



Hybrid AI-based 4D trajectory management system for dense low altitude operations and Urban Air Mobility

Yibing Xie ^a, Alessandro Gardi ^{b,a,*}, Man Liang ^a, Roberto Sabatini ^{a,b}

^a School of Engineering, RMIT University, Bundoora, VIC 3083, Australia
^b Department of Aerospace Engineering, Khalifa University, Abu Dhabi, United Arab Emirates

ARTICLE INFO

Editor: George N Barokas

Keywords:

Avionics
 Air Traffic Management
 Advanced Air Mobility
 Urban Air Mobility
 Unmanned Aircraft Systems
 UAS Traffic Management
 Demand-capacity balancing
 Decision support system
 Metaheuristics
 Machine learning
 Four-dimensional trajectory
 Path planning
 AI (artificial intelligence)
 avionics
 cyberphysical systems

ABSTRACT

Urban Air Mobility (UAM) has emerged as a promising solution to address some of the challenges of transportation congestion and associated pollution, especially in large cities. However, the development of drone transportation and UAM services is limited by the capacity of the low altitude airspace where these new vehicles will operate. Without suitable regulatory advancements and associated traffic management systems, air traffic in the densest low-altitude sectors may incur congestion, which, in addition to affecting operational efficiency, can increase systemic risk and fuel emergency occurrences, thereby affecting the safety of people and property in the air and on the ground. To address these challenges, this study aims to develop an intelligent Uncrewed Aircraft Traffic Management (UTM) system that leverages the complementary strengths of metaheuristic and machine learning algorithms for an effective management of dense low altitude airspace. The UTM system determines time-based three-dimensional airspace Demand-Capacity Balancing (DCB) solutions by processing real-time data updates and dynamically replanning flight paths and DCB decisions in any given context, while also providing UAM operators with relevant inputs for autonomous decision-making. Simulation-based verification activities in representative conditions show that the proposed UTM system has the ability to effectively resolve overload instances and minimize potential conflicts in low-altitude airspace, with an operationally acceptable running time. We conclude that the proposed hybrid algorithm can support a successful implementation of UAM services in and around cities, and it has high potential to address critical airspace resource constraints also in traditional Air Traffic Flow Management (ATFM).

1. Introduction

Uncrewed Aircraft Systems (UAS) and Vertical Take-Off and Landing (VTOL) technology are undergoing significant advancements, resulting in various new air services being proposed across different sectors [1-3]. It is foreseeable that commercial UAS operations and Urban Air Mobility (UAM) services will become integral parts of our daily lives within the next two decades. However, the operation of a variety of conventional and uncrewed aircraft in shared airspace is bound to introduce significant challenges, including congestion due to the limited capacity of urban airspace, efficiency loss caused by the uneven allocation of airspace resources, and potential conflicts or even collisions. Thus, UAM service providers, aircraft/UAS operators, and government agencies will need suitable UAS Traffic Management (UTM) systems to ensure safe, efficient, and equitable operation for all types of aerial vehicles [4,5]. To

address the challenges of high-density operations by UAS and other emerging UAM platforms, the National Aeronautics and Space Administration (NASA) and the Federal Aviation Administration (FAA) in the U.S. have launched the UTM and Advanced Air Mobility (AAM) programs, while the Single European Sky Air Traffic Management (ATM) Research Joint Undertaking (SESAR JU) in Europe has launched the U-Space program [6,7]. These initiatives worked to establish viable Concepts of Operations (CONOPS) and viable evolutionary pathways for UAS and other emerging aircraft considering their greater manoeuvrability and shorter tactical and strategic conflict response cycles. For instance, as part of U-Space and of the DACUS project in particular [8], SESAR and partner organizations analysed the need to transition tasks traditionally handled by ground-based systems and human ATM operators - such as separation assurance - to airborne systems integrated into each aircraft. This shift is necessary as the microscopic scale and faster

* Corresponding author at: Assistant Professor in Aerospace Engineering, Khalifa University Office R02036, KU Main Campus Al Bun St, Abu Dhabi Zone 1, United Arab Emirates.

E-mail address: alessandro.gardi@ku.ac.ae (A. Gardi).

pace of UAM operations in many cases exceeds the performance of existing aeronautical ground-based Communication, Navigation and Surveillance (CNS) systems in providing reliable air-ground conflict detection and resolution. Therefore, ground-based UTM entities are envisioned to focus on a more strategic and supervisory management of traffic [9,10], analogous to Air Traffic Flow Management (ATFM) or Air Traffic Flow and Capacity Management (ATFCM) services in traditional ATM. These services alleviate congestion and delays in conventional airspace through Demand Capacity Balancing (DCB) and careful traffic synchronization [11], but have been historically limited by legacy flight plans, mostly based on two-dimensional (2D) route selection and with low suitability for digital handling. The 4D Trajectory (4DT) concept, which combines three-dimensional spatial route descriptors and the time dimension, supports the definition of flight paths which can be assessed against any possible conflict with other aircraft. Moreover, 4DT can be optimized against multiple objectives and constraints, digitally negotiated between airborne and ground entities, verified against all other intents and support advanced airspace management strategies [12]. The concept of 4DT Based Operations (4D-TBO) is therefore one of the most important advances which have been proposed in the last decades as it will ultimately allow ATM to resolve the existing dichotomy between offline flight planning and online traffic deconfliction. In addition to the transition to digital datalink-enabled air-ground communications and collaborative decision-making, to fully implement predictivity and optimization capabilities, it is expected that ground-based 4D-TBO systems will be characterized by an increasing reliance on higher levels of automation and increasing adoption of Artificial Intelligence (AI) [13].

According to the literature, a DCB-oriented system exploiting 4D-TBO paradigm is the preferred UTM solution as it can best integrate with conventional ATM [13]. However, even though the basic principles and methods to achieve DCB in ATM and UTM are fundamentally similar, the operational timeframes and pace of ATM and UTM will differ significantly due to the different spatio-temporal scales of operation. Moreover, the high density and complexity of envisioned urban low altitude air traffic cannot be handled by conventional ATM systems and operational paradigms due to unacceptably high workload for air traffic controllers [14]. Furthermore, a successful UTM service will also require to consider the environmental impacts and community acceptance within cities [15,16]. Therefore, it is essential to develop a UTM system exploiting suitable AI-based functionalities to handle the specificities and requirements of low altitude airspace in dense urban areas, which is capable of dynamically responding to contingencies in real-time, and ensure a safe, efficient and environmentally sustainable operation.

1.1. Aims and structure

In our research, we aim to develop a new UTM system that can support flexible DCB and route planning decision support for high-density air traffic in low altitude airspace in urban areas. Several drivers are considered in the design, including safety and security of UAM services, efficiency, and a reduction of the uncertainty associated with dense operations and complex urban environments. There are three main contributions in this article. The first is to develop a DCB functionality based on a new powerful multi-objective hybrid optimization approach. This method utilizes a mixture of Metaheuristic Algorithms (MA) and Machine Learning (ML) techniques to generate and optimize strategic decisions and DCB tactical actions. The second contribution is to propose a 4DT-based route planning method, which determines the optimal route through 4DT selection and backtracking, and interacts with the DCB solution engine to ensure feasibility and consistency. The third contribution is the implementation of a virtual reality sandbox environment to prototype and test UTM/UAM system functionalities such as the ones proposed in this article, recreating high-density operation scenarios in low altitude airspace with simulated environment to

generate labelled datasets that can be used to train supervised and reinforcement learning models. The rest of this paper is structured as follows: [Section 2](#) conveys a review of relevant literature on ATFM and UTM systems and technologies, as well as most relevant and promising MA and ML algorithms in UTM/UAM. [Section 3](#) presents the proposed UTM system concept and airspace architecture. [Section 4](#) details the models and algorithms used in the proposed system, including reinforcement learning and heuristic algorithms. [Section 5](#) describes the verification activities conducted within virtual simulation environment, for method and data validation purpose. [Section 6](#) presents the conclusions and future research directions.

2. Artificial intelligence and traffic flow management

ATFM is one of the fundamental services established to ensure the safe, efficient, and orderly flow of air traffic [17,18]. Over the years, ATFM has relied on various strategies to accomplish DCB. For example, Ground Holding Problem (GHP), focused on managing the distribution of holding times at departure airport to solve the congestion either in the air or at the destination airport [19,20]. Other strategies consider enroute constraints and route diversions to balance the capacity and demand of different airspace sectors, ultimately improving the overall performance of the air traffic system [21]. Generally, there are three main phases in ATFM, which are recommended by the International Civil Aviation Organization (ICAO) [17]: a strategic planning phase taking place six months to a few days before operations; the pre-tactical planning phase occurring one day before departure and responsible for daily airspace coordination; and the tactical phase that is exclusively reserved for day-of-action management. ATFM utilizes various strategies during the planning phase to balance airspace demand and capacity, such as flight departure/landing time allocation, ground delays, and alternative routes (diversions) [22]. ATFM shall address DCB both at congested airport and at capacity constrained enroute/regional airspaces. The majority of conventional solutions focussed on effectively balancing traffic demand in capacity-constrained locations and significant research has been conducted around spreading flight delays, optimizing ground operations, and managing departure/arrival slots to reduce congestion-associated costs and delays [23,24]. However, relying solely on ground delays and slots is not sufficient and makes the achieved balance susceptible to weather perturbations and other unforeseen disruptions. As a result, researchers have also developed various optimization models to address the saturation and capacity imbalance in evolving enroute/regional airspace settings [25,26]. US' Next Generation Air Transportation System (NEXTGEN) [27] and EU's SESAR [5] programs aimed to enhance the performance of ATM and forecasting. 4D-TBO are considered one of the most promising advanced technologies within these programs, with the potential to significantly improve the safety and efficiency of air travel [28]. By adopting a 4DT, which incorporates time as the fourth dimension, air traffic controllers can better predict and manage the routing and synchronization of aircraft, resulting in more efficient use of airspace and reducing delays [29]. Considering more autonomous aircraft and UAS will be dominant in low airspace, the NEXTGEN and SESAR programs both have extended traditional ATM research to UTM domain to handle the challenges of automation in future dense operations. For example, UTM is divided into four Technical Competence Levels (TCL) to satisfy the demand for both Visual Light of Sight (VLOS) and Beyond Visual Light of Sight (BVLOS) flight operations for manned and unmanned aircraft in different risk associated scenarios [30]. From TCL 1 to TCL4, the complexity of deployment scenario increases, and the associated risk will increase. TCL 4 is to test UTM in true urban environment, with dense population, high traffic density, urban application, dense BVLOS operation, and large scale contingency management.

Various researchers have proposed new airspace structure concepts to meet the demands of performing UAM operations in low-altitude dense urban regions, including full mixing, stratification, zones, and

tubes [31-37]. Others applied metaheuristic algorithms for the resource optimization problems to help decision making. A large number of studies have demonstrated the outstanding performance of MA compared with classical mathematical approach to handle a wide variety of optimization problems in relatively short computation time [38]. They are often used to solve complex problems, such as multi-objective optimization of trajectory tracking for elevator systems [39], reliability design of optimized UAS [40], route planning with time windows and vehicle constraints [41]. Genetic Algorithms (GA) are widely used variants of MA, known for their parallelism, randomness, high efficiency, and global search optimization capabilities. They are especially useful for solving Non-deterministic Polynomial (NP) problems, where the objective function has "black-box" characteristics [42]. Researchers have successfully applied GA to various optimization problems. For instance, Wang proposed a vehicle routing solution combining Clustering Algorithm (CA) and GA, which reduces transportation mileage and improves vehicle loading rate while ensuring stability and fairness among members [43]. Jin demonstrated an archive-based GA optimization framework to integrate and solve road traffic control problems in aviation [44]. Compared to typical GA, this approach achieves faster convergence. Samsam utilized an improved Non-dominated Sorting GA-II (NSGA-II), to perform multi-objective optimization and to determine smooth multi-pulse trajectory for long-distance rendezvous of on-orbit targets [45].

In term of path planning methods for vehicles, bio-inspired algorithms such as GA [46], Ant Colony Optimization (ACO) [47], Gray Wolf Optimizer (GWO) [48], Particle Swarm Optimization (PSO) [49], Differential Evolution (DE) algorithms [50], have been applied with success. However, these algorithms are computationally demanding, making them unsuitable for solving complex real-world scenarios [51]. On the other side, Sampling-Based Planning Algorithms, like Probabilistic Roadmap (PRM) [52] and Rapid Exploration Random Tree (RRT) [53], are commonly used to solve problems in high-dimensional spaces and complex constraints. However, the search efficiency of these algorithms is low due to space modelling and state sampling [54]. The A* algorithm is a classic heuristic search algorithm and a type of Search-Based Planning Algorithm. It improves the search efficiency of the Dijkstra algorithm by adding a heuristic function. Today, the improved algorithm based on A* is widely used in search problems. For example, Meng [55] proposed an improved hybrid A* algorithm to improve the safety of path planning in autonomous parking systems, as well as the search efficiency in space and time. Neto [56] also used the A* algorithm in a UAS path planning method for the UTM system after comparing multiple path search algorithms.

