Proceedings of the 4th WRC Symposium
on Advanced Robotics and Automation 2022
Beijing, China, September 20, 2022

2022 WRC Symposium on Advanced Robotics and Automation (WRC SARA) | 978-1-6654-6369-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/WRCSARA57040.2022.9903965

Design and Implementation of Parallel Simulation System for UAV Swarms

Moulu Liu[1], Jinchao Chen[2], Chenglie Du[3], Wenquan Yu[4]

***Abstract*— Unmanned aerial vehicles (UAV) swarm is an organic whole formed by a certain number of drones cooperating with each other. It has more advanced and diversified functions and can complete more comprehensive and complex tasks. UAV swarms currently are faced many problems such as low intelligence, rough cooperative control algorithms, and inaccurate behavior prediction. Simulation modeling plays an important role in the research of these critical technologies. Traditional simulation modeling has serious simulation error accumulation and data lag. We propose the design of parallel simulation system for UAV swarms. We complete the related work of dynamic generation of simulation entities, state synchronization between simulation entities and UAVs, and ultra-real-time simulation. Through the performance test, the experimental results prove that UAV swarm parallel simulation system (UAVSPSS) has the ability to dynamically generate the simulation entity in a short time and complete the state synchronization of the simulation entity in a limited time. These results show that the UAV swarms parallel simulation system provides stability of simulation errors and real-time synchronization of data.**

I. INTRODUCTION

With the development of related technology and the expansion of mission fields for UAV, the action mode of UAVs has begun to change from a single UAV to UAV swarms. In the complex and changeable battlefield environment, the survivability of a single UAV is weak, the task execution ability is limited. In the UAV formation, the drones collaborate on tasks and complement each other's advantages, improving the efficiency of task execution, expanding the scope of tasks, enhancing combat capabilities, and improving environmental adaptability [1], [2].

In the research and development of UAV swarms cooperative control, modeling and simulation is still an important research method for scientific exploration. Many scholars focus on the four aspects of combat modeling framework, combat effectiveness analysis, autonomous control methods and simulation platform development, and research, optimize and innovate UAV swarms collaborative control algorithms [3]–[6]. Undoubtedly, these studies have contributed their own strength to the development of UAV swarms, but there are still many shortcomings. UAV swarms are the very complex system with strong dynamic evolution and adaptability. The simulation system still doesn't accurately predict the behavior of the next moment, which has extreme precision. The reason is that the errors between the model parameters and the physical equipment are constantly accumulating and expanding with the advancement of simulation [7].

Parallel simulation emerges as the times require. In fact, there is no unified statement about parallel simulation at home and abroad. Similar concepts include data-driven, data twin, symbiotic simulation and online simulation. In 2000, in a special academic discussion held by the National Natural Science Foundation of the United States, Frederica Darema put forward the concept and main content of dynamic data-driven application system [8]. According to the requirements of modeling and simulation, dynamic data-driven modeling and simulation dynamically injects data into the simulation modeling process, realizing the real-time evolution of models and parameters [9]. In 2002, Richard Fujimoto etc. proposed the concept of Symbiotic Simulation System [10]. In the same year, Michael Grieves, a professor at the University of Michigan in the United States, mentioned in a paper that a corresponding simulation entity can be constructed on a virtual simulation platform by using physical device data, and the life cycles of the two can be linked in both directions, proposed the concept of digital twin [11]. In 2004, the team of Wang Feiyue from the Key Laboratory of Complex System and Intelligence Science in China proposed the ACP method related to parallel systems [12]. In 2007, the "Deep Green" program proposed by the US Defense Advanced Research Projects Agency (DARPA) attempted to embed simulation into the command-and-control system to provide online real-time simulation for decision-making, improved decision-making capabilities and prediction accuracy [13]. In a word, parallel simulation dynamically updates the model and parameters by collecting the real data of physical system, correcting the simulation system state, improving the reliability of simulation modeling.

