# Design of Multi-tasking Cooperative Path Planning Simulation System with Multi-algorithms Fusion

*Jing Zhou[1] and Hui Cheng[2]

[1,2]*School of Mathematics and Computer Science, Jianghan University, Wuhan Hubei 430056, China*
[1]*zhou_8132@163.com,* [2]*Cheng126hui@126.com*

### *Abstract*

*For rescue after natural disasters and public calamity, an intelligent multi-tasking cooperative path and patrol planning and obstacle avoidance simulation system should be established for the complex environment. With the simulation system, the effectiveness and feasibility of the intelligent algorithms can be verified in software, meanwhile the risk of putting the software into use and the development cost of post-hardware are all can be reduced. And the multi-thread technology is used for the VS and Vega Prime simulation engine to control the data acquisition and model driving, then the stable 3D-scene simulation system with multi-barrier and multi-channel in the complex scenes is designed and developed. Five kinds of intelligent and efficient pathfinding and obstacle avoidance algorithms, including the A\* algorithm, the ant colony algorithm, the artificial potential field algorithm, the eight-direction recursive algorithm and the particle swarm optimization (PSO) algorithm, are all integrated into the system, then the individuals in each algorithm are divided into subgroups for each task, thus formed the hierarchical multi-tasking intelligent algorithm, which are suitable for the multi-task collaborative planning in complex scenes, and the algorithms can switch quickly each other, when one pathfinding algorithm fails, the another algorithm can be switched. Finally, the system is proved reliable and practical from the simulation results.*

*Keywords: 3D Scene; multi-tasking; intelligent algorithms; path planning; subgroup algorithm*

## 1. Introduction

In recent years, a variety of large-scale and multi-functional architectures are increased with the rapid economic and social development, such as shopping malls, theaters, stadiums. The rapid urbanization level and building technologies result in increasing occurrence probability of the disaster events which endanger public safety. The intelligent multiple tasks cooperation [1] and patrol planning simulation system for obstacle avoidance (rescue simulation system) is established, which can be applied for post-disaster searching after natural and public disasters. The effectiveness and feasibility of the algorithm can be verified in the software with the simulation system, so the risk of putting the software into use and the development cost of the later hardware can be reduced. The simulation system based on swarm intelligence algorithm for multi-agents cooperative path planning and patrol path planning plays a vital role of safety and security in the development of intelligent cities. Therefore, a multi-tasking cooperative path planning and obstacle avoidance simulation system should be established emergently.

The Vega Prime (VP) [2-3] simulation engine based on VS.NET is built by use of multi-thread technology, and the three-dimensional visual simulation system with multi-channel in complex scenes with much barriers is developed innovatively, meanwhile the algorithm interface is reserved in the system for each intelligent algorithm, and the patrols subsystem for monitoring the target area is added in the path planning system, which can

monitor the case of target-losing. The patrol of aircraft is enabled by the key response, and flight paths of aircraft patrol for target area are planned respectively in the system and the shapes of the patrol track are 8-shaped or oval.

## 2. Design of Three Dimension Scene Simulation System

The development tools of simulation system are listed as following.
(1) Model creation tool - Multigen Creator 3.0.
(2) Physical model driven tool - Vega Prime 2.0.
(3) Simulation platform compiler - VS.NET.

To achieve simulation engine, the VS.net integrated environment is used as the compiled platform for the simulation system, and the visualization simulation is achieved by calling the API function library of Vega Prime.

### 2.1. VS-VP Simulation System Development Process

The development process for visual simulation system based on Vega Prime platform is as following.
(1) Generate modeling (.flt) file by Multigen Creator 3.0.
(2) Establish application profile (.acf) with LynX Prime.
(3) Hybrid programming of Vega Prime API [4] and VS.NET.