Nowadays, intelligent systems that utilize ML algorithms have become increasingly popular for solving a variety of tasks. More researchers started to use ML algorithms to optimize UAS operations, such as image recognition analysis [57], path planning [58], and intelligent decision-making [59], etc. Unsupervised learning is a powerful ML process. Clustering algorithms, which are important and popular unsupervised ML algorithms, are used to discover the underlying structure in unlabelled data by grouping similar items into clusters [60]. One of the most commonly used clustering methods is K-means clustering, which divides a dataset into K clusters based on their similarity to each other. It has been widely deployed in various fields due to its effectiveness in uncovering patterns and relationships in large dataset [61]. In terms of applying unsupervised learning method to UTM problems, Hao used K-means algorithm for UAS image recognition and classification, the results proved that it is a promising application that addresses the high-risk problem of manual geological survey work [62]. Khalil implemented K-means clustering method in predicting drone classification and reducing path loss in search and rescue operations [63]. Chen [64] integrated K-means clustering and decision tree classification algorithms for mining with UAS, and the air traffic flow data shows great potential for developing forecasting systems with UAS that could aid in strategy formulation, operations management, and investment planning [48].

3. UTM system and DCB formulation

The proposed intelligent UTM system aims to balance traffic demand and airspace capacity in dense operations. Fig. 1 depicts the top-level architecture of the considered UTM system interactions, which are primarily organized, coordinated, and monitored through highly automated Application Programming Interfaces (API) [65]. As UTM is conceived as a complement to traditional ATM, a Flight Information Management System (FIMS) is required to ensure the interoperability of the UTM system with contemporary ATM systems. In the UTM system, UAM vehicle Service Suppliers (USS) serve as an intermediary entity between UAM vehicle operators and regulators. They act as a communication bridge for data exchange and information sharing between the UTM system and UAM vehicle operators. Additionally, the UTM system needs to access necessary operational information from data service providers, such as weather data, geographic environment data, drone performance data, and airspace surveillance data. The UTM system also needs to provide operational plans and status information to public safety departments. These data and information are critical to the intelligent UTM system operation. In this study, we consider the situations and corresponding data elements that UTM will encounter during actual operations. These include weather conditions, geographic zones, emergency alternate landing sites, CNS performance of airspace and aircraft, flight movement patterns of multiple UAM vehicles and some relevant flight dynamics characteristics. We also consider the diversity in UAM vehicle missions and in DCB tactical actions. High-fidelity flight dynamics and performance will not be considered yet at this stage, however, since these models are not yet sufficiently mature for validation. Therefore, the 4DT used in this article is mainly used to reflect aircraft intents and resolve conflicts.

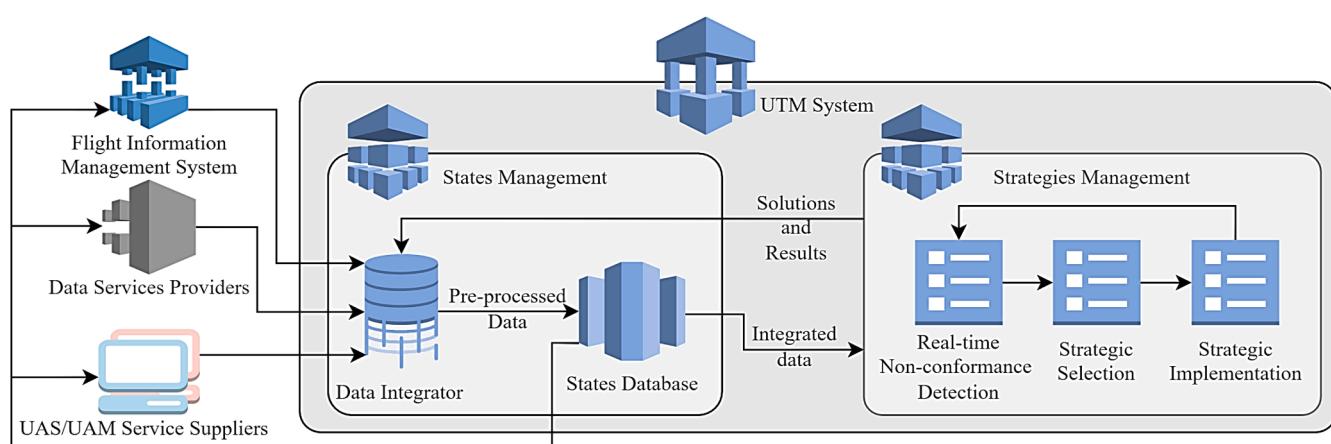
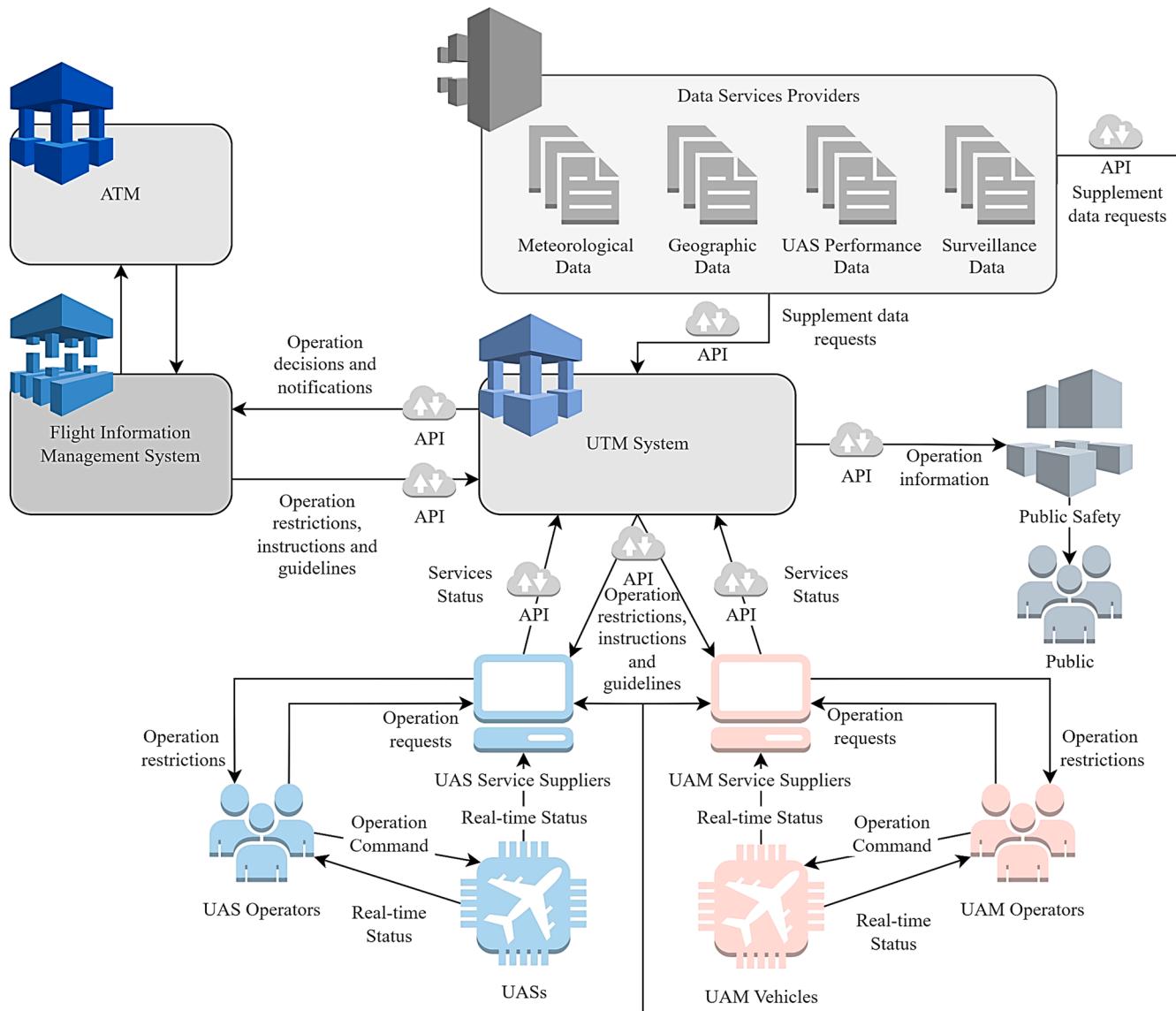
3.1. UTM system framework

Fig. 2 illustrates the proposed UTM system DCB functionality, which consists of two main modules: State Management and Strategies Management. The States Management module processes collected multi-dimensional data through the Data Integrator and stores them in the States Database. The Strategies Management module is responsible for non-conformance detection, decision-making, and task execution. The Non-conformance Monitoring component processes information integrated from the States Database to identify the current and predicted states of the airspace and sectors. The Strategies Decision-making component is responsible for state decision-making of the UTM system, and the Strategic Actions module simulates the execution of the UTM system's decision and outputs the states information after the decision is executed. The Data Collector collects multi-dimensional data from multiple sources, including CNS performance data, aircraft sensor data, weather service data and terrain data. When the UTM operator establishes tasks and makes strategic decisions, the data are transmitted directly to the Data Integrator component of the States Management module. The Strategies Management module obtains the integrated data from the States Database. The DCB decision plan and data output by the module are then returned to the Data Integrator component to update the States Database again. This data loop processing mode allows the system to respond to both the states data input by external components (feedback update) and the decisions and deployments made by the system (active update). Additionally, when the UTM operator needs to retrieve and monitor system operation, all the necessary data comes from the integrated data in the States Database. The attribute of the data is read-only, ensuring that the data in the States Database cannot be artificially changed.

3.1.1. States management

The State Management module is composed of two components: the Data Integrator and the States Database, as shown in Fig. 3.

This module mainly integrates four states attributes: airspace states, aircraft states, weather states, and mission states. When UTM is



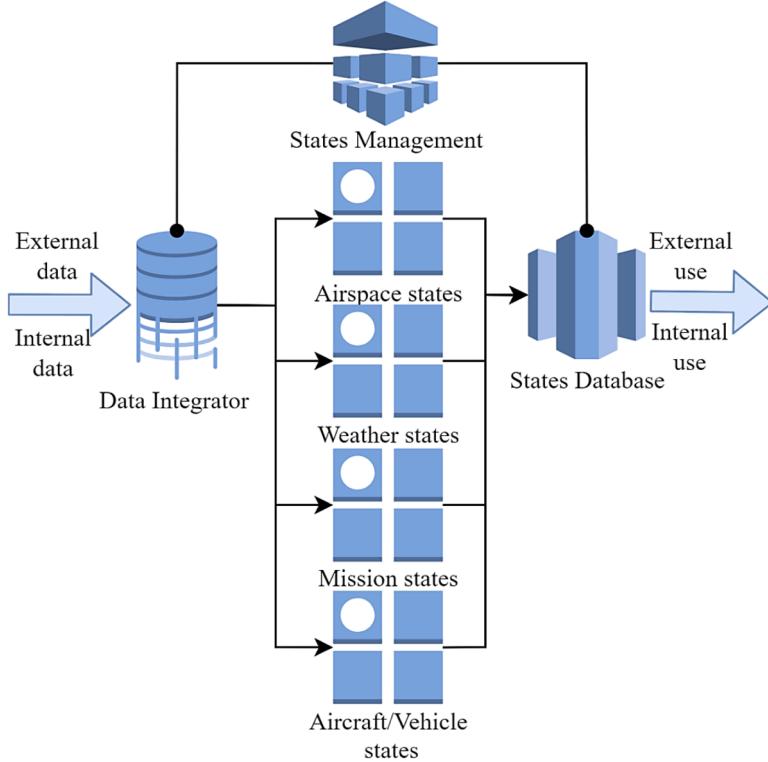


Fig. 3. Architecture of the states management module.

operationally deployed, the system generates a large amount of real-time data acquired from multiple sensors, as well as data generated internally while the system is running. Therefore, Data Integrator components are necessary to fuse and update the data, reducing uncertainty caused by uncoupling system data. The function of the States Database is to store and retrieve integrated data. These two components have a clear pre-processing sequence: all external input and internal output data are processed by the Data Integrator, and all external retrievals and data used within the system are called only from the States Database. Data in the States Management module can have one of two state characteristics: system update (SU) or external update (EU). Data with SU characteristics update state information in the States Database dynamically and automatically as the UTM system runs. This update mode enables each decision made by the Strategies Management component to include uncertainties from the previous state. Similarly, data with EU characteristics also dynamically update state information in the States Database. However, such data comes exclusively from external sensor components integrated with the UTM, such as aircraft sensors, weather collection stations, and UTM tactical decisions. The system only fetches and updates data related to it, without modifying the input state data. The majority of data contain both SU and EU characteristics. For instance, the system selects and executes DCB tactical actions in the airspace for all aircraft based on the current state data during calculations. The system obtains the traffic capacity state based on time, which belongs to the SU data as the states data is generated by the internal circulation of the system. However, to determine whether the actual airspace traffic capacity state matches the simulated state, monitoring data needs to be obtained from data service providers, which belongs to the EU data. The complete traffic capacity state includes both the calculation data in the system and external monitoring data.