In order to solve the problems faced by the current UAV swarms simulation, we propose an ultra-real-time simulation system for UAV swarms based on parallel simulation, and implement UAVSPSS. Compared with the traditional simulation system, UAVSPSS is driven by real-time data, which ensures the stability of simulation errors and the real-time synchronization of data. UAVSPSS and UAV swarms execute in parallel and control each other, optimizing the relevant parameters of UAVSPSS, improving the collaborative control capability of UAV swarms.

[1]Moulu Liu with the Department of Cybersecurity, Northwestern Polytechnical University, Xi'an, China merlotliu@mail.nwpu.edu.cn
[2]Jinchao Chen with the Department of Computer Science, Northwestern Polytechnical University, Xi'an, China cjc@nwpu.edu.cn
[3]Chenglie Du with the Department of Computer Science, Northwestern Polytechnical University, Xi'an, China ducl@nwpu.edu.cn
[4]Wenquan Yu with the Department of software system development, North Automatic Control Technology Institute, Taiyuan, China yuwenquan112233@163.com

The remainder of this paper is organized as follows. Section II tells the overall design of UAVSPSS. Then, in Section III, we describe in detail the implementation of UAVSPSS, including dynamic generation, state synchronization and ultra-real-time simulation for simulation entities. Section IV discusses the experimental results of dynamic generation and state synchronization for simulation entities. Section V summarizes related works of this paper.

II. OVERALL DESIGN OF UAVSPSS

This paper proposes the overall design of UAVSPSS, as shown in Fig. 1. Specifically, it includes six modules: simulation engine, communication module, data processing module, resource management, decision-making and learning module and resource support library.

The simulation engine, the core component of system operation, is responsible for advancing simulation time, scheduling, and running simulation entities, managing data logging, and controlling simulation events. The simulation engine manages and calls each functional module as needed to ensure the complete and accurate of the simulation process. The simulation engine synchronized startup with the parallel simulation system. The simulation engine will call the communication module and data processing module to complete the data interaction with the UAV cluster, call the resource manager to realize the dynamic addition, correction, and deletion of simulation entities, and call the decision-making and learning module to evaluate and study.

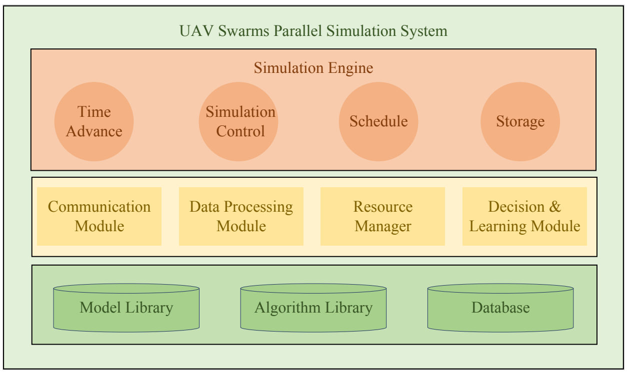

Fig. 1. Overall design of UAVSPSS

The communication module, based on the TCP /IP protocol, ensure data communication between the UAV swarms and the parallel simulation system. A series of data interfaces are defined inside the communication module to ensure the standardization of data output, and simple preprocessing is performed on the data to facilitate further analysis of the data. The data processing module performs a more detailed analysis operation on the data output by the communication module, analyzes it according to the characteristic parameters in the data, and formatted storage it.

The resource manager is responsible for real-time dynamic addition, modification, and deletion of simulated entities. According to the entity parameters obtained by data processing, the model library is matched, and the corresponding simulation model is associated, and is instantiated and loaded into the parallel simulation system. In addition, in the process of interaction between the UAV swarms and the parallel simulation system, to ensure the consistency of the state and information of the real UAV and the simulation entity, revise the parameters of the simulation entity by the real-time data which the UAV swarms provide.

