Hindawi
International Journal of Aerospace Engineering
Volume 2022, Article ID 5261663, 22 pages
https://doi.org/10.1155/2022/5261663

Research Article

Group Mobility Model for Complex Multimission Cooperation of UAV Swarm

Xiaoyan Gu,[1] **Feng He**,[2] **Rongwei Wang**,[2] **and Liang Chen**[1]

[1]*School of Information Management, Beijing Information Science and Technology University, Beijing 100192, China*
[2]*School of Electronic Information Engineering, Beihang University, Beijing 100191, China*

Correspondence should be addressed to Feng He; fenghe@buaa.edu.cn

Received 14 November 2021; Accepted 11 December 2021; Published 7 January 2022

Academic Editor: Jinyang Xu

In the unmanned aerial vehicle (UAV) swarm combat system, multiple UAVs' collaborative operations can solve the bottleneck of the limited capability of a single UAV when they carry out complicated missions in complex combat scenarios. As one of the critical technologies of UAV collaborative operation, the mobility model is the basic infrastructure that plays an important role for UAV networking, routing, and task scheduling, especially in high dynamic and real-time scenarios. Focused on real-time guarantee and complex mission cooperative execution, a multilevel reference node mobility model based on the reference node strategy, namely, the ML-RNGM model, is proposed. In this model, the task decomposition and task correlation of UAV cluster execution are realized by using the multilayer task scheduling model. Based on the gravity model of spatial interaction and the correlation between tasks, the reference node selection algorithm is proposed to select the appropriate reference node in the process of node movement. This model can improve the real-time performance of individual tasks and the overall mission group carried out by UAVs. Meanwhile, this model can enhance the connectivity between UAVs when they are performing the same mission group. Finally, OMNeT++ is used to simulate the ML-RNGM model with three experiments, including the different number of nodes and clusters. Within the three experiments, the ML-RNGM model is compared with the random class mobility model, the reference class mobility model, and the associated class mobility model for the network connectivity rate, the average end-to-end delay, and the overhead caused by algorithms. The experimental results show that the ML-RNGM model achieves an obvious improvement in network connectivity and real-time performance for missions and tasks.

1. Introduction

Over the last decade, the usage of UAVs has been strictly increasing in the area like agriculture, construction, industry, and defense. Usually, several UAVs are considered to perform a complicated mission cooperatively as they can handle practical tasks with complexity, diversity, and relevance, such as close air support, reconnaissance, air defense, and coordinated attack. Meantime, the UAV swarm can improve operational performance through a reliable Ad hoc network.

UAV warm is a complex system, which involves many aspects, such as task allocation, path planning, effectiveness evaluation, and networking protocol. Among the above aspects, networking plays the most important role for UAV mission execution [1] and usually constructs a hierarchical cluster network. To ensure the stability and reliability of the UAV cluster network, numerous networking methods and routing protocols have been proposed and analyzed to enhance the networking performance. For task allocation and cooperative execution, many scholars have already made a lot of achievements. Li et al. and Tu [2, 3] first proposed the concept of avionics cloud based on UAV swarm and designed a cloud model for avionics system integration. After that, Lu et al. [4] described a hierarchical and clustering architecture for avionics systems to meet the new requirements of information interaction and resource sharing in future large-scale and complex combat scenes. Wang et al. [5] also proposed a multilayer task scheduling model for layered and clustered UAV swarms. Their research mainly focused on the cooperative task processing and

resource scheduling for the UAV group but did not consider the group mobility model which in fact has an obvious influence on UAV mission execution.

Since UAVs are always in a mobile state during operation, if the group mobility of UAVs is not considered during networking or task allocation, the lack of the mobility model maybe results in serious loopholes or potential risks. Especially, while considering the real networking environment, the selection of mobile model is of great importance [6]. For example, some scholars have suggested that mobile nodes with different mobility characteristics could dramatically affect FANET network performance. Therefore, the study of group mobility models suitable for UAV clusters has important practical significance. The research on the mobility model for UAVs has been addressed in the literature, which can be roughly divided into two categories: the mobility model strategy and the mobility model performance test. For the mobility model strategy, the research can be further divided into three kinds: firstly, the random class mobility model [7–14], including the random waypoint mobility (RWP) model, random walk mobility (RWM) model, random direction mobility (RDM) model, and Gaussian Markov mobility (GMM) model; secondly, the reference class mobility model [15–19], including the reference velocity group mobility (RVGM) model, reference zone mobility (RZM) model, and other models derived from the reference point group mobility (RPGM) model; and thirdly, the associate mobility model [20–28], including the pheromone exclusion mobility (PEM) model and connected overlay mobility (COM) model. The above mobility models established plenty and consistent principles with the experimented movement rules for UAV. However, in UAV swarm practical collaboration scenarios, the abovementioned mobility models only consider the physical factors of node movement, such as speed, direction, and position of nodes, and do not take into account the indicators associated with the collaborative mission.

Since the mobility model needs to consider the application scenario, scholars are now developing a flexible mobile framework that allows us to model different applications and network scenarios. For UAV combat scenarios, literature [11, 13, 15, 16, 18–25, 29–32], respectively, modeled and analyzed the UAV movement problem from the perspectives of network coverage, connectivity, routing, etc. Among these jobs, literature [19, 23, 31] mainly designed appropriate mobility models for UAV clusters that perform different tasks such as attack, defense, and detection. These studies are more suitable for small-scale UAV clusters that implement a single task, but for large-scale UAV swarms, even a single UAV maybe perform different types of tasks. Meanwhile, the existing models seldom consider the various resource requirements of the tasks and the relationships between tasks. In a highly dynamic environment, the mobility model has an important impact on network performance and task execution efficiency. An appropriate mobility model will effectively increase the maintenance time of communication links and increase the connectivity of the network nodes and shorten the completion time of mission tasks. Therefore, it is an urgent problem to design a suitable mobility model for UAV cluster.

In this paper, a group mobility model was studied and designed, which is suitable for the scenario where UAV cluster performs complex tasks collaboratively. It solves the problem of how to move the large-scale UAV swarm when they are executing associated tasks through a hierarchical task dispatching method, eventually, to improve the task real-time ability and network connectivity. Based on the multilayer task scheduling model [5], the existing RPGM algorithm was improved and the multilevel reference node group mobility (ML-RNGM) model for the UAV swarm was proposed. To evaluate the performance of the proposed group mobility model, statistic data related to task scheduling and network link for better analysis were added. Three experiments were designed to analyze the performance of the ML-RNGM model with other classical mobility models.

The remainder of this paper is organized as follows: In Section 2, it presents the related work about mobility models in recent years. In Section 3, the task scheduling architecture for the UAV swarm is presented. In Section 4, the proposed UAV swarm mobility model and performance indicators are described in detail. The contrast experiments and the results obtained from the experiments are given in Section 5. Finally, conclusions are drawn in Section 6.

2. Related Work

Mobility model of flight vehicles has been widely studied recently. It should present the movement of UAV nodes and how their position, acceleration, and velocity change over time [33]. Table 1 gives a basic result of sorting the corresponding studies on the mobility model in recent years. According to the main movement strategies of these models, they can be divided into three categories: the random class, the reference class, and the associate class. The random class mobility model mainly means the movement strategy of the position and speed for the next moment depends on a random generation algorithm. As the reference class, nodes need to refer to the position and speed of other reference nodes. For the associate class model, nodes rely on the message interaction between nodes when they move.

The random class mobility model includes the traditional random mobility models, such as RWM, RDM, RWP, and the improved models, such as the Gaussian Markov mobility model, smooth random mobility model, and paparazzi mobility model. The traditional random class models randomly generate the velocity or direction of the state at the next moment according to the given time interval and the random strategy. For example, in the RDM model, when the node reaches the boundary, the speed and direction will be changed randomly. For the improved random mobility models, they consider the fact that the unconstrained movement relying on random strategy may easily lead to the phenomena of uneven distribution and density wave, etc. [8]. The Gaussian Markov mobility model generates the velocity and direction at the next moment according to the influence of the previous moment. The smooth random mobility model uses the derivative method to generate the state change for the next moment to ensure a smooth speed and direction. The paparazzi mobility model directly

Table 1: Mobility model categories in Ad hoc in recent years.

Class	Mobility name	Main schemes	Related literature
Random class	Random walk mobility	Random but limited speed and angle	[7]
	Random direction mobility	Random speed and angle. Not changing speed and angle until encountering borders	[8]
	Random waypoint mobility	Based on the random walk mobility by adding a wait time when arriving at the destination	[9]
	Gauss-Markov mobility	Similar to the random walk mobility, but adding a speed correlation factor	[10, 11]
	Smooth random mobility	Similar to the random waypoint mobility, but with smoothed changes for speed and direction	[12]
	Paparazzi mobility	Paparazzi movements: stay, waypoints, scan, oval	[13, 14]
Reference class	Reference point group mobility	Logical reference point and center which will affect the speed of the next moment	[15, 16]
	Particle swarm mobility	Random speed and waypoints, collision-free adjustment by particle swarm algorithm	[17]
	Reference velocity group mobility	Reference point with speed and direction, similar to reference point group mobility	[18]
	Freeway mobility	Reference direction which emulates the freeway motion	[19]
Associate class	Pheromone repel mobility	Random movement, pheromone driven	[20, 21]
	Hierarchical influence mobility	Nature-based model by using binary influence in different nodes	[22]
	Connectivity and coverage mobility	Ensure connectivity and coverage mainly for reconnaissance tasks	[23–27]
	Community mobility	Based on social relationships among individuals	[28]

divides the moving trajectory into five types with strong constraints on the random speed and direction. The random class mobility model is the most popular model for UAV movement, especially in simulation scenarios, but not suitable for FANET networks as UAVs cannot change their speed and direction rapidly and randomly in a practical scenario.

As for the reference class mobility model, the representative models [14–18] include the RPGM model, particle swarm mobility model, reference velocity mobility model, and freeway mobility model. All of these mobility models have features with individual logical reference points or group logical reference points. The RPGM model introduced the concept of logic reference points. A single node not only maintains its random behavior according to its logical reference point but also considers the group logical center while moving. When the number of nodes participating in the collaboration is large, the calculation is quite complex and time-consuming. Based on the RPGM model, the RVGM model uses the group movement speed and the local node's velocity to replace the logical reference point and the motion vector and can ensure that each node keeps the aggregation moving state all time. Particle swarm and freeway mobility models are also based on the reference point or reference direction, but with special consideration on particle swarm optimization (PSO) algorithm or freeway model algorithm to avoid the collision problem. Although the reference model has a certain similarity with the actual UAV cluster movement, it is a lack of consideration on the relationship of node tasks, especially associated with mission executions.

Representative models for the associated class include literature [20–28], such as the pheromone rejection model, hierarchical influence model, connected coverage model, and community model. The associate class mobility model mainly considers the connection between nodes to form a certain motion vector to realize high connectivity, wide coverage, or other targets. For example, the pheromone rejection model places the pheromone in the position when it has reached and makes the node choose movement paths according to the value of the pheromone around it to ensure the movement coverage. The community mobility model realizes node movement according to the relationship between individuals. The hierarchical influence mobility model mainly focuses on mobile nodes of various heterogeneous classes. The connected coverage mobility model is aimed at the connectivity and coverage area of the network by exchanging information such as location and speed between nodes. Although the associate class mobility models take the connection between nodes as the main attention aspect, it seldom considers the relevance of different tasks executed by different UAVs, especially in a practical scenario where the UAV swarm just performs a quite complicated mission cooperatively. Usually, the execution time of the mission should be the key indicators to evaluate the efficiency of different mobility models.