3.1.2. Strategies management

Fig. 4 shows the strategy management module, which can be divided into three components: Real-time Non-conformance Detection, Strategies Selection and Strategic Implementation.

3.1.2.1. Real-time Non-conformance Detection. Real-time Non-conformance Detection is responsible for monitoring the internal and external data of the system, which is comprised of four items.

- Aircraft Monitoring: monitors the flight states of the UAM vehicle such as flight speed, remaining flight energy, mission and sensor monitoring data, and load states. The aircraft state data help the system determine whether the aircraft can complete the flight mission and execute the strategic actions later issued by the system.
- Weather Monitoring: which monitors the impact of changes in meteorological environmental conditions to the UTM system. Weather data are mainly obtained from local weather service operators and aircraft internal sensors and allow the UTM system to make strategic adjustments to the airspace and sector environment, as well as the regional environment where each aircraft is located.
- Airspace Capacity Monitoring: which monitors the dynamic availability and demand states of the airspace. This data help to program the aircraft flight plan rationally by adjusting the capacity and demand of the airspace sectors.
- Traffic Density Monitoring: monitors the trajectory and density of aircraft in airspace or sectors. The main objective of this component is to allocate all aircraft to different airspaces and sectors reasonably to avoid airspace and sector overload situations.

3.1.2.2. Strategies selection. Strategies Decision-making is responsible for selecting appropriate strategic actions based on the current airspace, aircraft, weather, and mission status. This component has two sub-items: Conventional Planning and Non-conventional Planning. The Conventional Planning item triggers the task management function to generate, change, and cancel flight tasks for the aircraft, while the Non-conventional Planning item responds to the anomaly state of the airspace, aircraft, mission, and environment, and restores the balance of demand and capacity in the airspace and sector.

3.1.2.3. Strategic implementation. The UTM system can offer a variety of

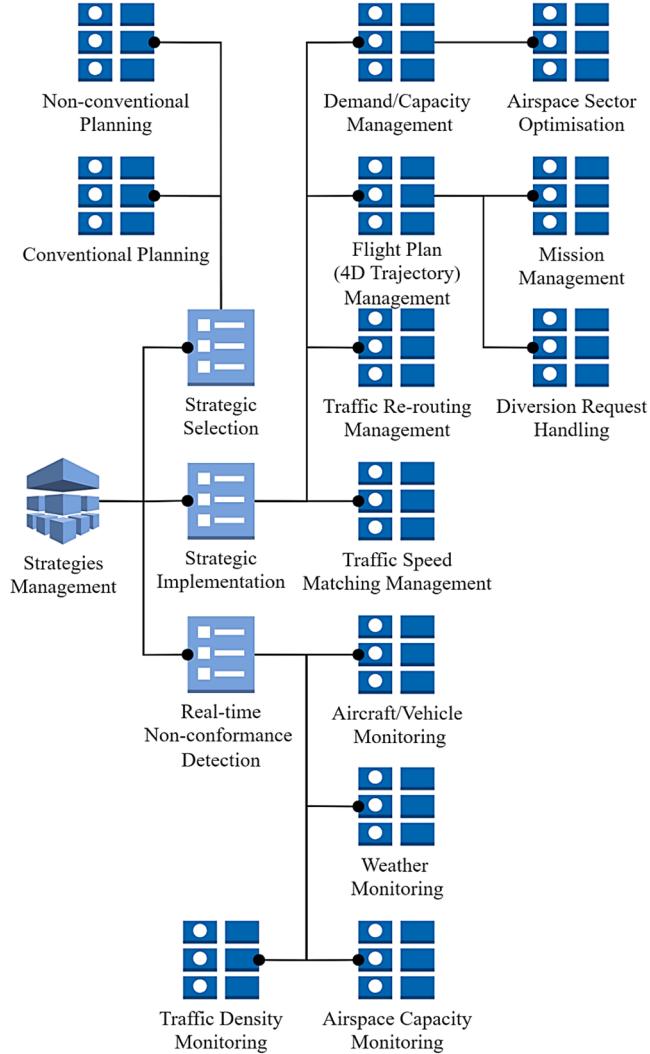


Fig. 4. Architecture of the strategies management module.

Strategic Actions, which can be divided into three types: airspace flow adjustment, flight plan adjustment, and trajectory adjustment. Demand/capacity management belongs to the airspace flow adjustment, which optimizes the capacity and demand of airspace sectors to solve the airspace overload situation. Flight Plan management has two sub-items: Mission Management and Diversion Request Handling. The Traffic Re-routing Management and Traffic Speed Matching Management are the sub-items of trajectory adjustment Strategic Actions. Traffic Re-routing Management generates new flight paths for aircraft to avoid congested or overloaded airspace and sectors. Traffic Speed Matching Management adjusts the flight speed of the aircraft.

3.2. Airspace structure

Due to the already discussed limited applicability of conventional ATM structures and procedures to UTM [10], the definition of a suitable airspace structure becomes an important part of the UTM system design. Pongsakornathien [66] introduced a CNS performance-based formulation that is particularly well-suited for autonomous airborne separation assurance and collision avoidance considering the specificities of urban airspace and beyond line-of-sight operations. The proposed structure, largely reflected in this paper, is hierarchical in nature, with the basic unit being a cuboidal cell. Multiple cells arranged horizontally form a layer and these layers are stacked on top of each other to create a multi-layer airspace structure. A set of horizontally and/or vertically

neighbouring cells form a sector. The pipe structure, on the other hand, is a four-dimensional tube-like structure that provides a fixed route for UAM vehicles. This structure can be visualized as a tubular topological route with multiple pipelines extending outward from a central node. To ensure the safety of the airspace, the concept of time-based flight trajectories (4DT) is used. When an aircraft enters a cuboidal sector, a part of the sector's capacity resources will be allocated to this aircraft based on its type. During this time, other UAM vehicles can be conditionally allowed to enter the same sector. Once the capacity of this sector is exhausted or the remaining capacity is not sufficient to accommodate another type of aircraft, this sector will be inaccessible for a specific timespan. It will remain so until its residual capacity allows to accommodate another specific type of aircraft.

To simulate the complex nature of urban low-altitude airspace, we created a simplified virtual environment with dense obstacles and geofences, shown in Fig. 5. The system can adjust the size of each cuboidal cell in the environment to better match the geographic information and scalability of the airspace. The horizontal and vertical cell dimensions can be compressed to as little as 1 meter or expanded to a size of 100 m, which provides a great flexibility for different operating environments. This feature is particularly useful in situations where the density of UAM vehicles operating in the airspace may vary. Moreover, the grey cells represent obstacles within the city, the cyan cells illustrate some discretised areas in the low-altitude airspace where UAM vehicles can operate. There are three red cells and one blue cell in the figure, which represent the original and alternate destination of the flight, respectively. When establishing a flight mission, all aircraft will be randomly allocated an original destination. At this stage, the alternate destination for all aircraft is the same.

3.3. DCB tactical actions

The following DCB tactical actions are introduced in the simulation environment:

- Keep on Plan: keep the existing flight plan unchanged, including flight speed, flight trajectory, and flight destination.
- Re-destination: the destination is changed to an alternate destination or an emergency landing site,
- Flight speed matching: accelerate 20 %: under the premise of maintaining the flight destination and flight mission, the aircraft cruise speed is accelerated by 20 %. Note that this speed matching is not suitable for climbing and descending.
- Flight speed matching, decelerate 20 %: under the premise of maintaining the flight destination and flight mission, the cruise speed is reduced by 20 %. Note that this speed matching is not suitable for climbing and descending.
- Hover 10 s: the VTOL aircraft hovers in the existing cell for 10 s, and then follows the original flight plan.

3.4. UAM vehicle missions

In our study, UAM missions are divided into two categories: point-to-point and loitering missions. Fig. 6 shows examples of both missions in the airspace. All point-to-point missions fly from random initial coordinates to a well-defined destination. The blue line in Fig. 6 shows an example path of this flight mission by a VTOL aircraft. There are also three aircraft performing surveillance missions in the airspace, and the aircraft under this mission will continue to loiter along a fixed flight trajectory in the airspace. Their trajectory is presented as magenta in the figure.

3.5. Dynamic airspace capacity computation

Dynamic airspace capacity management aims to ensure that the demand for airspace matches its capacity continuously. This can be

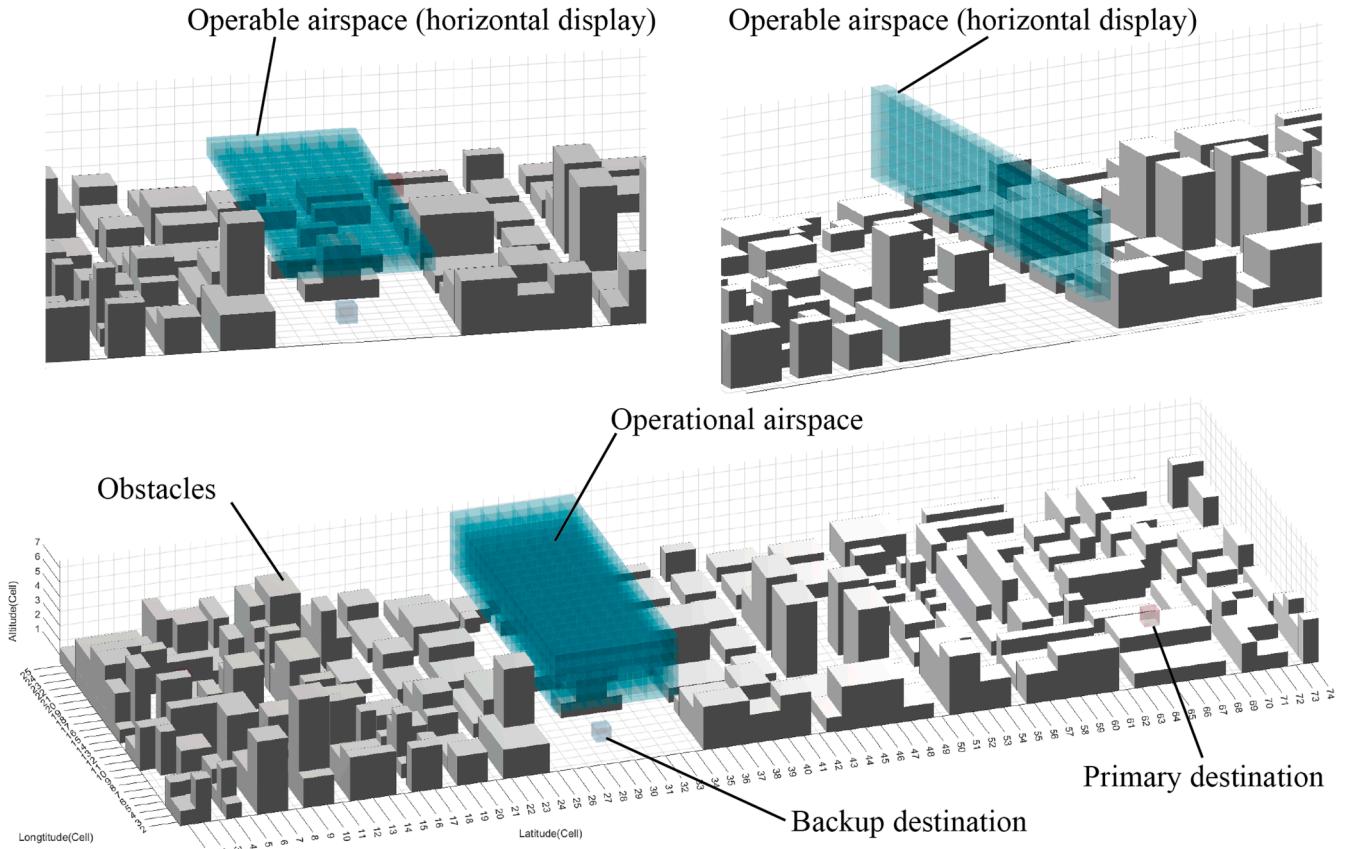


Fig. 5. Simulation environment, illustrating the adopted airspace structure.