The decision-making and learning module in charge of making decision and study by calling appropriate algorithms. Based on the real-time data provided by the UAV swarms, a reasonable plan is designed according to the task planning algorithm, and the path planning algorithm is used to design a reasonable path during the timeout simulation process to ensure the completion of the task. During the entire simulation process, the scheme is evaluated through a suitable evaluation algorithm, the optimal strategy is finally selected and sent to the UAV swarms. After the UAV swarms is executed, the algorithms, models and systems are corrected and optimized through the execution results.

The resource support library provides basic data support for the parallel simulation system to dynamically generate simulation models, modify simulation entities, and timeout simulation, including databases, algorithm libraries, and model libraries. The database mainly stores the basic data, decision historical data and simulation historical data of the system. The algorithm library contains the core algorithms necessary for the UAV to perform tasks such as path planning, task allocation, and formation control. The model library contains mainstream UAV models for the generation and simulation of simulated entities.

III. IMPLEMENTATION OF UAVSPSS

UAVSPSS is driven by the real-time data of physical equipment. UAVSPSS provides the full process of UAV swarms simulation, including dynamic generation of simulation entities, state synchronization of simulation entities and ultra-real-time simulation. In this section, we discuss in detail the implementation of UAVSPSS. Firstly, Subsection A talks about the full process of simulation entities generation, including generation, association, load. Next, Subsection B explains what happened in the whole process of simulation entities state synchronization. Finally, Subsection C describes in detail how ultra-real-time simulation works.

A. *Dynamic Generation of Simulation Entities*

In a parallel simulation system, the dynamic generation of simulation entities is a necessary operation to ensure that the simulation engine works normally and does not affect the entities being simulated. For the simulation entities that need to be dynamically added, the real system will provide the parallel simulation system with the drone's serial number, model, and related parameters such as current power, speed, and size. After UAVSPSS receives the data, it checks whether the drone simulation entity already exists by the unique identifier of the drone: if it exists, the drone simulation entity is directly called, and the matching is completed. if it does not exist, according to the drone The model and related parameters are matched in the model library, the UAV simulation entity is created and initialized, and the simulation entity is loaded in the simulation engine. If the corresponding model is not found, a pop-up warning will be displayed. The specific process is shown in Fig. 2.

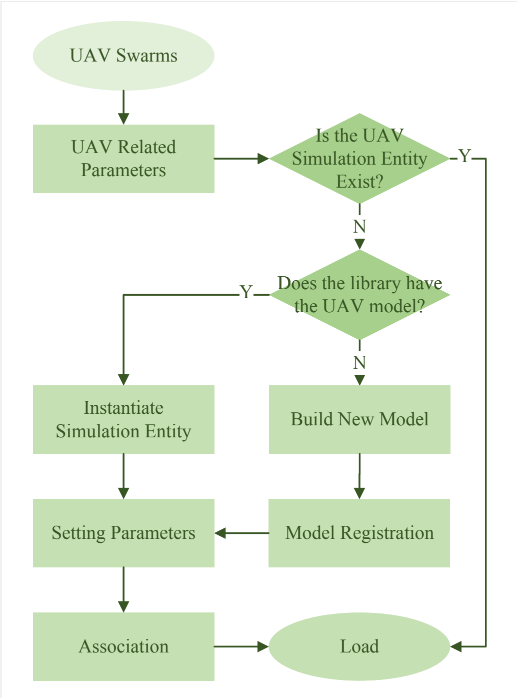

Fig. 2. Dynamic generation of simulation entities

B. State Synchronization of Simulation Entities

The environment in which the UAV swarms perform tasks is usually unknown, so changes in data such as status and operating parameters are inevitable during the execution process. With the advancement of the task process, to ensure the accuracy of the simulation of the UAV swarms by the simulation entities in the parallel simulation system, according to the real-time detection data of the UAV swarms and the UAV state information, system synchronize the state of simulation entities. The specific process is shown in Fig. 3.

