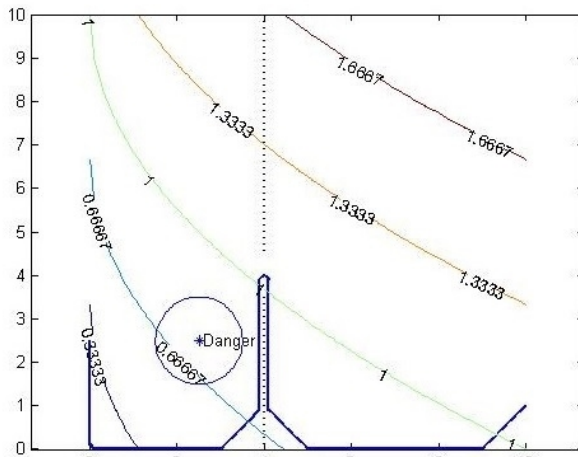

Fig. 3 The optimal path searched by the A-star algorithm

Compare Fig. 2 and Fig. 3, and we can see that the optimal path in Fig. 2 travels across the dangerous area in Fig. 3. By adding a measure of safety in the heuristic function, the algorithm calculate the optimal path which can avoid dangerous areas.

Conclusion

Path planning is an important research area in military simulation. It requires not only to consider factors of passableness and safety, but also to meet the real-time demand. As A-star algorithm is

recognized as an effective heuristic algorithm, it is critical to combine A-star algorithm with the condition of path planning in military simulation in order to solve the problem. This paper transforms the problem of optimal path planning between nodes into the problem of path planning between small rectangles, and when the side length of the rectangle is considerably small, we tend to see points in the rectangle as midpoint of the rectangle, which reduced the complexity of the problem on a large scale. The proposed heuristic function includes factors of passableness and safety with parameters α and β respectively. Different α and β will generate different optimal paths with different preferences and emphases. In conclusion, the proposed algorithm can solve the problem of path planning in military simulation.

References

[1] J. Baraquand, B. Langois and J. C. Latombe: IEEE transactions on Robatics and Automation, Man and Cybernetics Vol. 22 (1992), p. 224

[2] M.]begum, G. K. I. Mann and R. G. Gosine: Applied Soft Computing Vol. 8 (2008), p. 150-165

[3] E. W. Dijkstra: Numerische Mathematic, Vol. 1 (1959), p. 269.

[4] M. Zamirian, A. V. Kamyad and M. H. Farahi: Physics Letters A, Vol. 373 (2009), p. 3439.

[5] S. Russell and P. Norvig in *Artificial Intelligence A Modern Approach* Published by arrangement with the original publisher, Pearson Education,Inc., publishing as Prentice-Hall, 2003(2):97