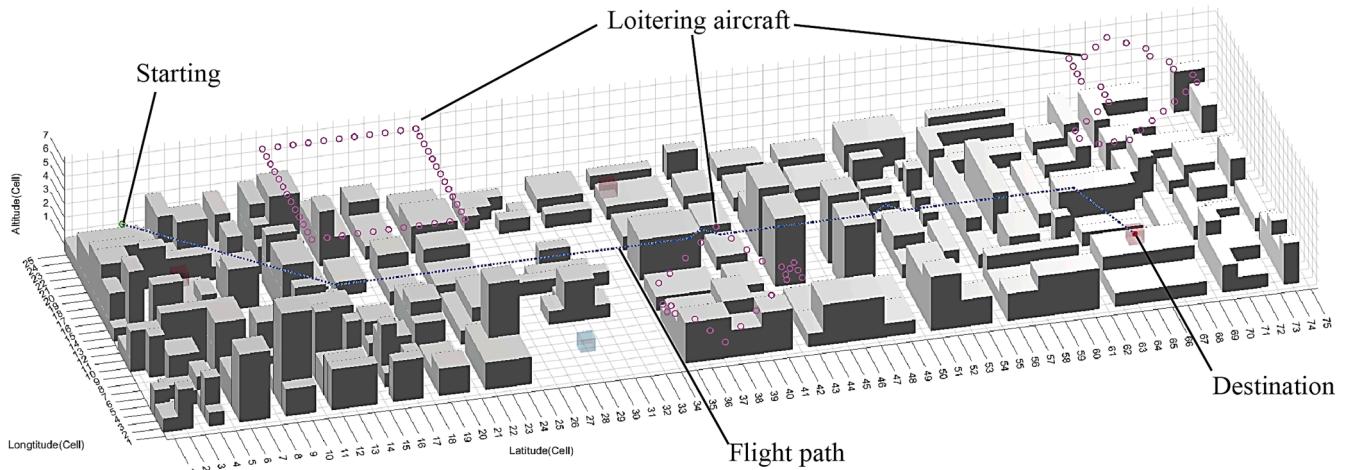


Fig. 6. UAM point-to-point and loiter mission trajectories.

achieved by adjusting the load by aircraft flows in the airspace or by adjusting the capacity of cells. It is important to note that although the cuboidal cells used in this study can be altered in their dimensions depending on the simulation environment, the cells are currently assumed static when the system is running. In other words, the volume of airspace cells will not change during the DCB operation. In this study, the initial capacity of each cell in the airspace is set at 100 %. However, the capacity factor can change based on different simulated environments. The coefficient changes based on two variable elements in the airspace – weather and the overall CNS performance state of the airspace. If the weather and CNS performance states are good, the cell capacity will remain at 100 %. However, if these variables change, the

capacity of the cell will be reduced. When both variables are estimated to decrease, the minimum capacity of the cell will decrease. As an example, in Fig. 7 the simulated airspace weather condition is windy, therefore the total capacity of cells in the airspace is reduced by 20 %. In addition, the CNS performance status is poor as well, so the total capacity of cells in the airspace will be further reduced to 85 %. In the end, the actual available capacity of each cell in the airspace compounds to 68 %. By varying the value of the total capacity, the system can provide more redundancy for aircraft operations. Moreover, the capacity occupancy factor of various aircraft is empirically classified based on the type and weight of the UAM vehicle in our system. Table 1 illustrates the assumed capacity occupancy for various types of UAM vehicles. In terms

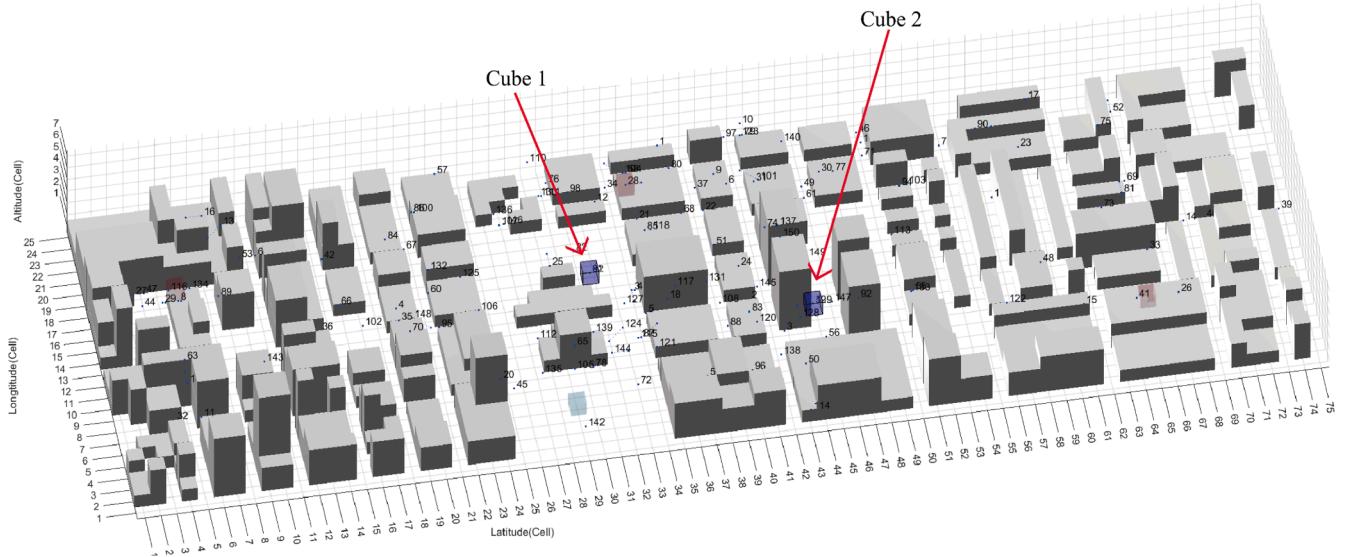


Fig. 7. Airspace simulation example.

Table 1
Capacity occupancy rate of various UAM vehicles.

Weight Class	Fixed-Wing	VTOL
Very Small	N/A	15 %
Small	35 %	25 %
Medium	50 %	35 %
Large	N/A	50 %

of weight, the classification adopts the maximum take-off mass (MTOM) of the drone. Similarly, the system also modulates the occupancy rate of individual aircraft in the airspace based on their CNS performance status. When the CNS performance of an individual aircraft is estimated to decrease, its required airspace occupancy rate is assumed to increase by 20 %. It is important to note that this mode does not change the boundaries of the airspace. Airspace capacity is a function of risk-based indicators for each predefined airspace volume, and the occupancy rate of UAM vehicles is a pre-defined quantitative assumption [8].

As illustrated in Fig. 7, two cells in the airspace both have two UAM vehicles enter at the same time. These two cells are shown as dark blue cubes on the diagram, called cube 1 and cube 2. The aircraft IDs in cube

1 are 91 and 62, and the ones in cube 2 are 59 and 129:

- Aircraft ID 62: VTOL aircraft, the weight class is Very Small, CNS performance is high, the actual Cell occupancy rate is 15 %.
- Aircraft ID 91: Fixed-wing aircraft, weight class is Small, CNS performance is high. The actual cell occupancy rate is 35 %.
- Aircraft ID 59: VTOL aircraft, weight class is Very Small, CNS performance is high. The actual cell occupancy rate is 15 %.
- Aircraft ID 129: VTOL aircraft, weight class is Large, CNS performance is high. The actual cell occupancy rate is 50 %.

Therefore, the actual capacity occupancy rate of cube 1 is $15\% + 35\% = 50\%$, and the actual capacity occupancy rate of cube 2 is $15\% + 50\% = 65\%$. Both are within the capacity limit of the airspace cell.

3.6. UAM vehicle movement grid

A grid-based approach is adopted in this paper to change aircraft heading and flight path angle in 3 spatial dimensions. Fig. 8 shows the detailed structure of 26 cells connected to the current stereo mesh. This method is improved based on our commonly used two-dimensional

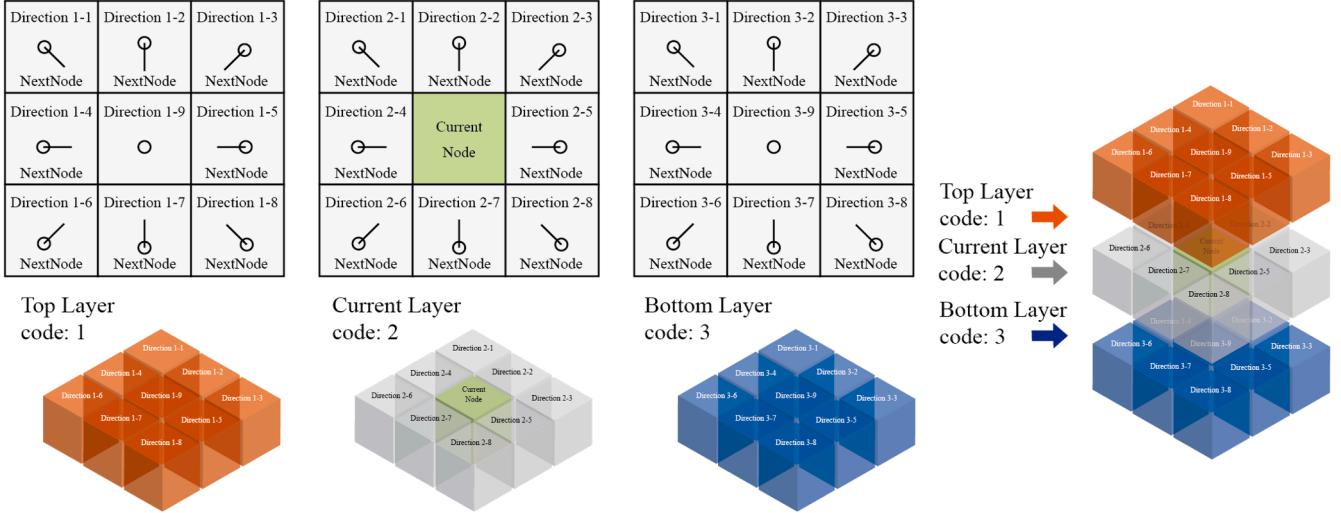


Fig. 8. 3D view of possible movement grid.

moving algorithm A star path search algorithm, which is a heuristic path search algorithm. Algorithms are described in detail in the Methodology chapter. This method has 2 features. First, it allows aircraft to move continuously in 3D space through connected nodes. Second, it allows them to select a mobile node based on the current flight direction in the airspace. The movement method is a three-layer superposition structure, and the structure takes aircraft location as the centre point. Each layer contains 9 grids. Among them, the upper layer is the layer structure reached when the aircraft moves upward, the current layer is the horizontal layer where the current aircraft is located, and the lower layer is the layer structure reached by the aircraft moving downward. The top and bottom both have 9 available nodes. There are 8 movable nodes in the current layer. It should be noted that continuous path planning needs to avoid unrealistic paths in practical applications, such as sharp turns and vertical climbs of fixed-wing aircraft. Therefore, it is necessary to constrain the path selection based on the flight dynamics characteristics of the aircraft. To this end, we have designed two ways of moving,

- 1) Continuous Linear Movement for Fixed Wing aircraft, and
- 2) Possible continuous sharp turns and vertical movement for VTOL aircraft.

3.6.1. Fixed wing aircraft movement grid

In this model, fixed-wing aircraft are not allowed to do vertical movement, so there are 8 movable directions on each layer, and a total of 24 movable directions. In actual path selection, the flight path selection is restricted by the direction of the nose. Therefore, without being restricted by obstacles, there are 3 optional flight path points per layer. Combined with climbing and descending to the adjacent layer, the movable path increases to 9. As illustrated in Fig. 9, the fixed-wing aircraft moved in direction 2-2 in the previous step, that is, it was cruising straight and level. Dark green indicates the optional direction of movement, the aircraft can choose to climb, move horizontally, or descend, to keep the aircraft within a reasonable range of linear movement. Moreover, fixed-wing aircraft in continuous flight can accelerate or decelerate within the allowable range but cannot hover. Appendix Table A.1 is the aircraft movement matrixes for fixed-wing aircraft, which lists all possible movement directions. The first column of the list specifies the previous flight direction. Starting from the second column are the corresponding possible directions.