The UAV swarms send the UAV state update message to UAVSPSS when the state changes. UAVSPSS parses the received data, performs consistency check, does nothing for entities that do not need to be changed, adds the state change list to the entities that need to be changed, and finally traverses the state list and updates the entity state. After the update is complete, the consistency check is performed again to complete the state synchronization. Due to the complexity and variability of the real environment, the UAV may lose contact or be damaged at any time. In this case, the UAV cannot send any information to the parallel simulation system in time. Therefore, if the state doesn't change, the simulation entity will periodically send probe messages to the UAV. If the UAV does not reply for a long time, the UAV should be regarded as disconnected, and can be regarded as an obstacle or danger if necessary.

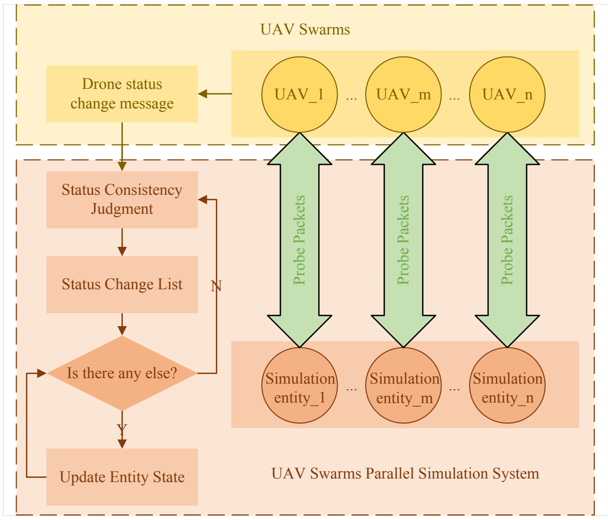

Fig. 3. State synchronization of simulation entities

C. Ultra Real-time Simulation

Ultra-real-time simulation is the core function of the parallel simulation system, which aims to help UAV swarms efficiently complete tasks, better deal with emergencies, and minimize the damage of UAV members in complex and changeable external environments. Ultra-real-time simulation includes two parts: timeout simulation and real-time simulation. In real-time simulation, through high-precision simulation of real UAVs, the situation at the next moment can be predicted as accurately as possible. In ultra-real-time simulation, multi-branch quickly simulation is performed, analyzed, and evaluated to obtain the optimal strategy, which is pushed to the physical environment for execution.

According to the above functional requirements, the specific process of ultra-real-time simulation implementation is shown in Fig. 4. UAV swarms detect and perform tasks in real environments and send data to UAVSPSS. UAVSPSS processes the received real-time data, obtains information such as the current state of the UAV, the surrounding environment, and the current task, and then predicts the behavioral intention of the UAV based on the algorithm and historical database, judges the possibility of its multi-branch. For behavioral states without branching intentions, perform real-time simulation to predict the situation at the next moment, and send possible exception events to the UAV swarms to make better decision. For cases with multiple possible solutions, system saves the current state, generate copies, models, and solutions with a similar number of branches, and prepares for hyper-real-time simulation deduction in the simulation engine.

UAVSPSS starts to execute the interrupt program after the state is saved, like the interrupt mechanism of the operating system. The interrupt program here is the What-if deduction program, that is, the deduction method of what results will be caused if the program is executed. Loop through the