By summarizing the movement strategies of the above three categories of mobility models, it is clear that those models are continuously evolving towards the direction of random individual behavior, group behavior, and adoption of complex missions. Besides, the constraints of node

motion vector are further taking into account the features about the task requirements and network performances. The abovementioned mobility models can be simply and quickly deployed into the UAV swarm, but they cannot guarantee the real-time information interaction between UAV nodes during the process of complex mission execution. In this scenario, each task within the mission has complicated relationships and timing requirements with others. For example, the random class mobility model only reflects its movement of a single UAV without the consideration of the relationship among other nodes. For the reference class mobility model, the logical reference points and their accurate positions are difficult to obtain in the practical scenario. For the associated class mobility model, if the reference vector is not selected properly, it will cause a large number of message exchanges between nodes, and this kind of information not only wastes wireless channel bandwidth resources but also has a negative impact on the whole network performance.

Therefore, in a complex UAV swarm cooperative mission execution scenario, the mobility model not only should consider how these different UAVs move but should also pay more attention to the effectiveness of cooperative task execution, the relationships among different nodes, and the reliability of the whole network. To solve these problems, a new mobility model for UAV swarm is proposed, which considers the features of the multilayer task scheduling strategy.

The reference nodes and task relationships were combined together to design the multilevel reference node group mobility (ML-RNGM) model, for better real-time performance guarantee for task and mission and better network connectivity. For the inner real-time performance evaluation of a UAV, literature [34, 35] gave general schemas to solve it.

3. Task Scheduling Architecture for UAV Swarm

3.1. Multilayer Task Scheduling Model. In this paper, a multilayer task scheduling model for UAV swarm is constructed. It is an integrating model for avionics, especially when considering functions and resources are distributed in different UAVs to perform complex missions under limited UAV resources. In this architecture, the dynamic management and dispatch of swarm resources are realized by using the concept of resource virtualization in cloud computing. Compared with the traditional avionics scheduling strategy limited within a single platform, the avionics cloud scheduling model for UAV swarm emphasizes the network as the center, allocates the common service resources according to the mission and task requirements, makes full use of the advantages of network resource sharing and task cooperation among UAVs, and increases the completion rate and real-time performance of mission and task. To full use of the advantage of avionics cloud scheduling architecture, a corresponding multilayer task scheduling model [4], namely, the ML-TS model, is proposed and shown in Figure 1.

The multilayer task scheduling model mainly consists of three steps: step 1, resource reporting, which is shown by the purple dot-dash arrow in Figure 1. In this step, the UAV internal network controller reports its resource information to the corresponding cluster head, the cluster head reports its resource information to the central controller, and this kind of resource reporting just form a hierarchical UAV resource pool. In step 2, task allocation, as shown by the red arrow in Figure 1, the UAV swarm begins to assign tasks from the center controller to each cluster and then to each UAV. To improve the real-time performance of missions and tasks, the missions are divided into subtasks and finally distributed from the central controller to the internal end systems (ESs) within each UAV [5]. In step 3, a collaborative operation, each normal UAV begins to fly to its destination according to the assigned tasks, receives information from other UAVs, and performs the dispatched tasks.

In step 2, the most essential substep for task allocation is mission decomposition. System engineering methods such as the Zachman method and Functional Flow Block Diagram (FFDB) can be used to decompose these complex missions. The smallest granularity of mission decomposition is that the divided tasks can be performed by a single UAV. The mission decomposition can be further divided into the following four stages:

(1) Determine the mission group which should be performed by the UAV swarm

(2) FFDB is used to divide the mission into different task segments

(3) Distinguish the specific functional requirements of each task segment according to the overall requirements and analyze the functional components and resource requirements required by tasks

(4) Analyze the information exchanging requirements among different task segments and determine the relationships between task segments through information exchange

Generally, a mission group includes several interrelated missions, and these missions can be further divided into tasks with special sequential orders. Thus, the execution of these tasks by the node UAVs also should obey the inherent sequential order. Figure 2 gives the diagram of the mission decomposition architecture. A mission group is a set of missions, and these missions have logical relationships with each other and contain different tasks. To illustrate the relationships among different tasks, FFDB is used to describe them. In Figure 3, boxes are used to represent tasks, and the directional connections between boxes are used to show the relationships, for example, the predecessor tasks and the successor tasks.

In Figure 3, there are 5 mission groups and 16 tasks in total. Each mission group contains a different number of tasks. For example, mission 1 contains 4 tasks and has three predecessor tasks which are task 1, task 2, and task 3. When the execution of the three tasks is finished, task 4 can be executed. For missions, also there exist relationships between them. For example, mission 1 and mission 2 are the predecessor missions for mission 3, and the output of mission 1

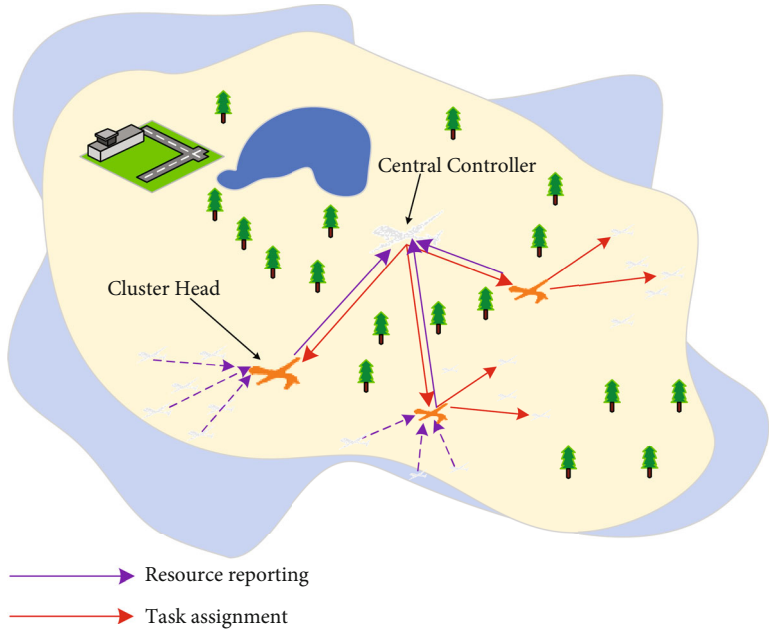

Figure 1: The multilayer task scheduling model of UAV swarm.

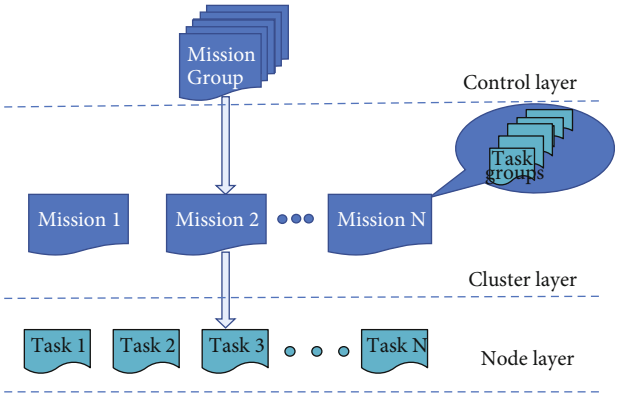

Figure 2: Mission decomposition architecture diagram.

and mission 2 can be seen as the input for mission 3. Typically, tasks should be executed in UAVs. As a different UAVs may contain a different number of processing resources, the numbers of tasks which can be executed in UAVs are different. Furthermore, each mission should be executed within a UAV cluster to guarantee the strict timing requirements for tight coupling tasks.

In the FFDB model, all missions and tasks can be ordered as a hierarchical tree according to the predecessor and successor relationships. For example, if a task A is one of the inputs of task B, then A is regarded as a child node of B, and B is regarded as the parent node of A. According to the principle, the final executed task can be seen as the root of the hierarchical tree, and the tasks without predecessors can be seen as the basic nodes. To evaluate the relevance degree between tasks, a variable H is introduced to show the level where the considered task locates. The calculation of the H value can be obtained according to the following steps: (1) establish a hierarchical tree according to the correlation between tasks; (2) find the longest path from the root node to the basic nodes, and the longest path should be seen as the maximal distance or the maximal level of the root node; (3) the basic nodes with the longest path apart from the root node should be seen as the first executing nodes, and the level of these nodes should be set as one; (4) the H level of the other nodes can be obtained according to the distance from the root node. A more edge apart from the root node, then the H level should minus one. For example, the root node is located at the highest level, and the level or H of the root node should be the longest path from the root node to one of the basic nodes. When the maximal H is figured out, the exact values of other nodes can be obtained according to the parent and child relationships. The lowest value of H is considered as one, which means the corresponding nodes are just the starting nodes and have the longest execution path to the root node. Figure 4 gives the general idea for the task correlation level tree. In addition, the level of each task is assigned according to the predecessor and successor relationships shown in Figure 4. With the assistance of H, the executing orders for these interrelated tasks can also be found. The larger the value H is, the later the task should be executed, vice versa. Besides, the level difference between two tasks also can be seen as the correlation degree for the two task executions. If the level difference is only one, then the two tasks must have a direct upstream or downstream relationship.

Table 2 summarizes the H value corresponding to each task correlation level shown in Figure 4. For this example, the minimum value of H is 1, and the maximum value is 6. The tasks with H value as 1 will be executed firstly, and the task with the maximal H value will be executed lastly. Tasks with higher H values should only be executed after tasks with lower H values are completed.

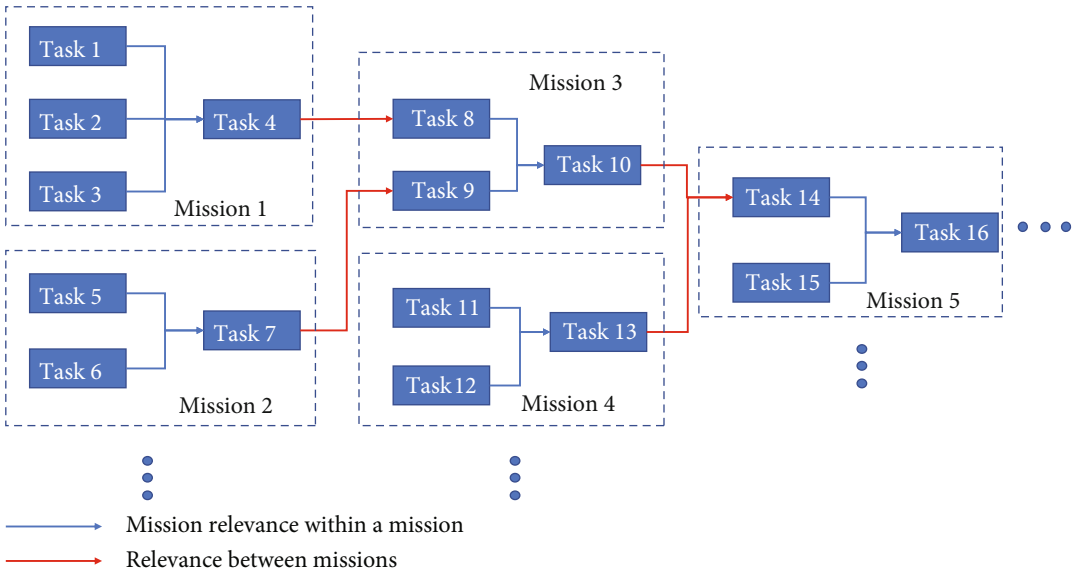

Figure 3: Associated task execution sequence flowchart.