3.6.2. VTOL aircraft movement pattern

Compared with fixed wing aircraft, the difference is that in this

model, the VTOL aircraft allows vertical movement, so in addition to the 8 directions of each floor, vertical climb and vertical descent are additionally added. The total number of possible movement directions is 26. As illustrated in Fig. 10, the VTOL aircraft was previously moving in direction 2-2, which means it was cruising straight and level. In the next direction, the aircraft has the option to climb, move horizontally, or descend. If it chooses to climb, it can either climb forward and upward or climb vertically to the upper layer. If it chooses to move horizontally in the current layer, it can either move forward or make a sharp turn to the nodes on the left or right. to descend, it can either descend forward and downward to the lower layer or descend vertically. It should be noted that VTOL aircraft cannot maintain their kinetic energy and continue moving when making sharp turns or vertical movements. These nodes are therefore highlighted in magenta and green in Fig. 10. The process for executing sharp turns and vertical movements in the system involves hovering as the aircraft decelerates to a velocity of 0 m per second. During sharp turns, the aircraft follows a specific rotation rate, while vertical movement does not require this step. When moving horizontally, the aircraft accelerates to the target flight speed, and when moving vertically, it accelerates to the corresponding climbing or descending speed. Appendix Table A.2 lists all selectable flight directions of the VTOL aircraft relative to the previous direction. As with the list for fixed-wing aircraft, the first column of this list is an enumeration of the previous flight direction. Starting from the second column are the corresponding movable directions.

4. Hybrid AI based methodology for UTM airspace DCB

The UTM system proposed in this article is an intelligent solution that incorporates a range of AI algorithms and data fusion inlaid methods. This approach enables efficient computing a variety of UAM vehicle flight routes in dense airspace. In addition, it could propose dynamic capacity adjustments based on different environmental factors. In conditions where airspace resources become saturated, the system can provide a global optimization solution with DCB tactical actions. To illustrate this UTM model, a mathematical formulation with hybrid AI based trajectory optimization approach is presented in Fig. 11.

4.1. Genetic algorithm

The genetic MA cycles through a number of steps, the detailed steps and the mathematical formulas of the algorithm are provided in Appendix B. The customising characteristics of the genetic MA for the adapted hybrid AI architecture are described in detail below.

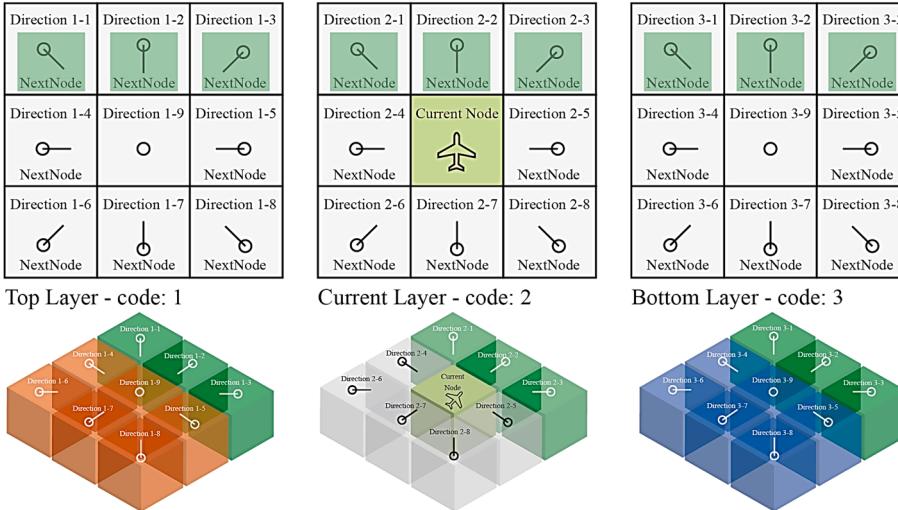
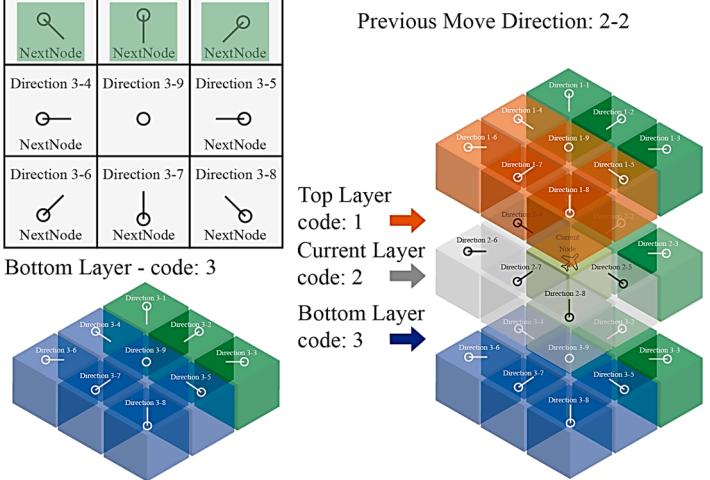


Fig. 9. Grid of possible movements in 3 dimensions for fixed-wing aircraft.

Previous Move Direction: 2-2

Top Layer code: 1 →
Current Layer code: 2 →
Bottom Layer code: 3 →



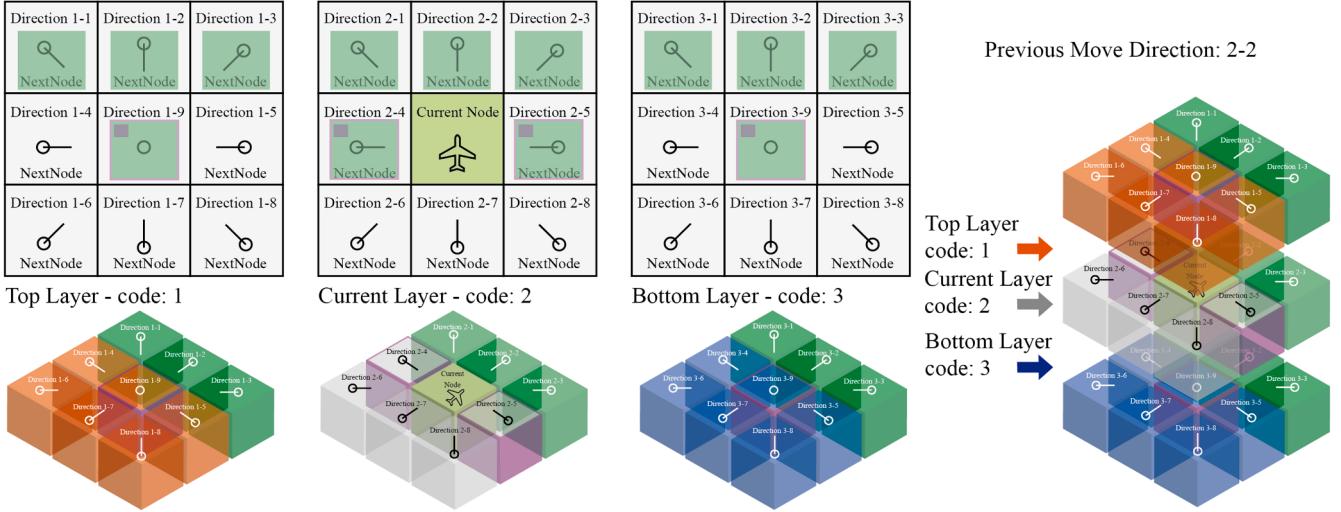


Fig. 10. Grid of possible movements in 3 dimensions for VTOL aircraft.

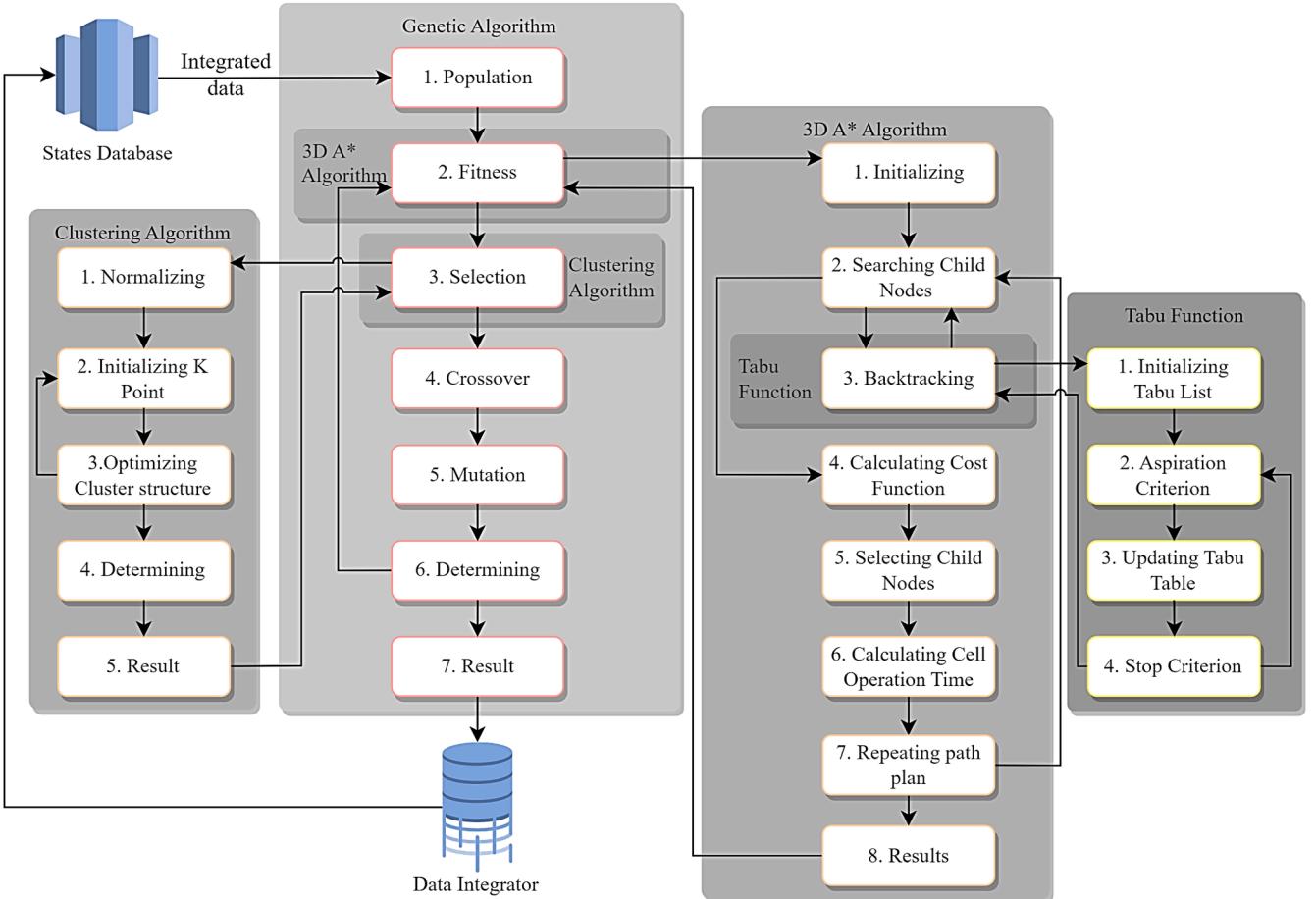


Fig. 11. Flow chart of the hybrid AI algorithm used in the UTM system.

4.1.1. Chromosome coding and composition

Fig. 12 shows the structure of the initial group. In this method, IG and NP represent the number of iteration groups and number of populations, respectively. In each individual, the number of aircraft randomly generates DCB tactical instructions. Each NP is a subset of real digital coding and representing space blocks. That is, each gene U corresponds to the aircraft in an airspace, which contains the aircraft flight mission, the

state data of the aircraft, the DCB tactical actions, and the flight route plan. The length of the real digital coding is equal to the number of aircraft in the airspace. In equivalent 1, S_i is the effective solution of this method and the corresponding chromosome coding.

$$S_i = (s_i^{U_1}, s_i^{U_2}, s_i^{U_3}, \dots, s_i^{U_N}) \quad (1)$$

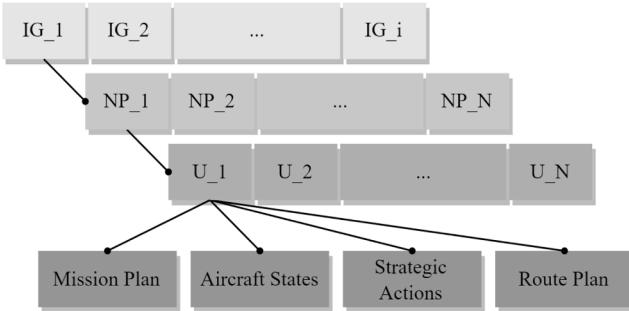


Fig. 12. MA solution coding and gene composition processes.