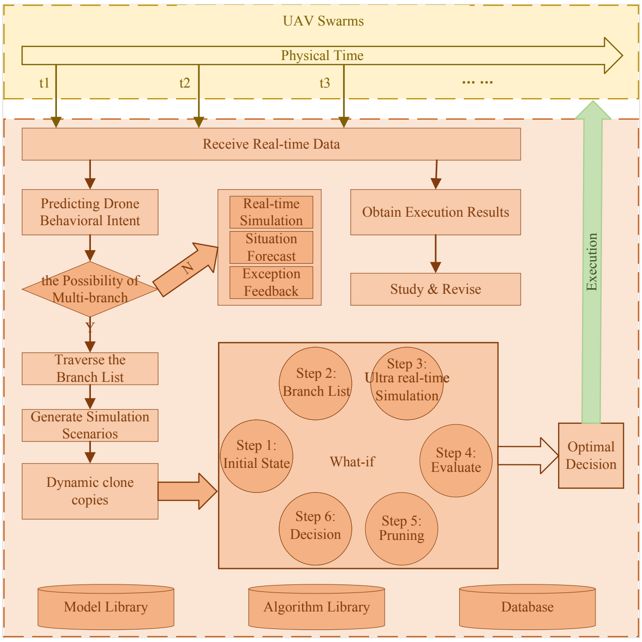

Fig. 4. Ultra-real-time simulation

TABLE I
EXPERIMENT SOFTWARE & HARDWARE ENVIRONMENT

CPU	Intel(R) Core(TM) i5-10400F 2.90GHz
Memory	16.00GB
Operation System	Windows 10
Development Platform	Qt 5.11.2

branch list, simulate and deduce each scheme in the shortest time possible, and evaluate it according to the analysis and evaluation function. For branch schemes that do not reach the expected threshold or have exceptions, they should be directly discarded to improve the quality of decision-making. After ultra-real-time simulation, the optimal decision is pushed to the UAV swarms for execution. System obtains the execution result, and the model is corrected and learned through the deviation between the actual result and the expected result.

IV. EXPERIMENT

A. Experiment Environment

Based on the design scheme described above, this paper implements a coarse-grained simulation system for UAV swarms based on parallel simulation under the Windows operating system. At present, there are many excellent simulation platforms on the market, such as ROS, MATLAB, etc. These platforms are very comprehensive and refined in function and granularity, and there are still excellent teams optimize them. Therefore, the realization of this system doesn't focus on the fineness of the simulation but reflects the idea of parallel simulation.

The practicability of UAVSPSS should be firstly verified to prepare further development and optimization. In this paper, the dynamic generation time of simulation entities is tested to judge whether the system can accurately dynamically generate and load simulation entities in a short time. The experiment is carried out on the state synchronization delay of the simulated entity, to judge whether the system can complete the state synchronization in a limited time. In this paper the experiments are completed in the hardware and software environment shown in Table 1.

B. Results Analysis

In this paper, the dynamic generation delay and state synchronization delay of the simulated entity are tested. The following are the relevant experimental results and analysis.

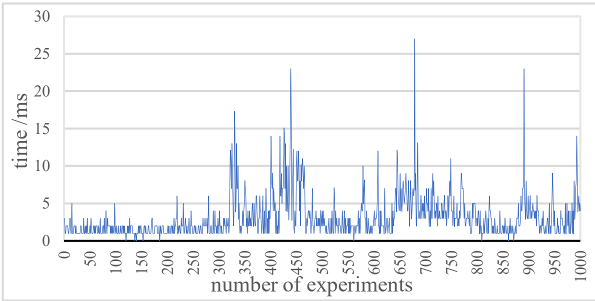

Fig. 5. Simulation entity dynamic generation delay

The UAV swarms parallel simulation system dynamically generates and associates simulation entities for newly connected UAV. Among them, the dynamic generation delay of the simulation entity is the total time that the parallel simulation system receives the UAV connection signal, matches the model, generates the simulation entity, associates the model and the system successfully loads the simulation entity. We repeated 1,000 simulation entity dynamic generation experiments, performed statistical calculations on the experimental results, removed invalid data, and calculated that the average delay was 3 ms, the median delay was 2 ms, and the maximum and minimum delays were 27 ms and 1 ms. In particular, 85.8% of the data is between 1 ms and 5ms, 11.6% of the data is between 5 ms and 10 ms, and only 2.6% of the data is between 10 ms and 30 ms. The experimental results shown in Fig. 5, indicates that the dynamic generation delay of the simulated entity is relatively stable and lower time-consuming. The delay from initialization to loading of most entities can be controlled within 10 ms, indicating that the system performs well. However, there are still a few invalid data and data with large delay in the experiment. The generation process ended prematurely, which is caused by the failure of the UAV connection, produces the invalid data. The reason of high delay is too many parameters and different task complexity of UAV. The dynamic generation of the simulation entity is before the execution of the task, and the UAV is still not taking off at that moment, so the delay of tens of millisecond is still within the acceptable range.