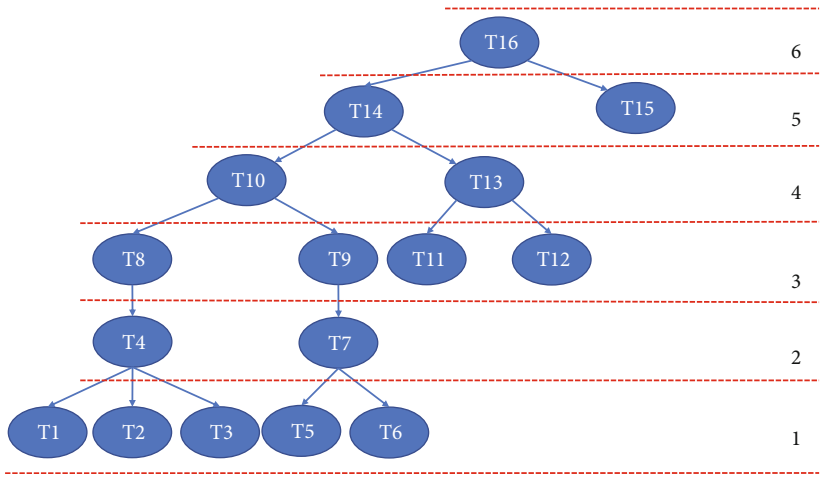

Figure 4: Task correlation level tree.

Table 2: H value of each task in Figure 4.

H value	Task code
1	T1, T2, T3, T5, T6
2	T4, T7
3	T8, T9, T11, T12
4	T10, T13
5	T14, T15
6	T16

After the mission decomposition is achieved, according to the FFDB model and H value, all missions and tasks should be dispatched into the UAV cluster network. The mapping strategy between the decomposition results and the network architecture is shown in Figure 5.

Within the control layer, the central controller performs the process of decomposing the mission group into missions. Typically, the central controller can adopt the greedy algorithm to achieve the optimal matching between the resources provided by UAV clusters and the requirements of missions. Then, each cluster is assigned several missions, and the information about the assigned missions is distributed into cluster head nodes. Within the cluster layer, the UAV cluster head node can adopt the genetic algorithm to further dispatch subtasks to the appropriate nodes to minimize the total execution time of missions carried out by different UAV clusters and nodes. Each individual UAV node should receive several assigned tasks within the node layer and execute these tasks according to its local scheduling strategy.

3.2. An Example of UAV Swarm Task Scheduling Architecture. Figure 6 gives an example of the task

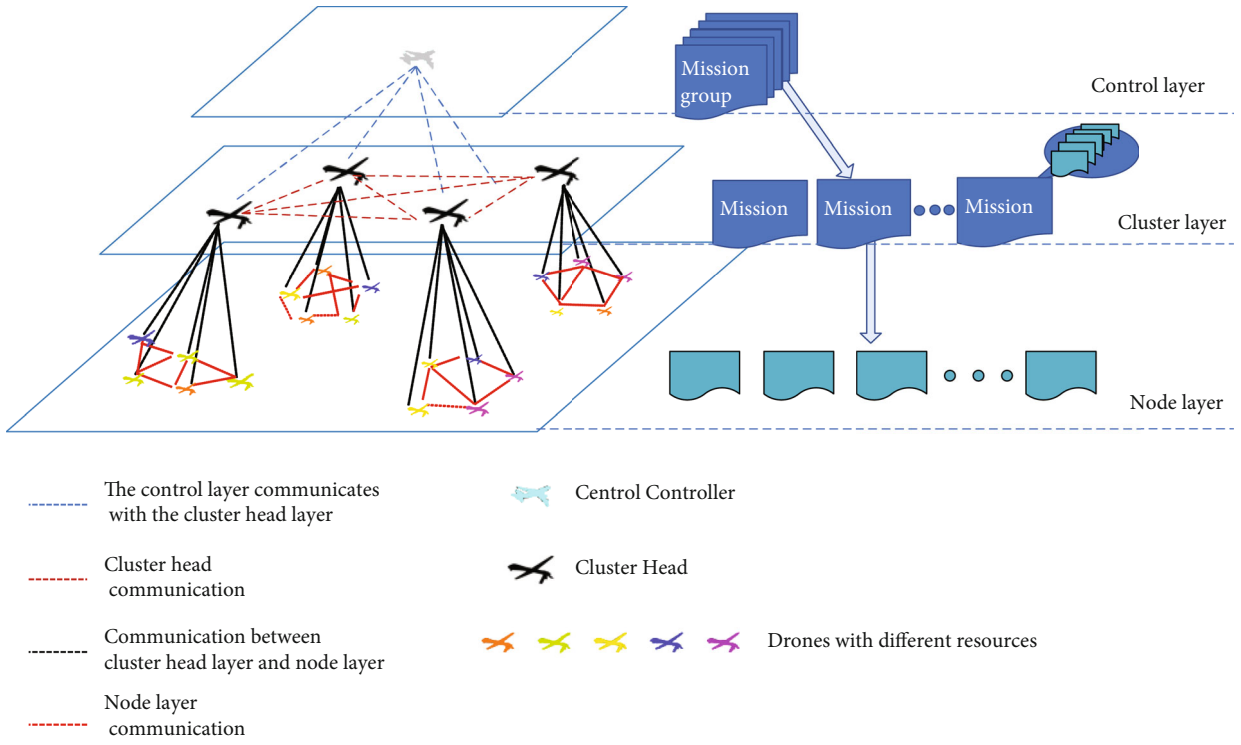

Figure 5: Mapping strategy of UAV cluster network architecture and mission task decomposition.

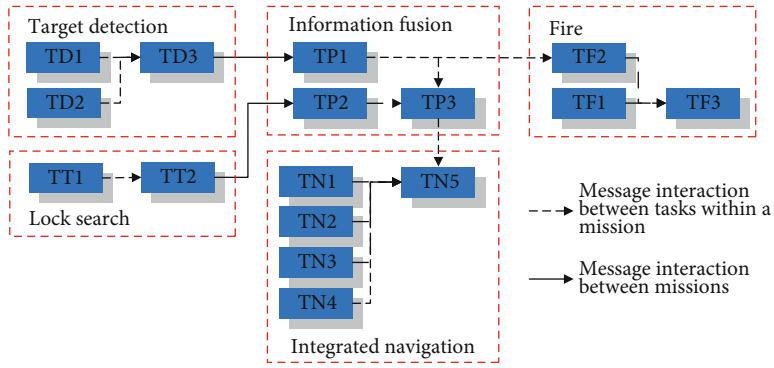

Figure 6: An example of the hierarchical task model: attack mission group.

scheduling architecture for the UAV swarm. In this example, it is discussed how to realize the task scheduling of a UAV swarm for an abstracted attack mission group. In a simplified scenario, the attack mission group is divided into five missions: target detection, lock search, information fusion, integrated navigation, and fire attack mission. Each mission can be further divided into different tasks according to functions, such as target detection including three tasks: TD1, TD2, and TD3. Tasks belonging to the same mission group will interact with each other by using message communication. If the previous task has not been completed, the subsequent tasks should be blocked and postponed. Taking the attack mission group, for example, the execution of TP1 depends on the completion of TD1, TD2, and TD3. If the execution of TD1 is postponed, TD3 and TP1 might not be implemented in time, and this consequence may not only seriously affect the real-time performance of a single task but also potentially affect the real-time performance of the whole mission group.

As shown in Figure 6, the attack mission group is taken as an example and the corresponding H value of each task is shown in Table 3. The minimum value of H is 1, and the maximum value is 5. For tasks TN5 and TF3, they are both the final executing tasks but with different relationship paths, so the maximal value H for the hierarchical mission group should be the largest value among the final executing tasks. For this example, TN5 and TF3 have the same H value as 5.

4. Multilevel Reference Node Group Mobility Model

As mentioned in Section 2, the movement strategy for UAVs that operate in the same mission group collaboratively is

Table 3: H value of each task in Figure 6.

H value	Task code
1	TD1, TD2, TT1
2	TD3, TT2, TN5
3	TP1, TP2
4	TF1, TF2, TP3, TN1, TN2, TN3, TN4
5	TF3, TN5

quite critical to ensure the real-time performance of the whole mission group. Different movement strategy has different influences on the completion of missions and tasks. The goal of this paper is to find an appropriate method to improve the task real-time capability and network connectivity of the UAV swarm. In this section, the enhanced UAV swarm mobility model will be described, which not only depends on the reference node movement strategy but also refers to the neighbor node's tasks to form a multilevel reference chain.

4.1. Reference Point Group Mobility Model. The reference point group mobility (RPGM) model is a basic group mobility model based on the reference point. Each node in the mobility group follows up with a logical center (or reference point) which determines the group's motion behavior. The nodes in the group usually are randomly distributed around the logical point. Different nodes can use their individual mobility models but should obey the guideline of the reference point. During the movement process, each node not only has its own individual random motion vector but also should be affected by the group logical center motion vector.

For RPGM model, the following parameters should be used. When a cluster network is constructed by UAV swarms, there should be a logical center in each cluster.

(1) t means time

(2) m_{ji} means node i in cluster j, and the index i indicates the node's index in the cluster

(3) $m_{ji}(t)$ is the position of node i in cluster j at time t, and $m_{ji}(t+1)$ means the position at the next moment

(4) $M_{ji}(t)$ is the random part of the moving vector for the node m_{ji} between t and $t+1$

(5) $W_j(t)$ is the motion vector of the cluster head in the cluster j at time t

(6) $V_{ji}(t)$ is the motion vector of m_{ji} at time t

For a cluster networking scenario, the cluster heads can be selected out as the reference points naturally, and all common nodes in a cluster should refer to the cluster head. The movement of the cluster head at a time t can be represented by a motion vector $W_j(t)$. It defines the motion of the cluster head itself and provides the general motion trend of the whole group. For each node, the random part of the motion vector $M_{ji}(t)$ is a random vector deviated by a group member from the reference point. Formally, the motion vector of the group member $V_{ji}(t)$ can be described as

$$V_{ji}(t) = W_j(t) + M_{ji}(t). \tag{1}$$

The vector $M_{ji}(t)$ usually is an independent identically distributed random process. Its length is randomly assigned in the interval $[0, r_{\max}]$ where $r_{\max}$ is the maximum allowed distance deviation, and the direction is randomly assigned in the interval $[0, 2\pi]$. Figure 7 gives the schematic diagram of the RPGM model.

In Figure 7, each node in the cluster moves according to the group cluster head and their respective individual motion vector for the next moment. For example, the motions of node m_{11} and node m_{22} should obey formulas (2) and (3). The vectors $W_1(t)$ and $W_2(t)$ represent the motion vectors of the cluster heads in group 1 and group 2, respectively, and nodes m_{13} and m_{21} are just the cluster heads for group 1 and group 2. Vectors $M_{11}(t)$ and $M_{22}(t)$ represent the random motions of nodes m_{11} and m_{22}.

$$V_{11}(t) = W_1(t) + M_{11}(t), \tag{2}$$

$$V_{22}(t) = W_2(t) + M_{22}(t). \tag{3}$$

Without any other constraints, the motion vector $V_{ji}(t)$ of nodes at the next moment is mainly determined by the node random motion vector $M_{ji}(t)$ and cluster reference point motion vector $W_j(t)$. The strategy of RPGM mobile is quite simple, but it has severe shortcomings. Firstly, the relevant parameters of the RPGM model need to be configured in advance [15, 16], such as the position coordinates of the logical reference points of each node and the group logical reference points of each cluster. This kind of advanced configuration belongs to off-line calculation and is not suitable for a real-time state with high dynamical changes. Secondly, this model does not consider the relevance between any collectively executed tasks; thus, it cannot satisfy the collaborative scenario with tight relationships, especially with the sequence orders among different missions and tasks. However, the concept of reference point provides a method to restrict the corresponding nodes into an acceptable scope to ensure UAV swarm network stability when these nodes are performing the related tasks. In the ML-RNGM model, the concept of reference point is also adopted but adds some changes to reflect the relations between tasks.