4.1.2. Fitness value function set

Before calculating the adaptation function set, the DCB tactical action selected by the current aircraft needs to generate a flying route of each aircraft. This function set is generated based on the global dynamic planning path algorithm of aircraft in the airspace, and the detailed method is introduced in the 3D A*algorithm section. Fitness value function set consists of three evaluation functions

- FV_1 : The sum of all aircraft evaluation values based on the 3D A*algorithm.
- FV_2 : Cell capacity overload in the global airspace.
- FV_3 : The maximum capacity of all cells in the global airspace.

The evaluation criteria for FV_2 and FV_3 are the smaller the value, the better. FV_1 's evaluation criteria are closer to the FV_1 value of the initial flight plan. The reason is that the closer the value of the FV_1 after iteration is, the closer to the initial state, the more similar operation in the airspace. This will greatly reduce the negative effects of DCB tactical action on the continuous operation of the airspace.

4.1.3. Selection mode

The Selection mode is divided into two types: weight calculation and K-means clustering. The selection of the mode depends on the number of effective NPs in the previous step of Fitness. If the number of effective NPs is less than one-fifth of the total number of NPs that need to be generated, then Mode 1 is chosen, and Mode 2 is chosen in other cases. In Mode 1, the three fitness values of the fitness value set are used as the three dimensions of the data for weight calculation. The weight ratio is 40 % for FV_2 and FV_3 , and 20 % for FV_1 . The data is then sorted based on the value. In Mode 2, the K-mean algorithm is used to group the NPs in

the current iteration. The specific algorithm process is described in the K-means clustering chapter. After the grouping is completed, weight calculation is performed based on the centre point of the cluster, using the same calculation method as that of Mode 1. The data is then sorted based on the value. Finally, no less than one-third of the total NP will be reserved as chromosomes for the next iteration.

4.1.4. Crossover operation

The optimal solution set selected in the previous step is subjected to a random chromosome crossover operation based on the value of P_c in the parameter. Before the crossover operation, the chromosome data needs to be encoded based on the type of aircraft in the airspace for classification coding. The decoding operation is performed after the interleaving operation to allow the IG dataset to be used for the next iteration. Fig. 13 shows the time code interleaving operation. The crossover mode used in this method is a single-point crossover operation that involves two steps. In the first step, the parents are grouped according to the codes, and a random gene code in each parent is selected as the crossover point. In the second step, all DCB tactical decision-making actions of both parents are exchanged starting from the intersection gene point to form the new NP solutions for the next iteration. This process is repeated until the number of newly generated NPs is equal to the value of number of populations in the parameter, and the newly generated NP solution must not have been calculated before.

4.2. Clustering algorithm

The clustering ML algorithm cycles through a number of steps, described in detail below.

Step 1: Data pre-processing

Before clustering, the input data needs to be pre-processed. This algorithm uses data normalization, and the processing flow is completed in the selection of the GA algorithm.

Step 2: Initializing K Point

Create K points as the initial centroids, set the maximum iteration number i as the algorithm termination condition, and randomly select K cluster centres as follow.

$$C = C_1, C_2, C_3, \dots, C_k \quad (6)$$

Step 3: Optimize Clustering Structure

Calculate the distance from each object to K cluster centres and assign each object to the category represented by the nearest cluster centre. After all the assignments are completed, the initial clustering result will be obtained. The cluster centre together with its assigned

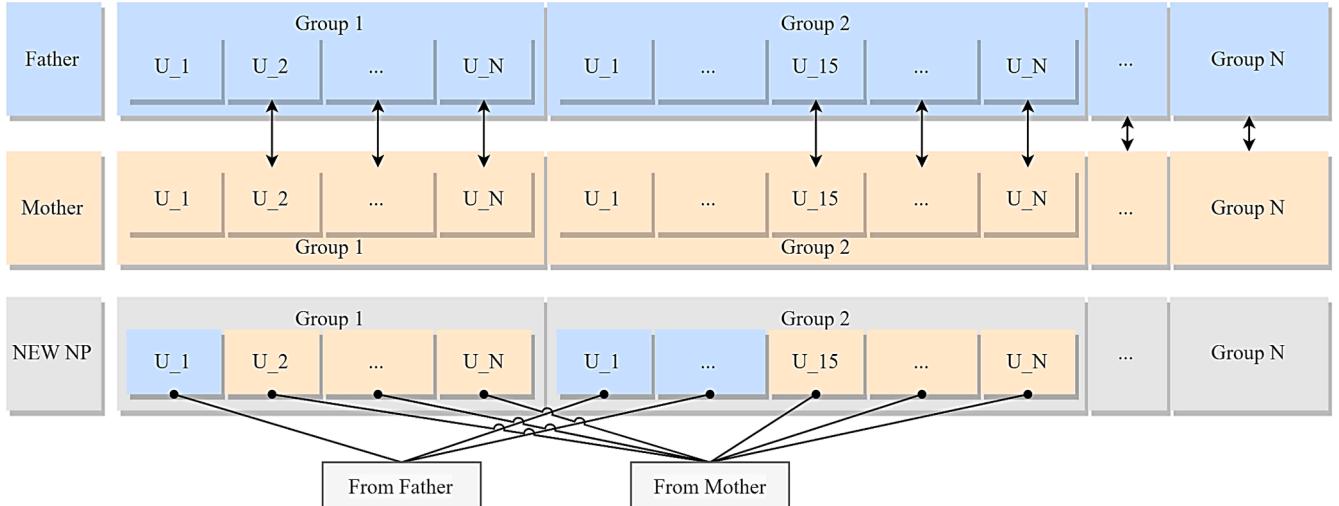


Fig. 13. Overview of GA crossover functions.

object as a class. In this algorithm, the distance is calculated as the square of the Euclidean distance, that is, each centroid is the mean of the points in the cluster. Then the cluster centre update formula is as follows.

$$J = \sum_{i=1}^k \sum_{x \in C_i} (\|x_i - \mu_i\|)^2 \quad (7)$$

$$\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} x \quad (8)$$

Step 4: Determine

After obtaining the initial clustering result, recalculate the class centre point of each class (calculate the mean value) to obtain a new cluster centre. Repeat the iterative loop until the maximum number of iterations is met.

Step 5: Result

Output the calculation result to the Selection step of the GA algorithm.

4.3. Heuristic path planning – 3D A* path planning

Step 1. Initialising

The GA algorithm's fitness function generates a sequence for processing aircraft based on their Euclidean distance and type. Using this sequence, the algorithm calculates a route plan for each aircraft in the airspace. It also obtains each aircraft's initial coordinates, target coordinates, initial flight direction, initial flight speed, DCB tactical action command, and current states.

Step 2. Searching Child Nodes

According to the current node, identify 26 movable nodes adjacent to it. When selecting the next movable node position in the current node moving direction, the aircraft type is taken into consideration. Specifically, the movement mode is divided into two types: fixed-wing aircraft movement mode and vertical take-off and landing (VTOL) movement mode. This allows for optimized path planning that takes into account the unique capabilities and limitations of each type of aircraft. The direction and path selection modes of different kinds of aircraft have been described in detail in the previous chapter. In addition, there are three types of nodes in the system that are immovable.

- Obstacle grid: The obstacle is a terrain attribute, and the node it occupies is a permanent non-movable child node in the environment.
- Temporarily immovable meshes: Temporarily immovable nodes fall into two categories. The first category is that when the system is planning multiple aircraft flight routes, the algorithm will identify the occupied sub-node based on the time dimension and avoid the node. The second type is based on the known flight nodes of the aircraft that is continuously loitering in the current airspace, and avoids occupying child nodes based on the time dimension.
- Backtracking grid: When the parent node cannot find a movable child node, the parent node will be recorded in the backtracking grid through the Tabu Function. Unlike temporary immovable nodes, nodes in the backtracking grid are immovable for a longer duration, that is, they cannot become movable nodes within a certain period of time.

In instances where aircraft are operating in a high-density environment with limited airspace, there may be cases where movable sub-nodes are unavailable due to the occupation of airspace resources. In such scenarios, the algorithm will proceed to the third step. Otherwise, it will move directly to the fifth step.

Step 3. Backtracking

When the system cannot find a valid movable child node within the current time dimension, the algorithm will perform a backtracking operation. Backtracking is divided into two steps, the backtracking node

and the tabu function. The backtracking operation is to trace the aircraft route node forward one node, so that it can avoid this node in the next calculation. This method effectively avoids the generation of invalid solutions. Fig. 14 demonstrates the application of a backtracking algorithm in a 3D path display and a route list display. The scenario features two fixed-wing UAS with pre-planned routes, a group of obstacles, and a group of temporarily closed cells. The mobility model for aircraft 3 is designed for fixed-wing aircraft, with only nine nodes permitted for movement in the fifth time unit. Unfortunately, two of these nodes are temporarily closed, five are obstacles, and two are already occupied by aircraft 1 and aircraft 2. This poses a challenge to route planning. To overcome this challenge, the algorithm leverages backtracking based on time units. It reverts to the third time unit as the parent node and replans the path of the fourth time unit. The algorithm also adds the path information from the previous fourth time unit to the Tabu Function to avoid retracing the same path.

The tabu function consists of four steps:

1. Initial Tabu List: records the parent node in the backtracking function into the *Tabu List*.
2. Aspiration Criterion: sets the tabu length (*TabuL*) of the node, and the value of tabu length is 10 nodes, which means that the node cannot be selected when searching for the next 10 nodes.
3. Updating Tabu Table: updates tabu length value in the Tabu List.
4. Stop Criterion: determines whether the output condition is met (i.e., if the taboo length value achieves the target).

Step 4. Cost Function F

After searching for a node that can be moved, the algorithm computes the cost function $f(ns)$ for that node, shown in Eq. (9).

$$f(ns) = g(ns) + h(ns) + d(ns) \quad (9)$$

$(ns = 1, 2, 3, \dots, \text{steps of path})$

The basic movement grid has 26-way possible movements, so two calculation methods are used when calculating the distance: Euclidean and Manhattan distances. The spatial coordinate system of the model adopts three-dimensional coordinates, which is that x stands for latitude, y for longitude and z for altitude. $g(ns)$ represents the cost from the starting point to the current node, the algorithm is based on Euclidean distances as follows:

$$g(ns) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (10)$$

$\text{Node}_{\text{initial}} = (x_1, y_1, z_1)$
 $\text{Node}_{\text{current}} = (x_2, y_2, z_2)$

$h(ns)$ represents the estimated cost from the current node to the target node. The related algorithm is based on Manhattan distance as follows:

$$h(ns) = |x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2| \quad (10a)$$

$\text{Node}_{\text{current}} = (x_1, y_1, z_1)$
 $\text{Node}_{\text{target}} = (x_2, y_2, z_2)$

$d(ns)$ represents the estimated cost from the current node to the target node. The related algorithm is based on Euclidean distances as follows:

$$d(ns) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (11)$$

$\text{Node}_{\text{current}} = (x_1, y_1, z_1)$
 $\text{Node}_{\text{target}} = (x_2, y_2, z_2)$

After the calculation is completed, the algorithm will record all movable nodes in the Openlist. Also record $f(ns)$, $g(ns)$, $h(ns)$ and $d(ns)$ in Openlist_Cost.