In the process of parallel simulation, in order to ensure the accuracy of the simulation, the physical equipment needs to synchronize state information with the simulation entity

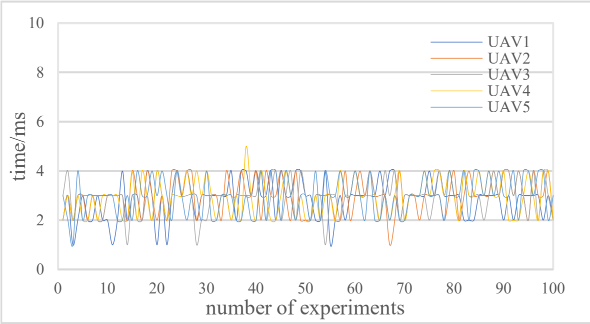

Fig. 6. State synchronization delay of multiple simulation entities

in real time, which requires the parallel simulation system to process the state synchronization information efficiently and accurately. In this paper, the state synchronization delay of the simulated entity is tested, and 5 UAVs are connected to perform 100 state information synchronization respectively. Among them, the state synchronization delay of the simulation entity is the total time from receiving the information from UAVSPSS to the completion of the state synchronization of the simulation entity. As for the communication delay from the UAV to UAVSPSS isn't considered in this paper. The data in Fig. 6 indicates that there is still a small fluctuation in the state synchronization delay of each simulated entity, and the fluctuation range is between 1 ms and 4 ms, fortunately there is no special exception data. Fig. 7 summarizes the statistical data such as average delay, median delay, maximum delay and minimum delay. We found that the average and median synchronization latencies for all simulated entities were around 3 ms, with the fastest reaching 1 ms and the slowest being 5 ms.

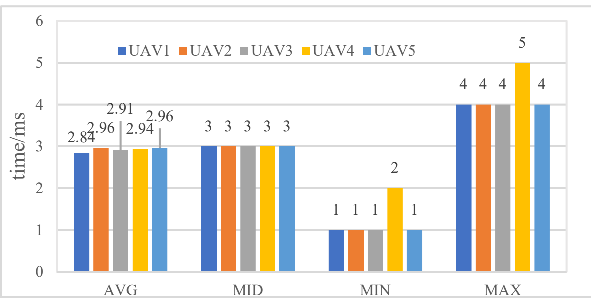

Fig. 7. Comparison of feature data of multiple simulation entities

The experimental data and analysis show that, in UAVSPSS, the dynamic generation delay of the simulation entity can be controlled within 10 ms, and the performance is stable, which indicates that the system can dynamically generate and load the simulation entity during a short time. At the same time, the state synchronization delay of the simulation entity is stable within 4 ms, and the performance is good, which can ensure that the system can complete the state synchronization of the simulation entity within a limited time.

V. CONCLUSIONS

This paper aims to design and implement an ultra-real-time simulation system for UAV swarms based on parallel simulation. We design the overall constructure of UAVSPSS. We implement dynamic generation, state synchronization and ultra-real-time simulation for simulation entities in UAVSPSS. Based on a quantitative and qualitative analysis of the experimental results of the dynamic generation delay and state synchronization delay of simulation entities, it can be concluded that the dynamic generation delay of simulation entities fluctuates significantly but the value is low, and the state synchronization delay of simulation entities is stable within the low delay range. The results indicate that the system can dynamically generate simulation entities in a short time and complete the state synchronization of simulation entities in a limited time. Low-latency synchronization of state information of simulation entities is the first essential to ensure the accuracy of real-time simulation. Although invalid data and high latency appeared in the experiment, this situation reflects the inadequacy of the system, and the system needs to be optimized before being put into use. In addition, the granularity of the simulation and the correction of the model also should be considered.