4.2. Multilevel Reference Node Group Mobility Model. To enhance the RPGM model to match the hierarchical task scheduling model, the multilevel reference node group mobility (ML-RNGM) model is proposed, in which the node dynamically selects the reference node to move according to the relationship between the tasks performed within the nodes. The ML-RNGM model mainly includes two parts: the basic model framework and the reference node selection algorithm. The basic model framework is

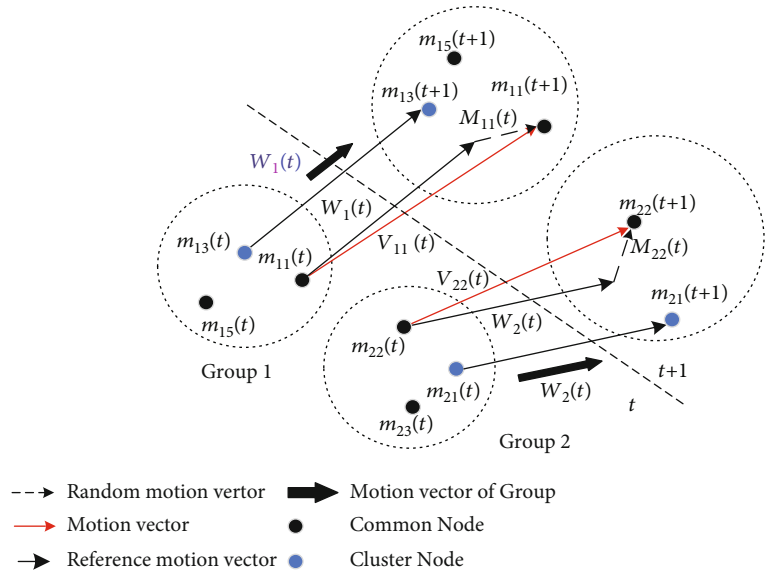

Figure 7: Schematic diagram of RPGM.

aimed at improving the RPGM model efficiency by changing the logical reference point to the actual physical reference nodes. The reference node selection algorithm primarily considers the impact of interaction information between related tasks, and it is based on the gravity model of spatial interaction [36, 37]. Thus, the proposed ML-RNGM model mainly consists of the following two strategies:

(1) *Dynamic Reference Node Selection*. The group logical reference points and the individual logical reference points in the RPGM model are replaced by the physical reference central controller and the physical reference nodes.

(2) *Multilevel Reference Chain*. The H value of the task is introduced to constrain the selection of reference nodes. The nodes executing the associated tasks can be formed into a multilevel reference link according to the interaction relationship between tasks.

The ML-RNGM model is designed to execute the collaborative tasks by UAV swarm, so the model's parameters mainly include two aspects. The first aspect contains the basic parameters needed by the RPGM model, and the second aspect mainly contains the parameters just reflecting the relationship between tasks and network architecture. Besides the symbols defined in Section 4.1, there are some other new parameters as follows.

(1) $W_{ji}(t)$ is the motion vector of the node m_{ji} from the reference node at time t

(2) $m_{cc}(t)$ is the location of the central controller m_{cc} in the UAV swarm at time t

(3) $V_{cc}(t)$ is the motion vector of the node m_{cc} at time t

(4) F_{ab}^{j} is the value of the associated gravity between node m_{ja} and node m_{jb}. F_{max} is used to represent the maximum associated gravity

(5) D_{ab} is the distance between node m_{ja} and node m_{jb}

(6) r is the radius of node communication

(7) H_i is the level of task i in the hierarchical task model

(8) T_{update} is the updating task of the reference node. Meanwhile, $T_{neigbor}$, T_a, and T_b are used in Algorithm 1 to represent different task sets

(9) N is the number of tasks performed by one node. For example, N_a and N_b represent the number of tasks performed in a node m_{ja} and m_{jb}, respectively

(10) $V_{ji}^{r}(t)$ is the motion vector of the reference node that is associated with m_{ji} between t and $t+1$

In a UAV swarm, different UAVs are divided into several clusters according to the ML-TS model, and each cluster just performs a mission. For simplicity, the movement of the central controller node can be determined in advance through some configured path points. The cluster head in a single cluster needs to refer the movement vector of the central controller and the movement vector of the task reference point. The normal node in a single cluster needs to consider the movement vector of the reference node and has its own random movement. For the basic model framework, it is stated in Section 4.2.1. For the second part, the reference node selection algorithm is introduced in Section 4.2.2.

4.2.1. Basic Model Description. The concept of the improved mobility model is shown in Figure 8.

Input: Mobility model parameter: α, β,
Neighbour node set: $M = \{m_{j1}, m_{j2}, m_{j3}, \cdots m_{jK}\}$,
Task set performed by node m_{ja} : T_a;
Output: Optimal selection: m_{best}
1. initialize: $F_{\max} = 0$, $F_{at}^{j} = 0$, $T_{\text{neigbor}} = \Phi$
2. for $t = 1, 2, 3 \cdots, K$ do
3. $b = M[t]$
4. $T_{\text{neigbor}} = \text{Get\_Data\_From\_Neigbor}(b)$
5. $D_{ab} = \text{TASK\_TO\_}H(T_a, T_{\text{neigbor}})$
6. $F_{at}^{j} = \alpha(w_{ja}\, w_{jb})/(D_{ab} + 1)^{\beta}$
7. if $F_{at}^{j} \geq F_{\max}$ then
8. Saves the current maximum associated gravity: $F_{\max} = F_t$
9. Saves the current optimal selection of reference node: $m_{\text{best}} = b$
10. end if
11. end for
12. $m_{\text{best}} \longrightarrow$ Optimal selection of reference node.

Algorithm 1: Reference node selection for node $m_{ja.}$

Figure 8 is a schematic diagram for the ML-RNGM model. The figure just illustrates the movement of nodes in two clusters from time t to time $t + 1$. According to the networking topology, there are two cluster heads and one central controller which is selected out as the physical reference central controller. Different symbols represent different node types, such as round for the normal node, square for the cluster node, and pentangle for the central controller node. Additionally, different colors and styles of arrows stand for different relationships, such as the red arrow shows the motion vector of nodes, the black arrow is the motion vector of the central controller, the green dotted line, and the solid line shows the motion vector of the central controller, and the purple dotted line shows the reference node motion vector.

For the cluster head, it needs to consider the motion vector of the central controller and the motion vector of the reference node. For example, the node that performs the final output task in a mission can be selected out as the reference node. In Figure 8, the motion vector of the central controller is $V_{\text{cc}}(t)$. Nodes m_{11} and m_{21} are the two cluster heads in the clusters G_1 and G_2, respectively. Similar to the RPGM model, at time t, the reference motion vectors $W_{11}(t)$ and $W_{21}(t)$ of the two cluster heads should obey the motion vector $V_{\text{cc}}(t)$ of the center controller, which is represented by the green dotted line and the motion vector $V_{ji}^{r}(t)$ of the reference nodewhich is represented by the purple dotted line.

$$W_{11}(t) = V_{\text{cc}}(t) + V_{11}^{r}(t), \tag{4}$$

$$W_{21}(t) = V_{\text{cc}}(t) + V_{21}^{r}(t). \tag{5}$$

Combining formulas (4) and (5) with formulas (2) and (3), the node motion vectors for m_{11} and m_{21} should obey:

$$V_{11}(t) = W_{11}(t) + M_{11}(t) = V_{\text{cc}}(t) + V_{11}^{r}(t) + M_{11}(t), \tag{6}$$

$$V_{21}(t) = W_{21}(t) + M_{21}(t) = V_{\text{cc}}(t) + V_{21}^{r}(t) + M_{21}(t). \tag{7}$$

Thus, the movement vector of the cluster head is determined by the movement vector $V_{\text{cc}}(t)$ of the central controller, the movement vector $V_{ji}^{r}(t)$ of the reference node, and the random moving vector $M_{ji}(t)$ of the node m_{ji}.

For other ordinary nodes, their motion vector, random motion vector, and reference motion vector have the same relationships. The reference nodes are determined by the reference node selection algorithm, which will be described in Section 4.2.2. The node m_{15} is shown in Figure 8 as an example. The reference node for m_{15} is node m_{11}. $m_{15}(t + 1)$ is the expected position of the node m_{15} when considering the random factor. The motion vectors have the following relationships:

$$V_{15}(t) = W_{15}(t) + M_{15}(t). \tag{8}$$

In formula (8), $M_{15}(t)$ represents the random motion vector as the random behavior of individual nodes in the execution of tasks, $W_{15}(t)$ is the motion vector of the reference node, which is determined by $m_{15}(t)$ and $m_{11}(t)$. $V_{15}(t)$ is the final motion vector at time t, which can be expressed as follows:

$$V_{15}(t) = V_{15}^{r}(t) + M_{15}(t). \tag{9}$$

In formulas (6), (7), and (9), the parameters $V_{11}^{r}(t)$ V_{21}^{r} and $V_{15}^{r}(t)$ are all related to the reference nodes. Since a suitable reference node can effectively improve the efficiency of the UAV cluster cooperative execution of tasks, the reference node selection algorithm will be introduced in the next section.

4.2.2. Reference Node Selection Algorithm. The ML-RNGM model is specially designed for the Multi-Layer Task

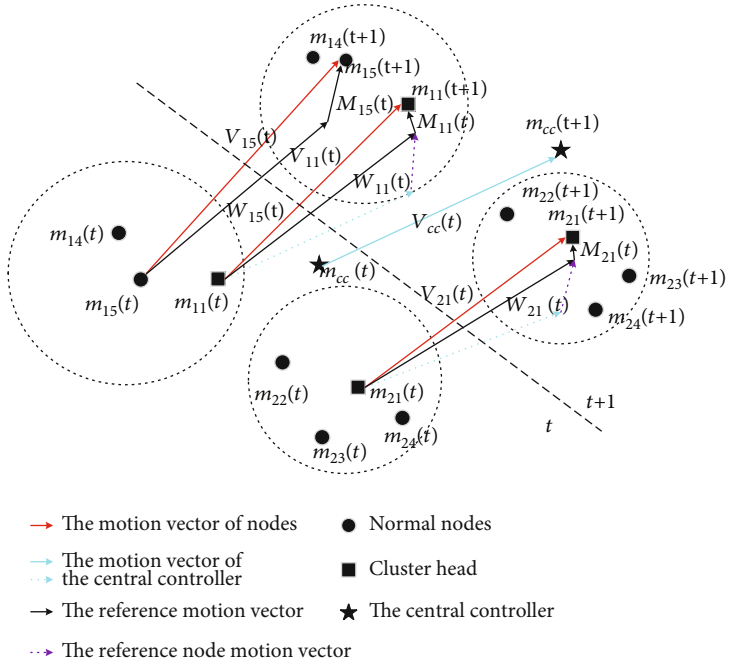

Figure 8: Schematic diagram of the ML-RNGM model.

Scheduling (ML-TS) model, which is shown in Figure 3. Considering the interaction relationship between different UAVs, the information transmission is added among UAVs when performing the associated tasks. However, these kinds of information exchanging would increase the finish time for mission and task execution. To reduce the time overhead caused by the distance and message scheduling of UAV nodes when performing the associated tasks, the gravity model is introduced to select the best appropriate reference point during the node movement process. For the gravity model, three aspects to describe the interaction process are put forward: the associated gravity, the associated distance, and the level H of each task. The three aspects are used to quantify the impact of information interaction between nodes. The values of the three aspects are calculated with formula (10), formula (11), and Table 3, respectively. In the process of node movement, nodes will suffer the associated gravity from different neighbor nodes, as shown in Figure 9.