Meanwhile, the API method functions provided by Vega Prime are invoked in VS.NET development environment, and the detailed design flow for the simulation system development is shown in *Figure* 1.
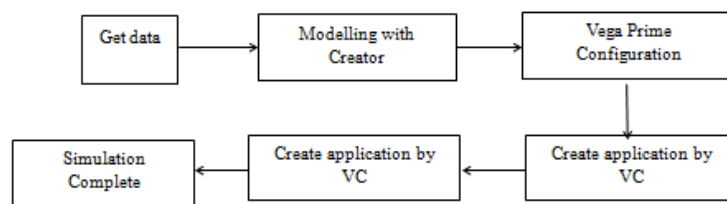


**Figure 1. Establishment Flowchart for Vega Prime Simulation System**

### 2.2. VS.NET and VP -based Application Framework

Steps of establishment for VP application are as follows, which are depended on the VS.NET framework.
(1)  Start the main thread

The static function VP::initialize (argc, argv) is called firstly to initialize the VP program, and the runvpApp() function is called to start the VP sub threads. Then setVpDefine() and postDefine() functions are all called to define the scene, and the vpKernel::instance() function is used to configure the system. Finally, the program enters into the frames loop to execute the vpUpdate() function to update the scene.
(2)  Create the window

An instance of window object is created, then win->setOrigin() function is called to set the original starting point of window, and the win->setSize() function is called to set the window size, then the bottom-left corner of the VS window is regarded as the starting point of VP window, and the size of the VP window is set to be the size of VS window.
(3)  Dynamic loading agents

Define getAcfName() function to get the file name for model container in VP, then the container of VP is determined, and the addchild() function is called to configure the parent-child relationship for VP classes, then setVpDefine() and postDefine() functions

are called to define the scene in the main thread, and vpKernel::instance()->configure() is called to configure and load scene.

(4) Create a transform

The VP transform is an abstract and dynamic coordinate system, with which viewer coordinate can be set to be changed along with the target. In the top-left channel of the established simulation system, transform is used to set the viewer coordinate to be at the rear of the car.

## 3. Research of Multi-tasking Path Planning Algorithm

Multi-tasking cooperative path planning methods [5-8] for simulation agents are inseparable with artificial intelligence algorithms and the intelligent algorithms for multi-tasking are used for multi-path planning and patrol planning in the simulation system. For multi-task and multi paths planning of simulation entities, the individuals in each intelligent algorithm can be divided into subgroups for each task, thus formed the hierarchical multi-tasking intelligent algorithm, which is applicable for multi-tasking planning in the simulation system. The different levels of tasks (targets) are assigned for the simulation entities in a complex scene by the algorithm, and then the reasonable paths can be planned hierarchically.

For the problem of multi agents and multi-tasking planning, the key of multi-tasking problem is to determine which target should be assigned to the corresponding agent. Thus the individuals in intelligent algorithm can be divided into multiple subgroups, and the U-dimension space is constructed to correspond with the task allocations for U agent-subgroups, thus the U-dimension vector is designed for each agent subgroup, where $x_i$ is set as the serial number of target position for the sub-agent to arrive. For example, six agents will arrive at three targets respectively, and then the target position vector x of the subgroups is shown in *Table* 1.

### Table 1. Target Position Vector of Each Subgroup

| Number | 1 | 2 | 3 | 4 | 5 | 6 |
|--------|---|---|---|---|---|---|
| $x_t$ | 1 | 2 | 1 | 2 | 3 | 3 |

From *Table* 1, subgroup 1 and 3 collaboratively arrive at target 1, and subgroup 2 and 4 cooperatively arrive at target 2, and subgroup 5 and 6 arrive at target 3 cooperatively.

Steps of multi-tasking planning which is achieved by intelligent algorithm based on the division of subgroups are as follows.

Step 1: Initialization.

(1) The whole group is divided into some overlapping subgroups.

(2) Each dimension of target position vector X within each subgroup is randomly selected from integer 1 to T. Numbers 1-T (target numbers) represent the serial number of the target.