Step 5. Child node selection

Select the minimum value of Openlist_Cost, and find the corresponding node in the Openlist as the parent node for the next node calculation. At the same time, record all the information of the node into

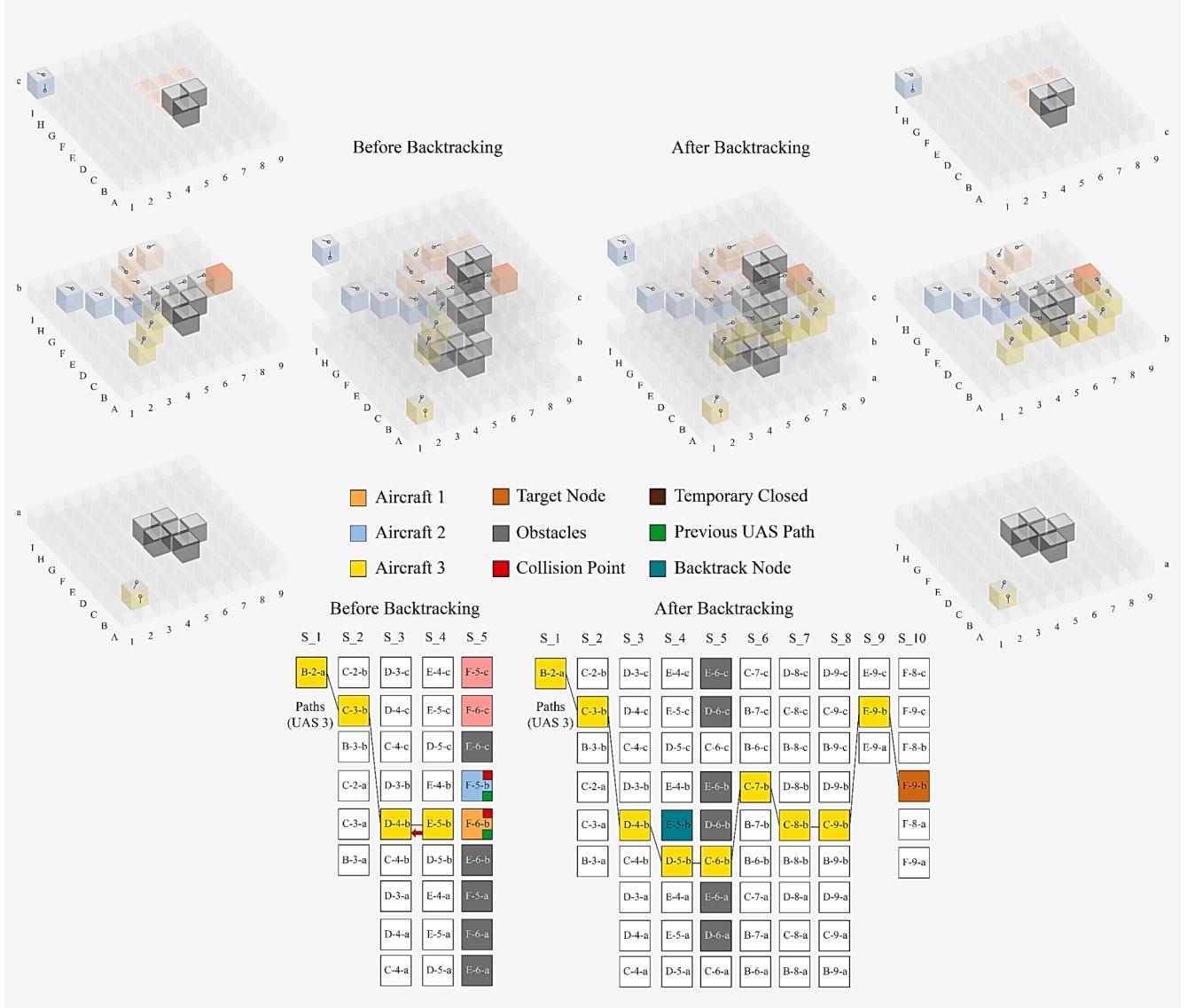


Fig. 14. Backtracking function process in 3D and diagram views.

Closelist and Closelist_Cost.

Step 6. Calculating Cell Operation Time

Calculate the flight time of the aircraft in this node. The calculation of flight time is based on the aircraft's flight speed, flight distance, and aircraft's special flight mode. In this step, the algorithm will determine whether the aircraft is still in the current space node or has moved to the

aircraft dynamic flight speed:

$$\text{if } S_{Dynamic} < S_{target} \Rightarrow \text{cruise}\backslash\text{Climb}\backslash\text{descent} \begin{cases} 1 \quad S_{Dynamic} = S_{Dynamic} + S_{acceleration} \\ 0 \quad S_{Dynamic} = S_{Dynamic} - S_{deceleration} \end{cases} \quad (12)$$

$$\text{if hover rotation} \begin{cases} 1 \quad S_{Dynamic} = ((S_{Dynamic} - S_{deceleration}), S_{rotation}, (S_{Dynamic} + S_{acceleration})) \\ 0 \quad S_{Dynamic} < S_{target} \Rightarrow \text{cruise}\backslash\text{Climb}\backslash\text{descent} \begin{cases} 1 \quad S_{Dynamic} = S_{Dynamic} + S_{acceleration} \\ 0 \quad S_{Dynamic} = S_{Dynamic} - S_{deceleration} \end{cases} \end{cases} \quad (13)$$

next node based on the time dimension. Among them, the flight speed includes dynamic flight speed, target flight speed, initial flight speed, climbing speed, descent speed, acceleration speed and deceleration speed; the special flight mode is hovering rotation speed. Eq. (12) is the fixed-wing aircraft dynamic flight speed, whereas Eq. (13) is the VTOL

Step 7. Repeat the path planning

After entering the next node, the algorithm returns to the second step to find a new movable node until the destination node is added to the Closelist. When the aircraft route nodes are formed, the algorithm will record all nodes in the Closelist_Path in order of time dimension, and

calculate the sum of $f(ns)$ of the current aircraft, where NS is the total number of steps of the UA path. The relative equation is:

$$F(n) = \sum_{i=1}^{NS} f(ns) \quad (i = 1, 2, 3, \dots, NS) \quad (14)$$

Step 8. Result

Finally, the required data is passed to the GA algorithm for the next operation. The data delivered includes converting the generated aircraft route plan into a temporarily immobile grid based on the time dimension.

5. Verification of the proposed UTM functionality

To perform the verification of the proposed system and algorithms, we developed a 3D virtual environment that simulates the urban low-altitude airspace structure. Fig. 15 presents the virtual environment that covers a zone located in the north of Melbourne CBD. This virtual zone spans 750 m in latitude, 250 m in longitude, and 70 m in altitude. The virtual environment enabled us to test the UTM system's capability to manage UAM vehicles in a complex urban airspace scenario.

5.1. Simulation environment and algorithm parameters setup

The development, testing and evaluation of the prototype system is based on MATLAB 2021b, running on Windows 11 operating system, 2.3 GHz Base 14-core i7-12,700 h CPU and 32 G RAM, using the CPU parallel computing function of Parallel Computing Toolbox to accelerate the calculation results.

5.1.1. Simulation design and parameters setting

The mock-scenario testing experiment utilizes several crucial parameters. The UTM DCB time unit has a step size of 1 second, while the airspace's smallest spatial unit is a cell measuring 10 m x 10 m x 10 m. These cells are arranged horizontally in 75 columns and vertically in 25 columns, creating a total of 7 layers. The simulated scene test involves a wide range of variables, which change randomly with each simulation. These variables can be broadly categorized into overall airspace variables and individual UAS (Unmanned Aircraft System) variables. Airspace variables include weather state and CNS (Communications, Navigation and Surveillance) performance state, with weather status divided into sunny, rainy and windy categories, and the CNS performance states can be Good or Fair. Table 2 presents the range of influence of cells in the airspace in different states. These variables are randomly selected by the system.

Table 3 shows the corresponding flight performance values of fixed-wing UAS and vertical take-off and landing (VTOL) UAS respectively.

Table 2

Cell capacity effects by airspace states.

States Capacity Effects	Airspace Weather			UAS CNS Performance		
	Sunny 100 %	Rainy 90 %	Windy 80 %	Good 100 %	Poor 85 %	

Table 3

UAS flight performance.

Weight class	Fixed Wing		VTOL			
	Small	Medium	Very Small	Small	Medium	Large
Min flight speed (m/s)	5	4	4	4	3	2
Max flight speed (m/s)	6	5	5	5	5	4
UAS capacity in cell	35 %	50 %	15 %	25 %	35 %	50 %
Rotation time (s)			4	4	4	8
Climb speed (m/s)	5	4	5	4	3	3
Descent speed (m/s)	5	5	4	4	3	2
Accelerate speed (m/s)	3	3	5	4	4	2.5
Decelerate speed (m/s)	4	4	5	5	4	3

The individual UAS variables for performing point to point missions are as follows:

- The categories of all UAS are randomly generated, and the categories are divided into fixed wing and VTOL.
- The initial coordinates of all UAS are randomly generated.
- The destination coordinates for all UAS are randomly selected based on the destination list.
- The initial flight speed is generated randomly within a range that spans from the minimum to the maximum flight speed. The initial flight direction of all UAS is randomly generated.
- The CNS performance status of all UAS is randomly assigned. When the status is abnormal, the UAS occupancy rate will increase by 20 %.
- The battery status of all UAS is randomly generated, ranging from 30 % to 80 %.

It's worth noting that the obstacles in the three-dimensional environment are quantified and will remain constant throughout the



Fig. 15. Three views and 3D depiction of the adopted simulation environment.

simulation test. Additionally, the status and flight paths of UAS moving in loiter mode to perform surveillance missions in the airspace will not change randomly during the simulation. Also, to simplify the scale of the system calculation, the flight speed of the UAS uses only ground speed, not airspeed.

5.1.2. Algorithm settings

The genetic algorithm's parameters are the number of iterations i , population size NP , crossover probability P_c , and mutation rate P_m . For achieve a balance between the convergence effect and the algorithm's running time, given the limitations imposed by the performance of the physical hardware, in this study, we set the value of i to 100 and the value of NP to the number of UAS in the airspace. We chose a P_c value of 0.7 and a P_m value of 0.15. It should be pointed out that the numerical setting of the number of iterations in the GA algorithm is very low. Finding an effective optimization solution within a limit iterative cycle poses a daunting challenge to the effectiveness of the system. The clustering algorithm's parameters include the number of centroids K and the number of iterations i . To account for the simulation experiment environment and hardware computing power, we set K to one-tenth of the total number of UAS and rounded up to the nearest integer. We also set the number of iterations i to 500 and output the iteration result when this condition is satisfied.

5.1.3. Comparison methodology and performance evaluation

To test and compare the effectiveness and feasibility of the UTM system, we place it in simulated environments with different operating densities. The airspace operating status is divided into three categories:

- Low-density operation, where the total number of UAS in the current time dimension does not exceed 100.
- Medium-density operation, where the total number of UAS in the current time dimension is greater than 100 but less than 150.
- High-density operation, where the total number of UAS in the current time dimension is more than 150.

Furthermore, we defined Key Performance Indicators (KPIs) for evaluate the quality of the UTM system.

- Success rate of the algorithm to obtain the effective solution of DCB: This KPI measures the effectiveness of the genetic algorithm in

finding a feasible solution to the DCB problem. A higher success rate indicates that the algorithm is more reliable and efficient.

- Comparison of the initial state of the airspace and the state after DCB, which evaluates the improvement in airspace utilization and congestion after the implementation of the DCB. It measures the reduction in overloaded cells and the increase in cell capacity utilization, which can be an indicator of the UTM system's ability to optimize airspace utilization. Therefore, we evaluate the actual performance of the algorithm by the reduce rate of overload cell, the occupancy reduces rate of the cell capacity, and the change rate of the airspace operating time.