For example, the node m_{15} is taken to illustrate the gravity model, and the dashed circle in Figure 9 is the sensing range of m_{15}, which usually is equal to the cover scope of the cluster G_1. Node m_{15} is subject to the associated gravity from different neighbor nodes. Its value is related to the number of tasks executed by any two corresponding nodes and the associated distance of the task, which can be defined as follows:

$$F_{ab}^{j} = \alpha \frac{\omega_{ja}\omega_{jb}}{(D_{ab}+1)^{\beta}}. \tag{10}$$

In formula (10), α and β are two adjustable parameters, which can be configured according to the actual number of

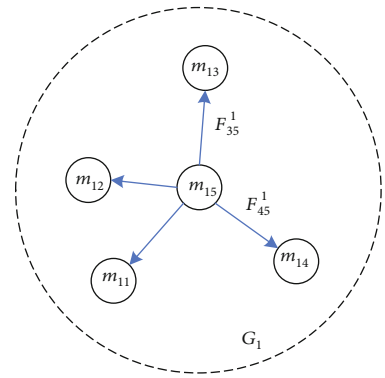

Figure 9: Illustration of the gravity model.

nodes and missions. w_{ja} and w_{jb}, respectively, represent the number of tasks in nodes m_{ja} and m_{jb}. D_{ab} represents the associated distance between the two nodes, and the parameter $D_{ab}+1$ is used to prevent the denominator from zero. D_{ab} can be expressed as follows:

$$D_{ab} = \left| \sum_{i=1}^{N_a} H_i - \sum_{j=1}^{N_b} H_j \right|. \tag{11}$$

In formula (11), the difference between H values of the two tasks is used to represent the associated logical distance, where H_i represents the level of task i in the task correlation level tree. The closer the H values of the two tasks are, the stronger the relationship between the two tasks would be. Ultimately, there would be a lower D_{ab} and higher F_{ab}^{j} between the UAVs which are performing the associated tasks.

In formula (10), F_{ab}^{j} gives the associated gravity of two nodes. For the cluster head or the reference node, the value F_{ab}^{j} is calculated to assist in assigning a reference displacement vector to the node. Besides, the selection process of the reference node is carried out by a periodic task with an updating cycle T_{update}. The periodic selection task will be executed in every node, responsible for calculating the associated gravity between the considered node and the neighboring nodes. After that, the neighboring node with the maximum gravity value will be selected out as the reference node. The process is shown in Algorithm 1.

In Algorithm 1, the parameter K represents the number of neighboring nodes for node m_{ja}, M represents the set of neighboring nodes, and T_a represents the set of tasks performed by node m_{ja}. Besides, the functions Get_Data_From_Neighbor and TASK_TO_H are two critical operations in Algorithm 1. Get_Data_From_Neighbor(b) is responsible for sending messages to the neighbor node b and returning T_b, and T_b is the set of tasks performed within m_{jb}. The function TASK_TO_H calculates the associated distance D_{ab} between two nodes according to formula (8). Finally, the output m_{best} represents the optimal reference node m_{ja}.

Combined with Figure 8, Algorithm 1 is deployed on each node in clusters. Taking G_1 in Figure 8 as an example, the normal nodes m_{14} and m_{15}, respectively, select m_{15} and m_{11} as the reference nodes according to the reference node selection algorithm. For the cluster head nodes, their reference nodes include the central controller and the nodes that perform the following associated tasks in other clusters. And these nodes can be obtained by using Algorithm 1. For the cluster head m_{11}, it needs to consider the central control and the associated nodes in another cluster.

4.3. Performance Indicators. To evaluate the performance of the ML-RNGM model with other typical mobility models, three performance indicators are constructed, including the average end-to-end delay, network connectivity rate, and the average link duration.

To measure the three indicators, $X_j(a, b, t)$ represents the link state between the nodes $m_{ja}(t)$ and $m_{jb}(t)$ at time t. If the two nodes are connected, its value is 1; otherwise, it is 0. To calculate the link duration, $C_j(a, b, t)$ is used to indicate the change of $X_j(a, b, t)$. If $X_j(a, b, t-1) = 0$ and $X_j(a, b, t) = 1$, then $C_j(a, b, t) = 1$. Subsequently, to define the network connectivity rate and the average link duration, the symbol N_c is used to represent the number of node pairs that can communicate with each other, and N_a is used to count the number of all nodes in the network.

(1) Average end to end delay

The average end-to-end delay represents the performance of information transmission in a UAV swarm with the given mobility model strategy. Usually, end-to-end delays can be used to evaluate the real-time performance of tasks. A low end-to-end delay shows that the movement strategy has a positive impact on the real-time performance.

(2) Network connectivity rate

The connectivity rate of a network is represented by the symbol p, which shows the probability of a pairwise connection between two nodes in the network. The definition of connectivity rate in literature [17] is referred to, and this indicator is defined as the ratio of the connectable pairs of nodes to the total pairs of nodes in the network, which is shown in formula (12). Here, the symbol I_c is used to stand for the number of connectable pairs, and the symbol I_a represents the total pairs in the network.

$$p = \frac{I_c}{I_a(I_a - 1)/2} = \frac{2I_c}{I_a(I_a - 1)}. \tag{12}$$

(3) Average link duration

The link duration stands for the impact of movement strategy on message transmission routing performance. The longer the link duration is, the lower the link change rate would be. Therefore, longer link duration can result in better link stability and lower routing cost.

The measure of the link duration is mainly based on the definition of link change time (LC) and link duration (LD) [17] as follows:

$$\text{LC}_j(a, b) = \sum_{t=1}^{R} C_j(a, b, t). \tag{13}$$

In formula (13), the symbol R represents the current time, and the symbol $\text{LC}_j(a, b)$ represents the link change time between nodes a and b in the cluster G_j. Therefore, the link duration of the cluster G_j can be obtained as follows:

$$\text{LD}_j(a, b) = \begin{cases} \dfrac{\sum_{t=1}^{R} X_j(a, b, t)}{\text{LC}_j(a, b)}, & \text{if } \text{LC}_j(a, b) \neq 0, \\ \sum_{t=1}^{R} X_j(a, b, t), & \text{otherwise.} \end{cases} \tag{14}$$

Thus, in the cluster G_j, the ratio of the sum of $\text{LD}_j(a, b)$ to the value of I_c is the average link duration $\overline{\text{LD}_j}$:

$$\overline{\text{LD}_j} = \frac{\sum_{a=1}^{N_j} \sum_{b=a+1}^{N_j} \text{LD}_j(a, b)}{I_c}. \tag{15}$$

In addition to the above three indicators, other performance factors are also considered, such as the algorithm overhead, the number of neighbor nodes, and the link length. Besides, the makespan of the individual tasks and the whole mission group for the proposed model combined

with the multilayer task scheduling (ML-TS) model will be evaluated.

5. Simulation Model and Case Study

5.1. Simulation Model. In this section, the simulation model and the cases will be introduced to evaluate the performance of the ML-RNGM model. For better illustration, the comparison experiments for the ML-RNGM model will be done with the random class, the reference class, and the associated class mobility models. OMNeT++ is used as the simulation platform for these mobility models. Besides, the simulation parameters are divided into the scenario simulation parameters, the mobility model class parameters, and the mobility model parameters. The scenario simulation parameters remain unchanged in the whole comparison experiments, which are shown in Table 4.

Table 4 gives the general scenario simulation parameters for a default attack mission, and the relationship between different tasks is shown in Figure 5. The process of the multilayer task dispatch is mainly carried out in the central controller. The central controller sends the detailed allocation results to each cluster and then sends them to each UAV node as shown in Figure 2.

For the parameters related to the mobility models, there is no message transmission between nodes for the random class and the reference class mobility models, so the configuration and deployment are relatively simple. For the associate class mobility model, there is a dynamic selection of the reference nodes through message transmission. In the simulation model, the reference node selection obeys a randomly assigned strategy. The mobility model class parameters are shown in Table 5.

Table 5 gives the detailed information for different mobile models. For the ML-RNGM model, set $\alpha = 1$ and $\beta = 2$ to simplify the gravity model, and it is mainly referred to as the universal gravitation.

To analyze the performance difference between the mobility model and the other three type mobility models, three experiments were designed as follows:

5.1.1. Experiment 1: Basic Performance Comparison. The primary performance experiment includes two test cases, and the difference lies in the number of UAV nodes and the velocity of nodes. The two cases will statistic the average end-to-end delay, the network connectivity, and the average link duration through the simulation method.

For the first case, the number of nodes is fixed to 16 in each UAV cluster, and the velocity of nodes will be set to 10 m/s, 20 m/s, 30 m/s, 40 m/s, and 50 m/s, respectively. For the second case, the velocity of nodes will be held at 20 m/s. To ensure that each drone will have a task execution, the number of UAVs in clusters will set to the same value as the number of mission tasks. The corresponding parameter settings of the number of UAVs and the number of tasks are shown in Table 6.

In this case, there will be 5 clusters in the UAV swarm. The mission group is set to 5 to observe the influence of different UAV numbers on the average end-to-end delay, the network connectivity rate, and the average link duration, respectively.

Table 4: Scenario simulation parameters.

Simulation parameter	Value
Area size (km^2)	20×20
Data rate (Mb/s)	10
Message kind	BE (best effort)
Message size (byte)	2048
Navigational angle (°)	(0, 360)
Wireless range (m)	500
Routing protocol	AODV
Mac protocols	IEEE 802.11
Collaborative task scenario	Attack mission
Mission group	1
Task number	16
Simulation time	100 s

5.1.2. Experiment 2: Movement Trajectory Comparison. In this experiment, the node number within each cluster is set to 16, and then, the node positions will be counted at the time of 5, 10, 15, and 20 minutes, respectively, to analyze the node distribution for each mobility model.

5.1.3. Experiment 3: Real-Time Comparison under Avionics Cloud. In this experiment, the four mobility models mentioned above will be deployed to the UAV swarm successively. The node number within each cluster is set to 10, and the number of mission groups is set to 5. Experiment 3 also includes two test cases. For the first case, the mean and maximum values of five kinds of data are statistics to analyze the influence of different reference node selection algorithms on the UAV swarm. These statistical data include the algorithm overhead, the number of neighbor nodes, node load, and link length. For the second case, five UAV clusters are assigned within the same attack mission group as shown in Figure 5. Different mobility models will be deployed for the five clusters to calculate the makespan of the individual tasks and the whole mission group for the attack mission.

5.2. Simulation Analysis. The of three experiments are collected and analyzed. In this section, these results will be shown. The statistical data of the average end-to-end delay, network connectivity, and the average link duration in experiment 1 are shown in Figures 9 and 10. The position distribution of nodes and clusters with various mobility models in experiment 2 is shown in Figure 11. In experiment 3, the various mobility models are applied into the multilayer task scheduling model for the UAV swarm, and the relevant simulation results are shown in Tables 7 and 8.

5.2.1. Basic Performance Comparison

(1) The Influence of Node Velocity on Different Mobility Models. Figure 10 shows the changing trends of the basic performance indicators with the increase of node velocity

Table 5: The mobility model class parameters.

Mobility model class	The specified mobility model	Parameters	Value
The random class	Gauss-Markov mobility [8]	α	0.5
		θ	(0, 360)
The associated class	Hierarchical influence mobility [21]	NaN	NaN
The reference class	Reference point group mobility [14]	θ	(0, 360)
The proposed model	The multilevel reference node group mobility	α	1
		β	1
		θ	(0, 360)

Table 6: The number of UAVs and task parameters.