(3) Initialize the speed and position of each individual in subgroup.

(4) All objective function values of individuals are evaluated, and the initial individual position is set as the optimal solution of individual.

Step 2: Repeat the following steps until the destination is found or the iterations number exceeds maximum.

(5) The transition probabilities are calculated for each individual to select the next position(X, Y). When (X, Y) exceeds its scope, the next position equals to the boundary.

(6) Compare the fitness function value of each individual with its history best value, and then update the optimal value.

(7) Find the nearly optimal solutions in each subgroup and the approximate optimal solution of the total swarming.

Based on the above subgroup algorithm, the targets and tasks of each agent are decomposed for each subgroup in the system, and the targets and tasks are assigned to different levels, thus the reasonable hierarchical planning tasks are accomplished, as shown in *Table* 1.

## 4. Multi-tasking Path Planning Simulation System Design

Each target is allocated to each subgroup by clicking the mouse on the VS-VP program main interface, thus each subgroup respectively obtains each sub-target position. Meanwhile, the coordinates of obstacles are obtained by the algorithm, and then the appropriate map matrix is built. When the path is calculated, the list of path coordinates is returned to VS-VP program. According to the path coordinates, the totally multi-tasking assignment [9-12] for each agent and paths planned for each subgroup are achieved respectively.

(1) Parameters of swarm intelligent algorithm are provided by VP program

The size (length, width, height) of simulation map in 3D scene can be predefined in the swarm intelligence algorithm, and the coordinates of obstacles or threats, motion parameter of simulation agents (such as patrol entrance angle), the target position coordinates and start position coordinates of each simulation agent all can be provided by VP program.

(2) List of paths for VP program is received from swarm intelligent algorithm

All the path-points are put into a list orderly when the path planning algorithm is successfully executed, then the list is returned to the VP application program, and the path is planned according to each coordinate point in the list.

(3) Patrol path planning subsystem

In response to the case of goal-losing in the practical application, trajectory-planning subsystem is added to patrol and monitor target area [13], and the aircraft patrol will be started by pressing the number keys. The flight path is simulated when the aircrafts patrol and monitor the target area in the VS-VP software platform, and the shapes of the patrol tracks are simulated as 8-shaped and oval respectively.

## 5. Simulation Results and Analysis

Initial interface for the simulation system is shown in *Figure* 2, the overlooking-channel is located at the left interface, which can observe the total planned paths, and there are four virtual agents initially in the scene (car 1, 2, and the aircraft 1, 2). And four main channels are on the right interface, each of which binds each agent observer respectively.



**Figure 2. The Initial Interface of Simulation System**

The operations of the whole simulation system are accomplished by the human-computer interaction of keyboard and mouse. The A* algorithm [14] is selected by pressing the 's' key, and the ant colony algorithm [15] can be selected by pressing the 'p' key. The artificial potential field algorithm [16] is selected by pressing 'w' key, and the particle swarm optimization (PSO) algorithm [17] will be selected by pressing 'a' key, while the eight-direction recursive algorithm [14] can be selected by press 'd' key. Meanwhile, a new object is dynamically loaded along a fixed track by pressing 'q' key and aircraft patrols along oval track by pressing the number '1' key, while another aircraft patrols along 8-shape track by press number '2' key.

Click the left-button of mouse on the top-view channel to set the destination of agent 1（car 1）, then the first target task is assigned to the agent 1, and click the right-button of mouse on the top view channel to set the destination of agent 2（car 2）, then the second target task is assigned to the agent 2. Thus, the target positions of target 1 and 2 can be set, which are assigned to the agent 1 and 2, and then the coordinates of target 1 and target 2 are displayed with red font below the top-view channel.

When the different tasks (targets) of each agent are obtained, simulation results for pathfinding and aircraft patrol based on each algorithm are shown respectively from *Figure* 3 to *Figure* 7.



