Abstract. Mountain scenario data is large and complex for there are large amount of semantic information in it. That is the reason why it is difficult for computer to calculate an accurate path in real-time for multi-agent marching in mountain scenario. The paper proposed an m2ACO (multi-agent in mountain environment using ACO) algorithm for multi-agent path planning in Web3D technology. And the pgACO (planar grid ACO) and A* path planning algorithm have been implemented in this paper to compare with the m2ACO algorithm. The experimental results show that the m2ACO is more accurate than the other kinds of path planning algorithms.

Introduction

"Internet+" and "VR+" make the traditional 3D techniques and machine learning algorithm more widely used in Web3D, especially in the large scenario and big data field, which is a research hot point now. It is difficult to load and render in the Web for large mountain scene, and especially with complex semantic information, it would take up more memory, GPU\cite{1} and CPU\cite{2} resources. And the path planning of multi-agent based on the mountain scene is difficult to be gotten on the web page: real time path planning calculation; find an optimal path; visualization of the behavior of multi-agent along the optimal path in real time\cite{3}.

To solve these problems, this paper selects a typical Huangyangjie mountain scene to test. Getting surface height data of the whole mountain scene y, realize the mapping relationship between the mountain ground coordinates and plane grid coordinates, and form the projection relation the 3D terrain and 2D plane grid. In this paper, 3 algorithms are used for the path planning of the mountain scene. Whether the A* path planning algorithm and the ACO path planning algorithm based on plane grid, or the m2ACO path planning algorithm proposed in this paper, they can improve the traditional path planning based on contour line. Because the path planning in this paper solve the optimal path planning based on potential energy contour which is not optimal and easy to fall into the cliff without way, it ensures that as long as there is a way, it will find an optimal path plan.

The Background Knowledge

With the development of the HTML5 standard, Web3D display becomes feasible and can be widely popularized\cite{4}. In fact as early as 2011 Flash3D had achieved a lifelike 3D effect on the page, and the user can interact flexible operation with the virtual scene. However, Flash3D does not support the HTML5 format, and although its Web plug-in can be easily updated, users still cannot avoid the "free plug-in" update operation, which bring bad influence.

Support vector machine is a classical path planning method, which is based on mathematical optimization. It is to solve the classification problem of the support vector machine (SVM)\cite{5-7} by using the interval maximization rule, to obtain the nonlinear integral surface, and then select the suitable intelligent moving path from the separated surface.

The methods based on intelligent path planning mainly includes the genetic algorithm inspired by the biological genetics, the particle swarm algorithm inspired by the birds and fish school, and the neural network algorithm inspired by the human brain neural system. Genetic algorithm\cite{8-11} has fast convergence speed and high quality, and it will have a good initial solution. However, its biggest
flaw is the high computational complexity. Particle swarm optimization algorithm\(^{[12]}\) is a quasi-three dimensional path planning in known local path planning or unknown 3D path planning. The neural network algorithm\(^{[7]}\) has the huge potentiality in solving highly nonlinear and uncertain system control method, but this algorithm needs a lot of planning obstacles, and with the increase of the number of obstacles, the computational complexity is increasing.

**The Mountain Scene Data Is Mapped to 2D Planar**

Huangyangjie of Jinggangshan Mountain, a typical mountain scene, is practical and significant as the research object, especially to meet user’s immersive experience in the mountain battle. In order to ensure the value and significance of the study, through the Jinggangshan red tourism Museum, we get the 3D model of the mountain terrain scene, which is obtained through satellite image data and 3D reconstruction.

![Figure 1. Mapping of 3D scene data in 2D plane.](image)

Mountainous surface information is mapped to the plane grid, to realize the mapping of 3D scene data in 2D plane, including geographic information and semantic information. Geographic information is \((x, y, z)\), and semantic information includes bamboo nail matrix, trenches, and Bush.

The specific algorithm steps are as follows.