UAV numbers	Mission group	Task numbers
5	5	5
10	5	10
15	5	15
20	5	20
25	5	25
30	5	30

for the four mobility models. In all subgraphs, the blue color line specifies the results of the random class model, the yellow and green color lines denote the results of the reference and associated class models, and the last red color line shows the statistical data of the proposed ML-RNGM model. According to the four curves in Figure 10, the average end-to-end delay will increase and the network connectivity rate and link duration will decrease when the node velocity increases from 10 m/s to 50 m/s. These kinds of changing trends are also consistent with the results in literature [17]. Besides, the above three subgraphs just show the random class mobility model has the worst performance in each indicator among the four mobility models. For the reference and associated class models, they perform much better than the random class. However, the proposed mobility model is always the best one than the reference class mobility model and the associated class mobility model. The indicates just show that the ML-RNGM model can increase the real-time performance of tasks by 21%, the network connectivity rate by 15%, and the average link duration by 21% compared with the best of the other three models when the number of nodes in UAV swarm is set as 16.

(2) The Influence of Cluster Number on Different Mobility Models. Figure 11 shows the changing trends of the basic performance indicators with the increase of node number within each cluster. The different color curves represent the same models as Figure 10. Considering the comprehensive performance evaluation combined with the three basic performance indicators, the random class mobility model significantly is lower than the other three mobility models. Except for the random class mobility model, the performance trend of the other three mobility models is roughly similar: the average end-to-end delay decreases slowly with the increase of the number of nodes, and the network connectivity rate and the average link duration also show an upward trend with the increase of the number of nodes. Not surprisingly, the proposed ML-RNGM model still has the lowest average end-to-end delay and the highest network connectivity rate and the average link duration than the reference class and the associate class mobility models. Based on the statistical analysis, the ML-RNGM model can increase the real-time performance of tasks by 17%, the network connectivity rate by 12%, and the average link duration by 24% compared with the best of the other three models when the velocities of all nodes are holding at 20 m/s.

5.2.2. Movement Trajectory Comparison. In the ML-RNGM model, the relationship between the UAVs is taken into account which are performing the associated tasks. Compared with the RPGM model, the real-time information transmission ability between tasks is greatly improved in the ML-RNGM model. At the same time, due to the existence of the central node and multilevel reference points in the model, the concentration degree of UAV members is increased, which also will be conducive to information sharing and collaborative operation for the UAV swarm. The experimental results are shown in Figure 11.

Figures 12(a)–12(d) show the results of the random class mobility model, Figures 12(e)–12(h) show for the reference class mobility model, Figures 12(i)–12(l) show for the associated class mobility model, and Figures 12(m)–12(p) show for the proposed model. In all subgraphs, the different color squares show the different UAV clusters.

In Figures 12(a)–12(d), the movement results of the random class mobility model are shown from $t = 5$ minutes to $t = 20$ minutes. It is clear that the random class mobility model tends to be out-of-order since the distance between nodes keeps increasing even though they are in the same cluster. If the distance exceeds the maximum available range for communication between nodes, the information cannot be exchanged between nodes. Focused on the reference mobility model and the associated class mobility models, according to Figures 12(e)–12(h) and 12(i)–12(l), both two models can greatly constrain the cluster nodes along with the present state and the clusters can hold a relatively concentrated trend. Aimed at the movement trajectories, the two mobility models have no apparent difference. Figures 12(m)–12(p) show the simulation results of the

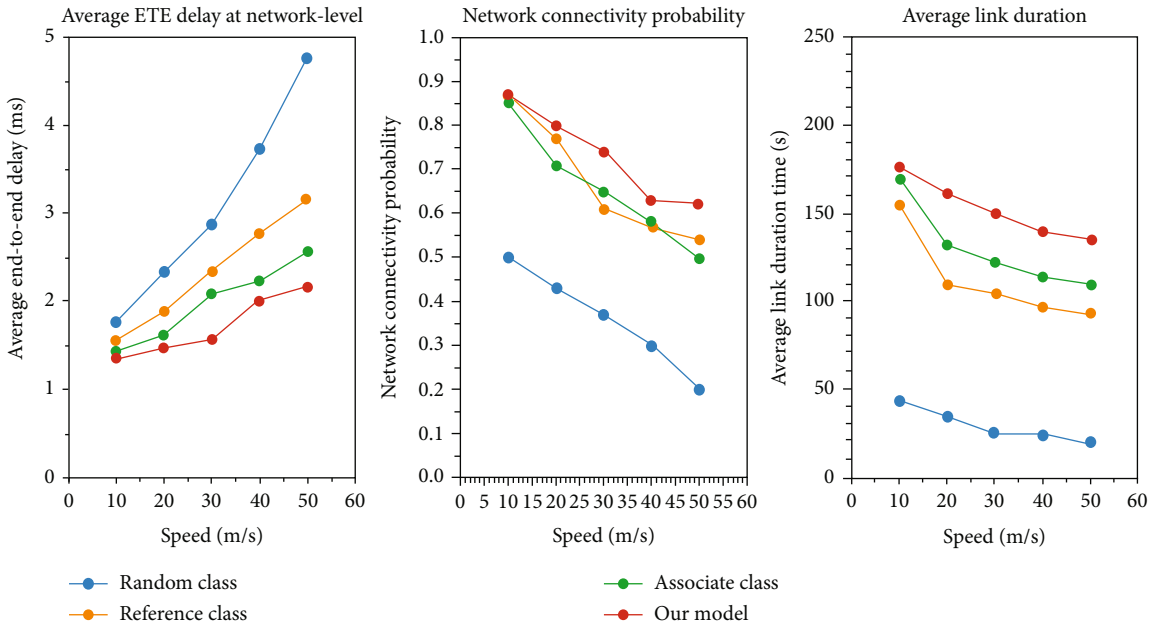

Figure 10: Basic performance evaluation with different velocities.

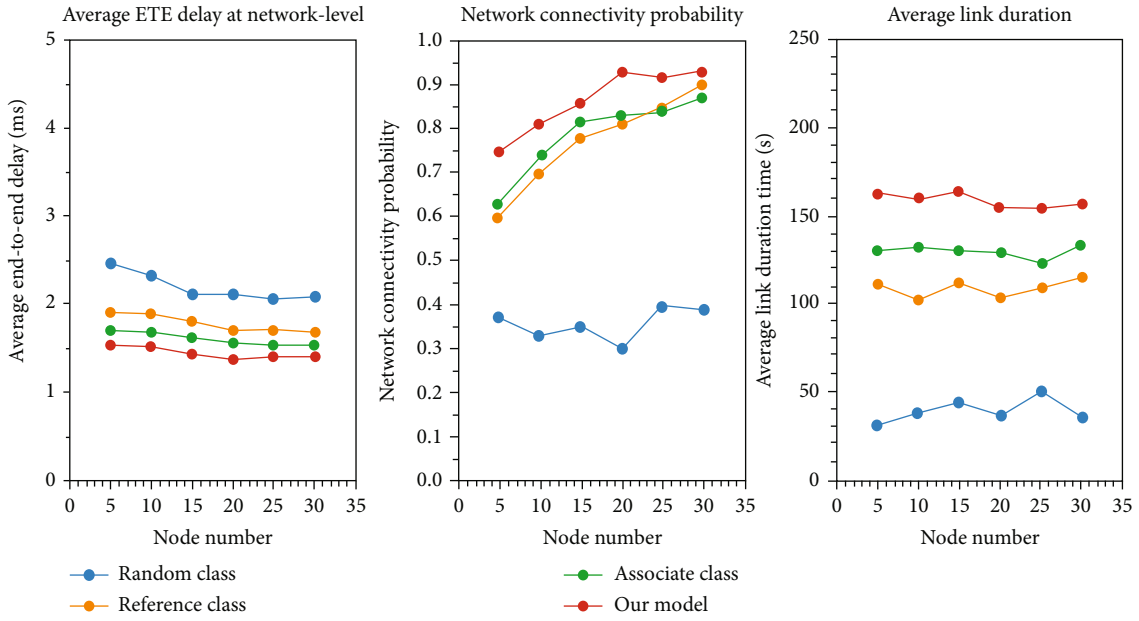

Figure 11: Basic performance evaluation with different node numbers.

Table 7: The performance of different reference node selection algorithms.

Statistical parameters	Random class		Reference class		Associate class		The proposed model	
	Mean	Maximum	Mean	Maximum	Mean	Maximum	Mean	Maximum
Algorithm overhead (packet/s)	0	0	0	0	1.36	2.2	1.8	2.6
Neighbor nodes (node)	0.4	2	2.4	4	2.6	5	3.3	6
Message average ETE delay (ms)	NaN	NaN	1.54	1.8	1.47	1.77	1.33	1.55
Link length (node)	1.3	2	2	2	2.3	4	3.5	5
Node load (node)	0	0	1.8	3	2.2	4	1.3	4

Table 8: Average makespans of individual tasks and the whole mission group with different moving strategies (unit: ms).

Mission group	Random class model		Reference class model		Associate class model		The proposed model	
	Average makespan of individual tasks	Average makespan of the whole mission group	Average makespan of individual tasks	Average makespan of the whole mission group	Average makespan of individual tasks	Average makespan of the whole mission group	Average makespan of individual tasks	Average makespan of the whole mission group
G1	NaN	NaN	4.66	74.56	4.41	70.56	2.65	40.96
G2	NaN	NaN	4.13	66.08	4.35	69.60	2.54	40.64
G3	NaN	NaN	4.51	72.16	4.55	72.80	2.77	44.32
G4	NaN	NaN	5.04	80.64	4.41	70.56	2.66	42.56
G5	NaN	NaN	4.51	72.16	4.32	69.12	2.43	38.88
Average makespan of the same mobility model	NaN	NaN	4.57	73.12	4.41	70.53	2.61	41.47

model. Compared to the subgraphs of Figures 12(e)–12(l), the proposed model always holds a high concentration state within each cluster, and different clusters keep relatively tight grouping features.

5.2.3. Stability of Mobility Model Comparison

(1) Reference Node Selection Algorithm. In the experiments mentioned above, several basic performance indicators about networking are considered. The difference between the proposed ML-RNGM model and other existing mobility models is whether to use an appropriate reference node migration strategy and how to choose the available reference nodes. For example, the random class mobility model does not use the strategy of reference point to guide the following movement. Both the reference class mobility model and the associated class mobility model have the strategy of reference point and adopt the fixed or random methods to select the reference points, respectively. When encountering the complicated collaborative scenario with hierarchically dispatched missions and tasks, these two move strategies appear hardly to handle these situations. The proposed mobility model selects the reference node by the H value of tasks to shorten the makespans both for the overall mission group and for individual tasks. Table 7 gives the performance comparison of the four reference node selection algorithms. The results are obtained according to 10 times independent simulations and represent the average performances. In Table 7, the algorithm overhead is evaluated according to the number of messages generated by the reference node selection algorithm. The value of neighbor nodes represents the number of nodes that maintain one-hop connection with the considering node at the current moment. The message end-to-end delay mainly indicates the delay of message transmission between two nodes. The link length stands for the number of nodes connected to the considered UAV in the process of movement. Finally, the node load is to show the number of nodes that have the same reference node. For a node, the neighbor node is usually different from the value of the node load, but the neighbor node can provide a prediction value for the node load. Besides, according to the definitions of the above indicators, smaller values of algorithm overhead, message average ETE delay, and node load would be better, thus the overall performance of the mobility model would be nice. If the values of neighbor node and link length are large, also the algorithm performance is fine. The experimental results are shown in Table 7.

Table 7 lists the mean and maximum values of the observed data in the simulation process. The random class mobility model and reference class mobility model do not need message transmission to select the reference points, so their algorithm overheads are zero. On the other side, the associated class mobility model and the proposed model need the operation of selecting the reference nodes through message transmission among nodes, so the algorithm overheads of the two models are nonzero. Generally speaking, the model has a higher overhead than the associated class mobility model. Besides, the algorithm overhead is positively correlated with the number of neighbor nodes. If the number of neighbor nodes is relatively large, then the network connectivity rate would be high potentially, which means the algorithm overhead of selecting the reference node from the neighbor nodes will increase due to more communication possibility. In the row of the average end-to-end delay of messages, the random movement strategy for the random class mobility model usually causes a long distance between nodes even beyond the available communication range and then fails the message transmission. Thus, the end-to-end delay for the random class mobility model is not a stable value, and NaN is used to illustrate it. On the whole, the comparative advantages of the ML-RNGM model with the other three mobility models are similar as the trends shown in Figure 10. For the average end-to-end delay or the maximum delay, the proposed model has obvious advantages in real-time performance guarantee.