**Figure 3. Agents (Cars) Avoid Obstacle and Find Path Successfully with Ant Colony Algorithm (Blue and White Line are the Planned Track of Car 1, 2 Respectively), meanwhile Aircraft 1 and 2 Enter their Patrol area along a Tangent**



**Figure 4. Agents (Cars) Avoid Obstacles and Find Path Successfully with PSO Algorithm, and Aircraft 1 and 2 Patrol along 8-Shaped and Oval Track Respectively (Blue and 8-Shaped Line is the Planned Track of Aircraft 1,**

**while White and Oval Line is the Track of Aircraft 2) and a New Object is Dynamically Loaded along a Fixed Blue line by Pressing 'q' Key**
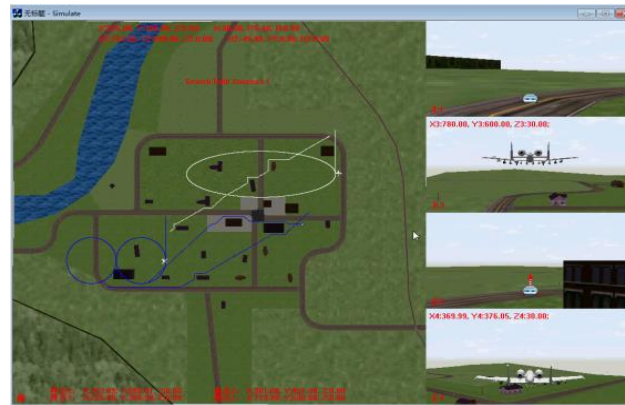


**Figure 5. Agents (Cars) Avoid Obstacles and Find Path Successfully with Artificial Potential Field Algorithm, and Aircraft 1 and 2 Patrol along 8-Shaped and Oval Respectively and a New Object is Dynamically Loaded along a Fixed Blue Line by Pressing 'q' Key**



**Figure 6. Agents (Cars) Avoid Obstacles and Find Path Successfully with the Eight-direction Recursive Algorithm, and Aircraft 1 and 2 Patrol Along 8-Shaped and Oval Respectively and a New Object is Dynamically Loaded along a Fixed Blue Line by Pressing 'q' Key**



**Figure 7. Agents (Cars) Avoid Obstacles and Find Path Successfully with A* Algorithm (Blue and White Line are the Planned Track of Car 1, 2 Respectively)**

The advantages and disadvantages between the various algorithms also can be obtained through the simulation system research. Such as the paths planned respectively by A* algorithm and ant colony algorithm from the same start and end points are shown in *Figure* 7 and *Figure* 8.
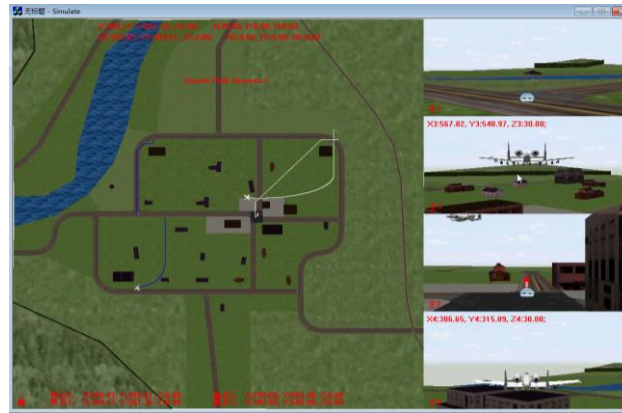


**Figure 8. Agents (Cars) Avoid Obstacles and Find Path Successfully by Ant Colony Algorithm with the same Start Point and End Point as Figure 7**

Shown as *Figure* 7 and *Figure* 8, the path planned by A* algorithm tends to be often against along the obstacle and corner, while the path planned by ant colony algorithm is shorter. So the ant colony algorithm is more efficient than A* algorithm. Compare the recursive algorithm and PSO algorithm in the simulation system. The position of target 1 is maintained unchanged and only the position of the target 2 is reset, then the paths respectively planned by the recursive and PSO algorithm are the two blue lines from the start point to the end point. As shown in *Figure* 9, the path (located at the below blue line) planned by PSO algorithm is shorter apparently, so the PSO algorithm is more efficient than the recursive algorithm.