Based on these conclusions, the next step should optimize the system in the following aspects: (1) Data integrity should be ensured as much as possible to reduce the need to transmit data irrelevant to state synchronization through state synchronization when the simulation entity is dynamically generated. (2) Optimize the event manager to ensure that when multiple UAVs perform state synchronization at the same time, the synchronization is completed according to priority and time sequence. (3) Combine the existing simulation platform to carry out higher-precision simulation. (4) Combine better UAV swarms cooperative control algorithms and model learning frameworks.

ACKNOWLEDGMENT

This research is supported in part by the National Natural Science Foundation of China No. 62106202, the Aeronautical Science Foundation of China No. 2020Z023053004, and the National Key Research and Development Program No. 2018YFB2101304.

REFERENCES

[1] Niu Yifeng, Xiao Xiangjiang, and Ke Guanyan, "Analysis of UAV Swarm Operation Concept and Key Technologies," National Defense Science and Technology, vol. 34, no. 05, pp. 37–43, 2013.

[2] Zong Qun, Wang Dandan, Shao Shikai, Zhang Boyuan, and Han Yu, "Research Status and Development of Multi-UAV Cooperative Formation Flight Control," Journal of Harbin Institute of Technology, vol. 49, no. 03, pp. 1–14, 2017.

[3] J. Chen, M. Li, Z. Yuan, and Q. Gu, "An Improved A* Algorithm for UAV Path Planning Problems," in PROCEEDINGS OF 2020 IEEE 4TH INFORMATION TECHNOLOGY, NETWORKING, ELECTRONIC AND AUTOMATION CONTROL CONFERENCE (ITNEC 2020), 2020, pp. 958–962.

[4] Y. Sun, J. Chen, and C. Du, "Path Planning of UAVs Based on Improved Ant Colony System," in 2020 IEEE International Conference on Progress in Informatics and Computing (PIC), Shanghai, China, Dec. 2020, pp. 396–400. doi: 10.1109/PIC50277.2020.9350789.

[5] Y. Sun, J. Chen, C. Du, and Q. Gu, "Path planning of UAVs based on improved Clustering Algorithm and Ant Colony System Algorithm," in PROCEEDINGS OF 2020 IEEE 5TH INFORMATION TECHNOLOGY AND MECHATRONICS ENGINEERING CONFERENCE (ITOEC 2020), 2020, pp. 1097–1101.

[6] Zou Liyan, Zhang Mingming, Bai Junru, and Wu Jian, "Research Review of UAV Swarm Combat Modeling and Simulation," Tactical Missile Technology, no. 03, pp. 98–108, 2021, doi: 10.16358 /j.issn.1009-1300.2021.1.141.

[7] Duan Wei, "Connotation, Development and Application of Parallel Simulation," Journal of Command and Control, vol. 5, no. 2, pp. 82–86, 2019.

[8] F. Darema, "Dynamic Data Driven Applications Systems: A New Paradigm for Application Simulations and Measurements," in Computational Science - ICCS 2004, Berlin, Heidelberg, 2004, pp. 662–669. doi: 10.1007/978-3 -540-24688-6_86.

[9] Han Shoupeng, Qiu Xiaogang, and Huang Kedi, "Dynamic Data-Driven Adaptive Modeling and Simulation," Journal of System Simulation, no. S2, pp. 147–151, 2006.

[10] R. Fujimoto, D. Lunceford, E. Page, and A. M. Uhrmacher, "Grand challenges for modeling and simulation," Schloss Dagstuhl, vol. 350, 2002.

[11] M. Grieves, Origins of the Digital Twin Concept. 2016. doi: 10.13140/RG.2.2.26367.61609.

[12] F.-Y. Wang and S. Tang, "Artificial societies for integrated and sustainable development of metropolitan systems," IEEE Intell. Syst., vol. 19, no. 4, pp. 82–87, Jul. 2004 , doi: 10.1109/MIS.2004.22.

[13] Zhou Yun, Huang Jiaomin, and Huang Kedi, "Research Review on Key Technologies of the Deep Green Project," Journal of System Simulation, vol. 25, no. 07, pp. 1633–1638, 2013, doi: 10.16182/j.cnki.joss.2013.07.021.