In Table 7, the results of the link length and the node load are displayed. These two indicators are not only related to the reference point selection strategy but also related to the structure of the mission group. For example, when the

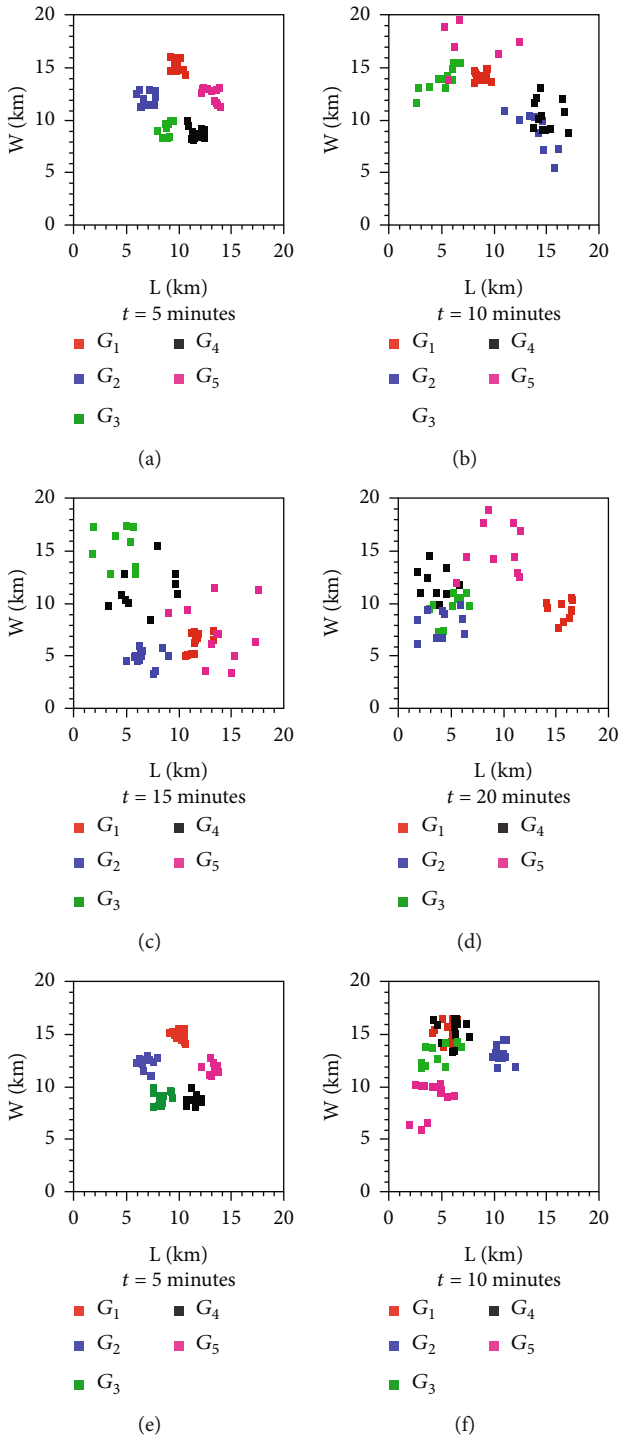

Figure 12: Continued.

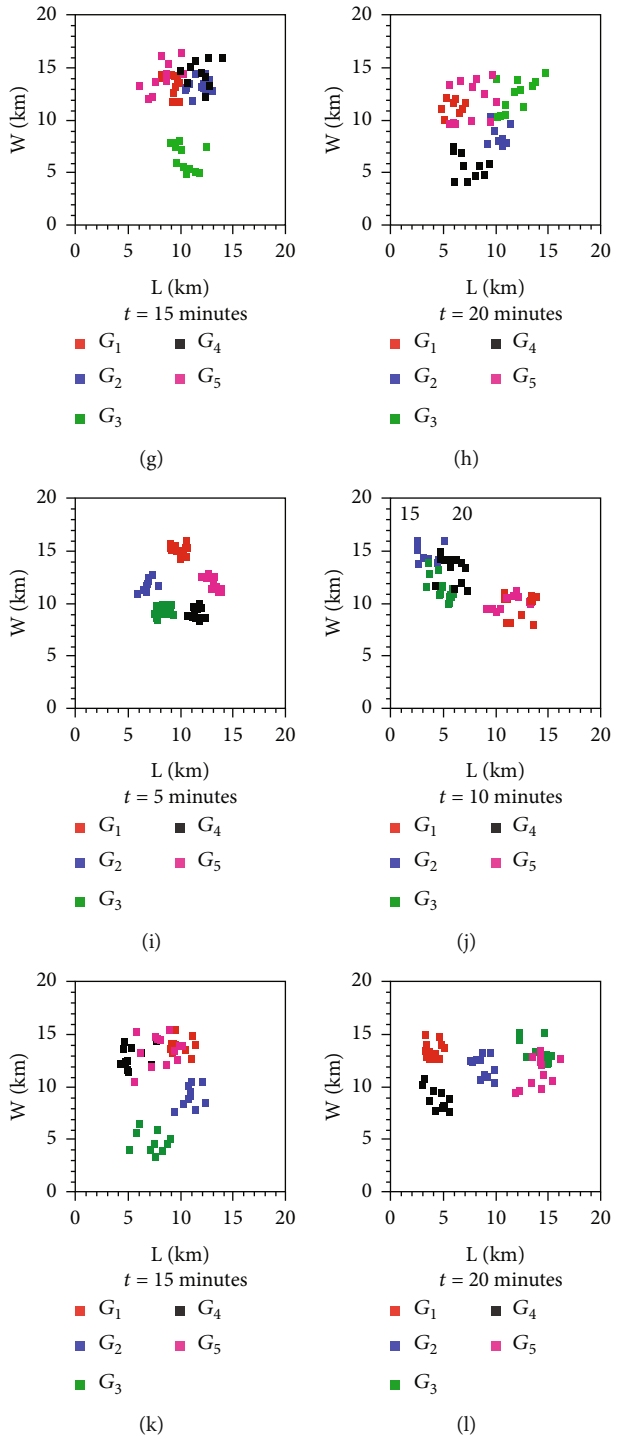

Figure 12: Continued.

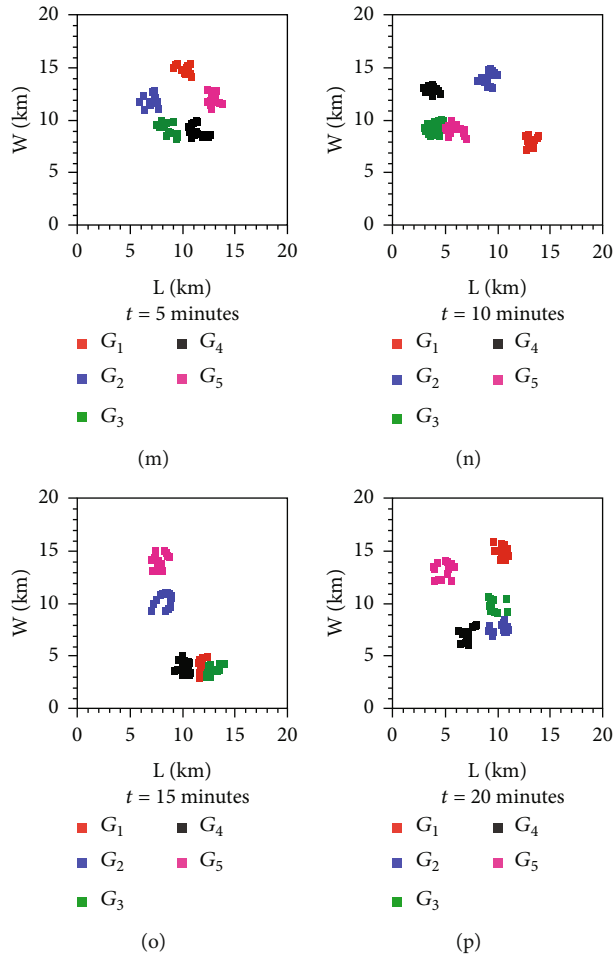

Figure 12: Trajectory comparison for the four mobility models from $t = 5$ minutes to $t = 20$ minutes.

sixteen tasks shown in Figure 5 are dispatched to different UAVs based on the ML-RNGM model by using the task allocation method described in Section 3, the UAV that executes the task TN5 has the maximum load as 4. But the maximal link length, like TD1➔TF3 or TT1➔TN5, could be 5. Although the model might have a larger algorithm overhead and a larger load of individual nodes, the other three mobility models have no obvious advantages for the average node load. For example, the average value of node load for the ML-RNGM model is smaller than the reference class mobility model and the associated class mobility model, which shows that the model can automatically balance the reference node selection. If more nodes choose the same node as the reference node, the load of the selected reference node might be too large, the network connectivity rate could drop sharply, and the task execution ability of the whole UAV cluster will reduce greatly. Furthermore, the average link length of the ML-RNGM model is also clearly larger than other models.

(2) Applications of Multilayer Task Scheduling Model for Avionics Clouds. In this section, to verify that the proposed mobile model can effectively shorten the execution time for the UAVs to perform complex missions collaboratively, these four mobility models are applied to the avionics cloud environment. Besides the basic performance indicators, it needs some new indexes, such as the makespan of the mission group and individual tasks, to achieve a complete comparison for the four mobility models. In literature [5], it also discusses the problem of the makespan. The definition of the makespan for individual tasks or the whole mission group is the sum of the completion time of its previous task and the communication overhead between nodes. However, in literature [5], the communication overhead is just set as a constant when calculating the makespan value. The makespan of individual tasks can be computed by formula (16).

$$t^{T_n} = t^{T_{n-1}} + \mathrm{exe}(T_n) + f(n) \times O_{\mathrm{com}}, \quad i = 1, 2, 3 \cdots n, \tag{16}$$

where t^{T_n} is the makespan of the task T_n and task T_{n-1} is the predecessor task for T_n. According to formula (16), the task T_n cannot be executed until the task T_{n-1} has been finished. Moreover, the symbol O_{com} represents the communication overhead function $f(n)$, and $\mathrm{exe}(T_n)$ represents the number of messages between nodes during the execution of the task T_n and the execution overhead of the task T_n, respectively. Finally, the makespan of the whole mission group is

determined by the following:

$$t^{AT} = t^{T_n} - t^{T_1}. \quad (17)$$

Here, t^{AT} denotes the makespan of the overall mission group, and its value is just the makespan difference between the last executed task and the first executed task.

In the detailed simulation model, the communication overhead adopts a variable value related to mobility models. Depending on the API (Application Program Interface) provided by OMNeT++, the value of communication overhead changes according to the distance between two nodes plus a normal noise.

To evaluate the stability of different mobility models, the same mission group is assigned to the experimented UAV clusters which have been deployed with the four mobility models. Then, the makespans of individual tasks and the whole mission group during the simulation are recorded. To simplify the experiment, the execution overhead of individual tasks is ignored and $\text{exe}(T_n) = 0$ is set. The results of different movement models are shown in Table 8.