**Figure 9. Fix the Start Point of the Car and End Point of the Target (Task), then the Two Blue Paths are Planned Successfully by Recursive Algorithm (The Upper Blue Track) and PSO Algorithm (The Lower Blue Track) Respectively**

The system can also be used for quickly switching algorithms, when one pathfinding algorithm fails, another algorithm can be instantly switched. For example the artificial potential field algorithm is easy to fall into local optimal solution which leads to path planning suspension, as shown in *Figure* 10, when the pathfinding algorithm falls into local optimal solution, the pathfinding has been suspended, then the car (agent)

successfully continues planning path and completing the task by pressing the's' key to switch the artificial potential field algorithm to the A* algorithm, as shown in *Figure* 11.



**Figure 10. Path Planning has been Suspended by the Artificial Potential Field Algorithm (White Line Drawn by Car 2 has been Aborted)**
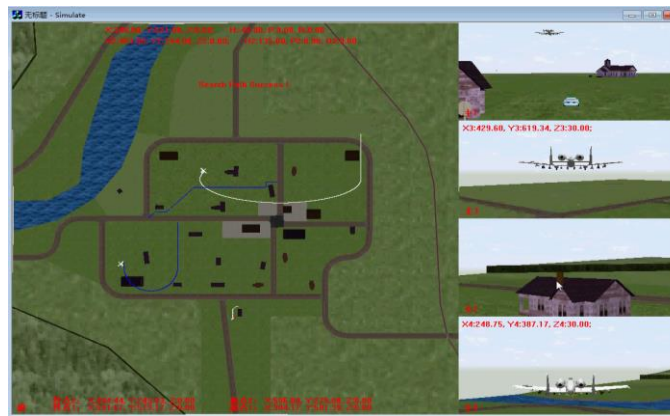


**Figure 11. The Car Continues Finding Path Successfully by Switching to A* Algorithm (The Short and White Line from the Stagnation Point to Target is Planned and Drawn Successfully)**

## 6. Conclusions

An intelligent multitask cooperative path planning 3D scene simulation system is designed, which can find path collaboratively and patrol autonomously in the large and complex map including 23 obstacles. The system fuses five kinds of intelligent and efficient obstacle avoidance path planning algorithms and the algorithms can be switched each other quickly by the keyboard interaction. When one pathfinding algorithm fails, another algorithm can be instantly switched to continue planning the path. Meanwhile the system is reliable and practical that is proved from the simulation results and the effectiveness and feasibility of the subgroup algorithm can be verified from the simulation system.

## Acknowledgments

# References

[1] G. Z. Liu and X. C. Zhang, "Research and Implementation of multi-mission planning simulation system based-on HLA", Computer Engineering and Design, vol. 33, no. 6, **(2012)**, pp. 2410-2416.

[2] Y. Gao and Y. Xing, "Design and implementation of parallel A* algorithm in Vega", Computer Engineering and Applications, vol. 48, no. 7, **(2012)**, pp. 231-239.

[3] Z. L. Hu and J. S. Yu, "Research of Software Integrated Technology of Vega Prim Based on MFC programming Framework", Journal of System Simulation, vol. 21, no. 14, **(2009)**, pp. 4291-4294.

[4] W. J. Pan and Z. H. Xia, "Research on the Multi-objective Distribution of Multi-helicopter and Track Planning", Science Technology and Engineering, vol. 13, no. 34, **(2013)**, pp. 10226-10230.