**Step1**: Define a planar grid object and the direction of the upper ray;

**Step2**: Initialize ray position that intersects from the grid injection point, and define the variables of rays and terrain: \(yPos\): number = 0;

**Step3**: Start the ray detection function, and the results of the intersection of the ray and the terrain mesh are put into the result;

**Step4**: Extract \(yPos\) value and semantic information from result;

**Step5**: Form 2D plane nodes and the Mapping of corresponding point of 3D mountain surface, and store coordinate and semantic information. The algorithm is executed.

Through the above work, we can obtain the accurate coordinate data of mountain terrain, and can obtain the relationship between the mountain surface and the plane 2D grid, which means the Mapping relationship between 2D and 3D is achieved.

**The \(m^2\)ACO, \(pg\)ACO, and \(A^*\) Path Planning Algorithm**

**The \(m^2\)ACO**

**Algorithm flow chart**: First obtain terrain data of Web3D mountain scene, taking into account the location information and semantic information of each node, and start \(m^2\)ACO path planning. From the beginning, go the next position in turn, and search the end position and determine whether to reach the end. If reaching the end, all the nodes are required to carry out a pheromone update; if not reaching the end, it continues to search for the next position, until the completion of this iteration. If iterations are not wholly completed, it enters the next generation \(m^2\)ACO algorithm, until the
completion of all the iterations, to get an optimal path. The flow of the algorithm is shown in the following figure.

The improved heuristic function: according to the heuristic function to calculate the probability of each point in the visual area:

\[ H_{(i,j,k)} = D_{(i,j,k)} w_1 * S_{(i,j,k)} w_2 * Q_{(i,j,k)} w_3 * Z_{(i,j,k)} w_4 \]  

Here, the \( D_{(i,j,k)} \) is the length of start point to the end point, which push ant selects more shorter path; \( S_{(i,j,k)} \) is safety factor, when the point is not effect to reach, the value of \( S_{(i,j,k)} \) is zero. That pushes ant select the security points. When the people arrived at cliff or the area of bamboo nail matrix, bush or fire area, all these area is as infeasible region. The \( Q_{(i,j,k)} \) is the length from the next point to the end point; \( Z_{(i,j,k)} \) is the fire area, if there is no fire, then \( w_4 \) is zero. The \( w_1, w_2, w_3 \) is coefficient, and they represent the importance of the these factor.

**pgACO algorithm for path planning**

The \( pgACO \) algorithm is the abbreviation of the ACO algorithm on the plane grid, and it is ACO path planning on the plane grid, but not a real ACO 3D path planning algorithm, so the accuracy of the algorithm is lower than \( m^2ACO \). The \( pgACO \) algorithm is based on horizontal grid in Figure 1, and when extracting the x end and y end coordinates according to the end clicked by user, extract the x start and y start coordinates of virtual avatar location. Then calculate the optimal path from virtual avatar to the end.

**Path Planning Based on A* Algorithm**

A* path planning is a classic algorithm, but the error between its path and the optimal path is relatively large, and the error cannot be corrected in the search course by itself.

A* path search algorithm: find the next position from the starting point along eight directions, and A* path finding algorithm on the plane grid is used in the experiment \(^{[12]}\).

**Experimental Results**

(a) \( m^2ACO \) algorithm  (b) \( pgACO \) algorithm  (c) A* algorithm

Figure 3. Three kinds of algorithm path planning results.
The $m^2$ACO algorithm is more accurate than $pg$ACO algorithm and $A^*$ algorithm, and $m^2$ACO algorithm uses time only 174ns, $pg$ACO algorithm uses time is 333ns, $A^*$ algorithm uses time is 83ns.

Summary
This paper proposed an $m^2$ACO algorithm which making path based on the detail semantic information of the mountain scenario. In this paper, another two kinds of algorithms have been implemented and three experiments have done in this paper. From the experimental results we can see that the $m^2$ACO algorithm path planning is real-time and more accurate than $pg$ACO and $A^*$ path planning algorithm.

Acknowledgement
This research was financially supported by the National Natural Science Foundation of China(61272270).

References