Table 8 shows the simulation results of the makespans for the four mobility models, which mainly include the average makespan of individual tasks and the mean makespans of the whole mission group for clusters G_1 to G_5. For the ML-RNGM model, the average makespan of individual tasks is about 2.61 ms, and the average makespan of the whole mission group is approximately 41.47 ms. These two statistical values are dramatically smaller than the other three mobility models. Besides, for the random class mobility model, in most cases, it cannot execute deployed tasks such as the failure of communication. Thus, the average makespan of individual tasks and the whole mission group for the random class mobility model cannot be counted, so both of them are described by NaN. Based on the data statistics in Table 8, the model reduces the average makespans of individual tasks and the whole mission group by 42% compared with the best of the other three models.

5.3. Results and Discussions. The logical reference point of the proposed mobility model is changed to the physical reference node. Thus, a node can adjust its motion vector in real-time according to the movement change of its neighboring nodes. Meanwhile, the introduction of the reference node selection algorithm further constrains the randomness of the node movement and improves the stability of the network. Besides, when each node refers to the movement of the logical adjacent node which just will execute the following task, it can effectively shorten the communication distance between nodes and reduce the transmission delay of messages and packet loss.

The proposed model has the high concentration, and it is very beneficial to the intertask communication within the mission group in the UAV swarm. Meanwhile, high concentration reduces the time overhead brought by information exchange between the associated UAVs and effectively guarantees the makespan of the subsequent mission groups assigned to the considered UAV cluster. The model potentially has shorter message transmission delay and finally results in better real-time performance.

Therefore, in the scenario where UAV clusters perform complicated tasks collaboratively, the proposed mobile model can effectively shorten the execution time for missions and tasks. The end-to-end delay and some other comparison indicators have been analyzed to illustrate the performance differences of the four reference node selection algorithms.

6. Conclusion

In this paper, a new group mobility model is proposed for the UAV swarm, namely, the ML-RNGM model. The UAVs equipped with wireless Ad hoc capabilities are required to achieve a lower average end-to-end delay while maintaining connectivity with the reference node and a better improvement in the average link duration. A reference node selected parameter (H level) has been introduced into the mobility model to select the most suitable reference node that ensures the real-time communication among UAVs.

Simulation results show that the movement strategy of the mobility model has a positive impact on the network connectivity and the real-time performance of mission and tasks, and the movement strategy can be listed as follows:

(1) Use the physical reference node instead of the logical reference point

(2) Use the multilevel reference node selection algorithm

According to the comparison results among the random class mobility model, the reference class mobility model, the associated class mobility model, and the ML-RNGM mobility model, the adopting of reference node strategy in the mobility model is beneficial to ensure the connectivity rate of clusters and the real-time performance of message transmission. Besides, the strategy of selecting reference nodes according to the task relevance will enhance the connectivity between UAVs which are performing tasks within the same mission group and further optimize the completion time of individual tasks and the overall mission group. In addition, the ML-RNGM model has the advantage of concentrating UAVs within a limited scope, which would be much suitable for the scenario with dense and concentrated combat targets. The mobility model proposed in this paper is suitable for multi-UAV cluster systems to perform complex missions, especially when the missions have high demands on real-time performance and network connectivity. By decomposing complex missions and assigning them individually to each UAV, the whole cluster can be moved according to the reference nodes.

For future research, the stability of different routing protocols can be studied based on the mobility model in the collaborative UAV scenario. In addition, high precision clock synchronization technology is worth pursuing to ensure the cooperation of UAV clusters. The impact of clock synchronization on UAV cluster collaboration can be studied.

Furthermore, the semiphysical networking environment can also be deployed to perform UAV cooperation.

Data Availability

The underlying data supporting the results of our study could ask for the author Xiaoyan Gu (E-mail: xiaoyangu@-bistu.edu.cn) or the corresponding author Feng He (E-mail: fenghe@buaa.edu.cn).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (grant numbers: 62071023 and 71701020) and the Equipment Pre-research Field Fund (grant number: 61403120404).

References

[1] C. E. Palazzi, A. Bujari, G. Marfia, and M. Roccetti, "An overview of opportunistic ad hoc communication in urban scenarios," in *2014 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*, pp. 146–149, Piran, Slovenia, 2014.

[2] Z. Li, Q. Li, and H. G. Xiong, "Avionics clouds: a generic scheme for future avionics systems," in *2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC)*, pp. 6E4-1–6E4-10, Williamsburg, VA, USA, 2012.

[3] X. J. Tu, *Research on Key Technologies in Avionics Clouds Design and Performance Analysis*, Beijing University of Aeronautics and Astronautics, Beijing, 2013.

[4] J. Lu, F. He, and H. G. Xiong, "System and network architecture of avionics cloud," *Avionics Technology*, vol. 48, no. 3, pp. 1–9, 2017.

[5] R. W. Wang, F. He, X. Zhou, J. Lu, and E. S. Li, "Avionics cloud multi-layer task scheduling model for UAV swarm," *Acta Aeronautica et Astronautica Sinica*, vol. 40, no. 11, pp. 221–232, 2019.

[6] D. Orfanus and E. Freitas, "Comparison of UAV-based reconnaissance systems performance using realistic mobility models," in *2014 6th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pp. 248–253, St. Petersburg, Russia, 2014.

[7] M. M. Zonoozi and P. Dassanayake, "User mobility modeling and characterization of mobility patterns," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 7, pp. 1239–1252, 1997.

[8] E. M. Royer, P. M. Melliar-Smith, and L. E. Moser, "An analysis of the optimum node density for ad hoc Mobile networks," in *ICC 2001. IEEE International Conference on Communications. Conference Record (Cat. No.01CH37240)*, vol. 3, pp. 857–861, Helsinki, Finland, 2001.

[9] E. Hyyti and J. Virtamo, "Random waypoint mobility model in cellular networks," *Wireless Networks*, vol. 13, no. 2, pp. 177–188, 2007.

[10] B. Liang and Z. J. Haas, "Predictive distance-based mobility management for PCS networks," in *IEEE INFOCOM '99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320)*, vol. 3, pp. 1377–1384, New York, NY, USA, 1999.

[11] G. Wang, J. Zhang, and B. Li, "Study on improved algorithm of random direction mobility modeling in high-dynamic environment," *Acta Aeronautica Et Astronautica Sinica*, vol. 28, no. 2, pp. 376–379, 2007.

[12] C. Bettstetter, "Smooth is better than sharp: a random mobility model for simulation of wireless networks," in *Proceedings of the 4th ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, pp. 19–27, Berkeley, CA, USA, 2001.

[13] Z. Heng-Yang, X. Dan, L. Yun-Hui, C. Xuan-ping, and Z. Hy, "A smooth gauss-semi-Markov mobility model for wireless sensor networks," *Journal of Software*, vol. 19, no. 7, pp. 1707–1715, 2008.

[14] O. Bouachir, A. Abrassart, F. Garcia, and N. Larrieu, "A mobility model for UAV ad hoc network," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 383–388, Orlando, FL, USA, 2014.

[15] X. Hong, M. Gerla, G. Pei, and C. C. Chiang, "A group mobility model for ad hoc wireless networks," in *Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, pp. 53–60, Washington, Seattle, USA, 1999.

[16] M. Sivajothi and E. R. Naganathan, "Analysis of reference point group mobility model in mobile ad hoc networks with an ant based colony protocol," *Proceedings of the International Multi Conference of Engineers and Computer Scientists (IMECS)*, vol. 2174, no. 1, pp. 342–346, 2009.

[17] X. F. Li, T. Zhang, and J. F. Li, "A particle swarm mobility model for flying ad hoc networks," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pp. 1–6, Singapore, 2017.

[18] K. H. Wang and B. Li, "Group mobility and partition prediction in wireless ad-hoc networks," in *2002 IEEE International Conference on Communications. Conference Proceedings. ICC 2002 (Cat. No.02CH37333)*, vol. 2, pp. 1017–1021, New York, NY, USA, 2002.

[19] F. Bai, N. Sadagopan, and A. Helmy, "Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks," in *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428)*, vol. 2, pp. 825–835, San Francisco, CA, USA, 2003.

[20] E. Kuiper and S. Nadjm-Tehrani, "Mobility models for UAV group reconnaissance applications," in *2006 International Conference on Wireless and Mobile Communications (ICWMC'06)*, pp. 33–33, Bucharest, Romania, 2006.

[21] M. Sayeed and R. Kumar, "An efficient mobility model for improving transmissions in multi-UAVs enabled WSNs," *Drones*, vol. 2, no. 3, pp. 31–38, 2018.

[22] M. U. Ilyas and H. Radha, "The influence mobility model: a novel hierarchical mobility modeling framework," in *IEEE Wireless Communications and Networking Conference, 2005*, vol. 3, pp. 1638–1643, New Orleans, LA, USA, 2005.

[23] E. Yanmaz and H. Guclu, "Stationary and mobile target detection using mobile wireless sensor networks," in *2010 INFOCOM IEEE Conference on Computer Communications Workshops*, pp. 1–5, San Diego, CA, USA, 2010.

[24] J. Schleich, A. Panchapakesan, G. Danoy, and P. Bouvry, "UAV fleet area coverage with network connectivity constraint," in *Proceedings of the 11th ACM international symposium on Mobility management and wireless access*, pp. 131–138, Barcelona, Spain, 2013.

[25] K. Daniel and C. Wietfeld, "Using public network infrastructures for UAV remote sensing in civilian security operations," in *Conference on homeland security affairs*, pp. 1–11, Germany, 2011.

[26] Z. Han, A. L. Swindlehurst, and K. J. R. Liu, "Optimization of MANET connectivity via smart deployment/movement of unmanned air vehicles," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 7, pp. 3533–3546, 2009.

[27] E. Yanmaz, "Connectivity versus area coverage in unmanned aerial vehicle networks," in *2012 IEEE International Conference on Communications (ICC)*, pp. 719–723, Ottawa, ON, Canada, 2012.

[28] M. Musolesi and C. Mascolo, "A community based mobility model for ad hoc network research," in *Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality*, pp. 31–38, Florence, Italy, 2008.

[29] G. Jayakumar and G. Ganapathi, "Reference point group mobility and random waypoint models in performance evaluation of MANET routing protocols," *Journal of Computer Systems, Networks, and Communications*, vol. 2008, article 860364, 10 pages, 2008.

[30] E. Yanmaz, C. Costanzo, C. Bettstetter, and W. Elmenreich, "A discrete stochastic process for coverage analysis of autonomous UAV networks," in *2010 IEEE Globecom Workshops*, pp. 1777–1782, Miami, FL, USA, 2010.

[31] S. Rashed and M. Soyturk, "Analyzing the effects of UAV mobility patterns on data collection in wireless sensor networks," *Sensors*, vol. 17, no. 2, p. 413, 2017.

[32] V. Gazi, "Stability analysis of swarms," *IEEE Transactions on Automatic Control*, vol. 48, no. 4, pp. 692–697, 2003.

[33] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad-hoc network research," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, 502 pages, 2002.

[34] F. He and E. Li, "Deterministic bound for avionics switched networks according to networking features using network calculus," *Chinese Journal of Aeronautics*, vol. 30, no. 6, pp. 1941–1957, 2017.

[35] X. Zhou, H. Xiong, and F. He, "Hybrid partition-and network-level scheduling design for distributed integrated modular avionics systems," *Chinese Journal of Aeronautics*, vol. 33, no. 1, pp. 308–323, 2020.

[36] P. Deville, C. Song, N. Eagle, V. D. Blondel, A. L. Barabási, and D. Wang, "Scaling identity connects human mobility and social interactions," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 113, no. 26, pp. 7047–7052, 2016.

[37] X. Y. Yan, W. X. Wang, Z. Y. Gao, and Y. C. Lai, "Universal model of individual and population mobility on diverse spatial scales," *Nature Communications*, vol. 8, no. 1, 2017.