[5] Y. J. Xi and D. Y. Zhou, "Trajectory Planning for Coordinated Attack with Multiple Uninhabited Air Vehicles", Computer Simulation, vol. 27, no. 3, **(2010)**, pp. 69-73.

[6] H. Huang, D. Q. Zhu and F. Ding, "Dynamic task assignment and path planning for multi-auv system in variable ocean current environment", Journal of Intelligent and Robotic Systems: Theory and Applications, vol. 74, no. 3, **(2014)**, pp. 999-1002.

[7] D. Q. Zhu, H. Huang and S. X. Yang, "Dynamic task assignment and path planning of multi-AUV system based on an improved self-organizing map and velocity synthesis method in three-dimensional underwater workspace", IEEE Transactions on Cybernetics, vol. 43, no. 2, **(2013)**, pp. 504-514.

[8] S. W. Moon, E. M. Oh and D. H. Shim, "An integral framework of task assignment and path planning for multiple unmanned aerial vehicles in dynamic environments", Journal of Intelligent and Robotic Systems: Theory and Applications, vol. 70, no. 4, **(2013)**, pp. 303-313.

[9] S. Z. Yuan, M. Li and J. Tang, "Research on Path Planning for Patrol and Reconnaissance of Loitering Attack Missiles", Electronics Optics and Control, vol. 16, no. 11, **(2009)**, pp. 1-4.

[10] J. Zhou and C. Peng, "Research and analysis of path planning simulation system for three-dimensional complex scene", Computer Simulation, vol. 32, no. 6, **(2015)**, pp. 364-367.

[11] Y. L. Zhang and X. M. Niu, "Simulation Research on Mobile Robot Path Planning Based on Ant Colony Optimization", Computer Simulation, vol. 28, no. 6, **(2011)**, pp. 231-234.

[12] B. Q. Ye and Y. Wang, "Research of Obstacle Avoidance Algorithm for Robot Based on Artificial Potential Field", Journal of Chongqing University of Technology(Natural Science), vol. 26, no. 9, **(2012)**, pp. 82-85.

[13] J. Zhou and X. C. Fu, "New particle swarm optimization algorithm for path planning simulation of virtual character", Journal of Computer Applications, vol. 34, no. 9, **(2014)**, pp. 2562-2565.

[14] X. Wang, X. Dong and H. Zheng, "Vega Prime real-time 3D virtual reality technology", Chengdu: Southwest Jiaotong University Press, **(2012)**.

[15] O. K. Sahingoz, "Flyable path planning for a multi-UAV system with Genetic Algorithms and Bezier curves", International Conference on Unmanned Aircraft Systems, **(2013)**, pp. 41-48.

[16] M. M. Zhan, "Research on Mission Planning for Multi-UAVs", Master Degree Thesis of Hefei University of Technology, **(2012)**, pp. 5-32.

[17] B. G. Wang, "The Research of Unmanned Aerial Vehicles Mission Planning System", Thesis for Master's Degree of Shenyang Aerospace University, **(2013)**, pp. 23-46.

# Authors
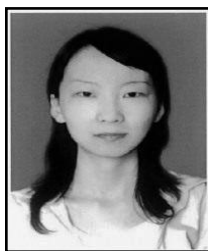
**Jing Zhou**, received the B.E. from Wuhan University of Technology in department of Control Technology in 2002, the M.E. degree in pattern recognition and intelligent systems in 2005 and Ph.D. degree in control science and engineering in 2009 from Huazhong University of Science and Technology (HUST), Wuhan, China. She has published 16 journal articles and conference papers. Her current research interests include virtual reality, intelligent algorithms and system integration.

**Hui Cheng**, received the B.E. degree in radio engineering, M.E. and Ph.D. degree in pattern recognition and intelligent systems from Huazhong University of Science and Technology (HUST). He is a professor of computer science at Jianghan University in Wuhan, China. He has published over 20 journal articles and conference papers. His current research interests include pattern recognition, image processing and intelligent algorithms.