5.2. Verification results and discussions

5.2.1. Comparison of different operation densities

We conducted 100 random scenarios for each of the three airspace operating density states. Fig. 16 presents the detailed comparison results of the UTM system's KPIs. Overall, in the low-density operating state scenario, our UTM system demonstrated a high success rate of 93 % in generating effective DCB solutions, resulting in the complete elimination of the overload state of Cells in the airspace. The average cell overload reduction rate in the simulated scene reached 99.74 %, and even the lowest value reached 97.49 %. This shows that even for scenarios where no efficient solution is obtained, the better DCB solution still manages to eliminate at least 97.49 % of the airspace cell overload conditions over the full cycle time of airspace operation. Furthermore, our system successfully reduces the maximum occupancy of a single cell by 25 % to 194.12 %, with an average reduction of 71.76 %, in this density state scenario. This data corresponds to the ratio of the maximum occupancy rate of a single cell in the airspace before and after optimization to the actual maximum capacity of the cell. On average, the overall run time for airspace showed only a slight change of 9 %, while the scenario with the greatest reduction in overall run time showed a 46.97 % improvement. The scene with the largest increase in overall runtime increased by only 26.25 %. It should be pointed out that the purpose of optimizing the airspace operating time is not to shorten the airspace operating time, but to reduce the fluctuation and change of the airspace operating time before and after optimization. Excessive increase and decrease of the operating state of the airspace will inevitably affect the subsequent operating state. Therefore, maintaining airspace operation time is the optimization goal of the system. These results demonstrate the

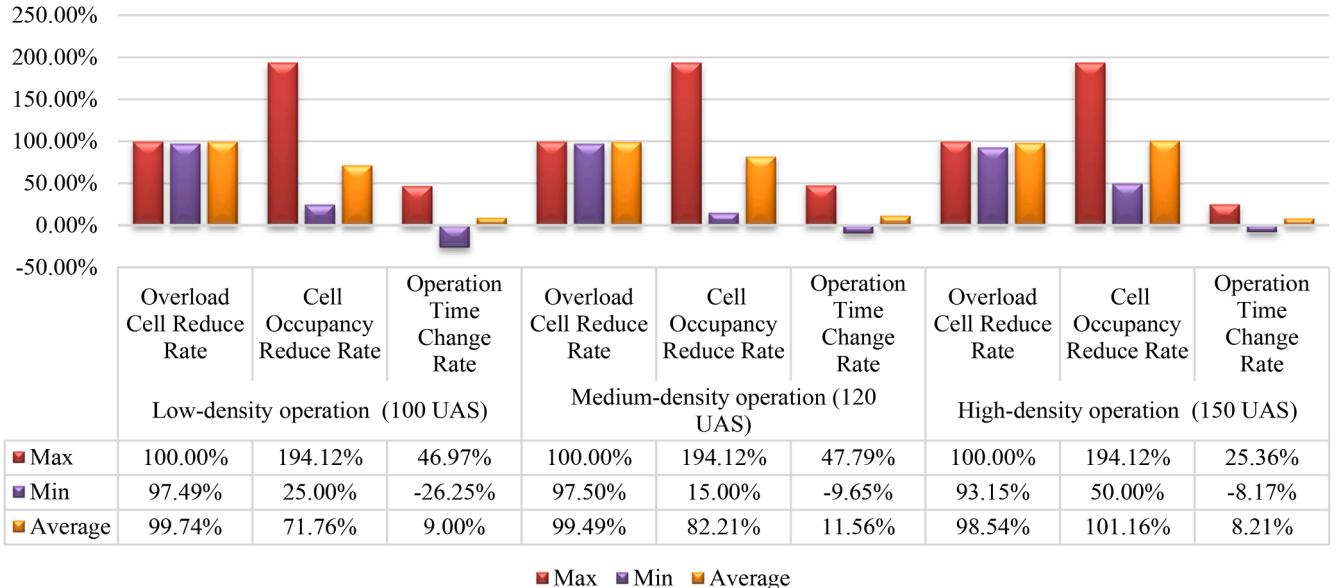


Fig. 16. Overview of optimization results for different operating densities.

effectiveness of our UTM system in managing the low-density operating state scenario.

As the density of UAS in the airspace increases, the probability of the system obtaining an effective DCB solution within a set number of iterations will slightly decrease in medium-density and high-density operating test scenarios. The success rates were 86 % and 80 %, respectively. The average value of eliminating overloaded cells in the airspace will slightly decrease to 99.49 % and 98.54 % as the operating density increases. The optimal value of the maximum occupancy rate of a single cell also increased to 82.21 % and 101.16 % correspondingly. It fully shows that in more intensive operation scenarios, the overall overload of the airspace will increase, and the overload of each cell will also increase. The optimization rate of the system to the state of the airspace is also higher. The value of the airspace running time also keeps floating in the range of about 10 %. This also shows that the hybrid AI algorithm architecture can effectively solve the congestion problem in more congested airspace. The results demonstrate the effectiveness of the UTM system based on the hybrid AI algorithm architecture in generating DCB solutions. By changing the DCB tactical action, the algorithm effectively reduces the number of cell overloads in the airspace, lowers the maximum capacity occupancy of cells in the time-based dimension, and optimizes the global running time of the airspace.

5.2.2. Specific operational scenario case study

We conducted data analysis and legend interpretation for one of our simulated scenarios. Table 4 presents the basic status and parameters of the No. 63 virtual airspace scene, which was characterized by rainy weather and weak overall CNS performance. As a result, the maximum carrying capacity of a single cell was reduced to 76.5 %. After 39 iterations, our system successfully found DCB tactical actions for all 120 UAS in the airspace, which met the no-overload state requirements. Table 5 compares the initial state of the scene with the optimized DCB solution. Initially, within 347 s of the overall operation, we observed a total of 91 overload points in the airspace, with the maximum capacity of a single cell exceeding the cell carrying capacity limit by 43.5 %. However, after optimization, all overload points disappeared within 356 s of airspace operation, and the maximum capacity occupancy was reduced to 75 %, which was below the capacity limit of 76.5 %. These results demonstrate the effectiveness of the DCB solution in optimizing the UAS operations and preventing cell overloading in the airspace.

Figs. 17, 18, and 19 depict the iterative trends of Fitness Value 1, 2 and 3 (FV1, FV2, FV3) when the simulation scenario searches for a global DCB tactical action solution. Since each iteration generates a set of DCB solutions, the values are displayed within a range (depicted as grey bars). These values decrease with iterative optimization, especially FV2, whose ultimate goal is to reduce to 0 within a limited number of iterations. The actual verification result aligns with the original design idea.

FV1, represented in Fig. 17, is the sum of all UAS evaluation values based on the 3D A* algorithm, which means the sum of all UAS step number function values in the airspace. During the iteration period, FV1 decreased by 27.8 % on average, with a drop of 40.2 % in the extreme case. This indicates that the UAS flight path is optimized based on DCB tactical actions, reducing the total number of UAS flight steps in the airspace.

FV2, shown in Fig. 18, represents the number of cell capacity

Table 4
States of simulation airspace no. 63.

Simulation Scene Code	63
UAS Number	120
CNS Performance	Poor
Weather State	Rainy
Cell Tolerance	76.50 %
Iteration Times	39

Table 5
Comparisons of airspace no. 63 states before and after the optimization.

Initial	Overload Cells	91
	Max Cell Overload	120 %
	Operation Time (s)	347
Optimized	Overload Cells	0
	Max Cell Overload	75 %
	Operation Time (s)	356

overloads in the global airspace. In the initial stage, there were 91 overload points during the entire operation cycle. After one iteration, the model provided a solution set of DCB tactical actions with a maximum of 73 and a minimum of 16 overload points. The fitness value gradually decreased as the model iterated until a DCB tactical action solution without an overload point was found.

Fig. 19 displays FV3, which is the maximum occupancy of a single Cell in the global airspace. This function's conditional value changes as the simulated environment changes, as the maximum capacity drops with the weather state and CNS performance state. Therefore, the base value of this function varies in each simulated scenario. Since a large number of DCB solutions are generated in one iteration, the optimization trend may fluctuate. In the initial state, the maximum value was 120 %, but after the first iteration, the maximum value increased to 152 %, the minimum value decreased to 100 %, and even the mean value reached 126 %, which was still 6 % higher than the initial state. Even in the last iteration where a DCB solution was found, the maximum value in the function interval remained the same as the initial state. Therefore, FV3 will decrease as the solution is found, but it will not exhibit a clear downward trend like FV1 and FV2. This highlights the optimization tendency and goal of the model, prioritizing the overall optimization of the airspace over the optimization of a single Cell. However, an effective global airspace DCB tactical action solution can only be achieved when both are optimized to reach the optimization goal. Fig. 20 illustrate the flight route of all UAS in the simulated airspace for this scenario. All 120 UAS in the airspace follow the non-overload route calculated by the system. The green dots are the initial coordinates of each UAS. In addition, the figure also shows the state that multiple drones appear in one cell in the same time dimension during the full operation cycle. These cells are represented in dark blue.

Figs. 21 and 22 depict the overall status of the simulated airspace operation in the 1st and 30th seconds, respectively. In the initial state (1st second), all 120 UAS appear at different coordinates, with their flight targets being the initial destination of the Cell, which is marked in red, or the alternative destination selected by the DCB action, which is displayed in light blue. By the 30th second, some UAS have reached their destinations and are no longer present in the airspace. During this time dimension, if there are multiple drones in the airspace cell without capacity overload appears, the cell is displayed as a dark blue cube. At this point, both UAS No. 54 and No. 114 are present in this cell simultaneously.

Figs. 23 and 24 illustrate the 4D trajectory of UAS No. 54 and UAS No. 114, respectively, within the airspace. During the 30th second of operation, the two UAS briefly appear in the same cell, as shown in Fig. 22. UAS No. 54 is a small-class fixed-wing drone with a processing queue of No. 12, an initial flight speed of 4 m per second, and 72 % remaining battery power, providing unlimited options for DCB tactical actions. The DCB action selection code for UAS No. 54 is 4, resulting in a 20 % reduction in flight speed, while the CNS performance status of the UAS is good, ensuring that the cell capacity occupancy rate remains at 35 %. UAS No. 114, on the other hand, is a very small class VTOL drone with a processing queue of No. 95, an initial flight speed of 5 m per second, and 72 % remaining battery energy, providing unlimited options for DCB tactical actions. The DCB action selection code for UAS No. 114 is 5, which involves hovering at the initial coordinate for 10 s, while the CNS performance status of the UAS is good, ensuring that the cell capacity occupancy rate remains at 15 %. Despite the limited capacity of

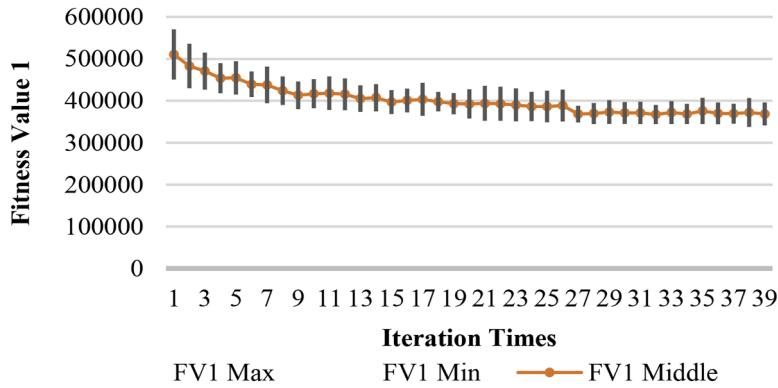


Fig. 17. Iterative optimization results of FV1.

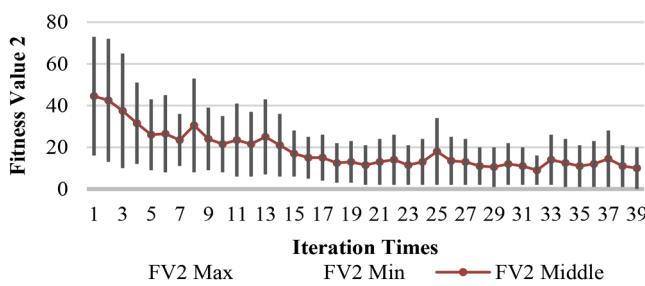


Fig. 18. Iterative optimization results of FV2.

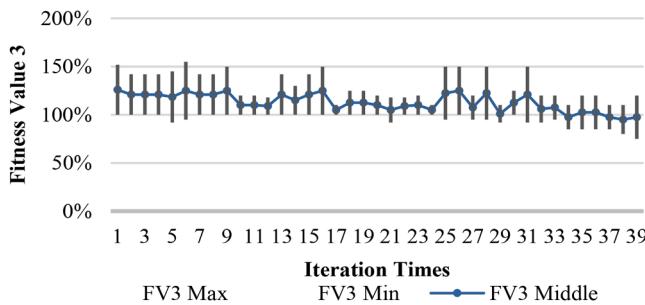


Fig. 19. Iterative optimization results of FV3.

the cell due to the overall state of the airspace, with a maximum capacity of 76.5 %, the total capacity of the two UAS is only 50 %, and thus the cell is not overloaded. These trajectories demonstrate the effectiveness of the model in optimizing airspace operations by providing UAS with options for DCB tactical actions. UAS can adjust speed and take alternative actions to avoid airspace congestion and overload, resulting in smoother and more efficient operations.

The case study demonstrates the remarkable capability of the system to achieve an efficient global DCB tactical action solution in a high-density airspace environment, where multiple types of UAS with complex mission types operate in the presence of obstacles and uncertain factors. However, despite its success, the system still faces challenges, including long calculation time. It takes an average of 30 to 40 min to obtain an effective DCB solution for a small-scale scene, which is primarily due to the high fidelity of the 4D trajectory. To overcome this challenge, the system can leverage parallel computing, GPU acceleration, and cloud computing services. Additionally, deep reinforcement learning algorithms can be implemented to simulate the labelled optimization results obtained by the system, which can enhance the utility of the UTM system. Further research and development in this area are necessary for the UTM system to reach its full potential.

5.3. Limitations

The results and findings of this work should be evaluated in light of some limitations. To expedite the verification of the DCB method based on the hybrid AI algorithm architecture, several calculation parameters in the model were simplified and assumed. One significant simplification was not to adopt high-fidelity three-degree-of-freedom (3DOF) or

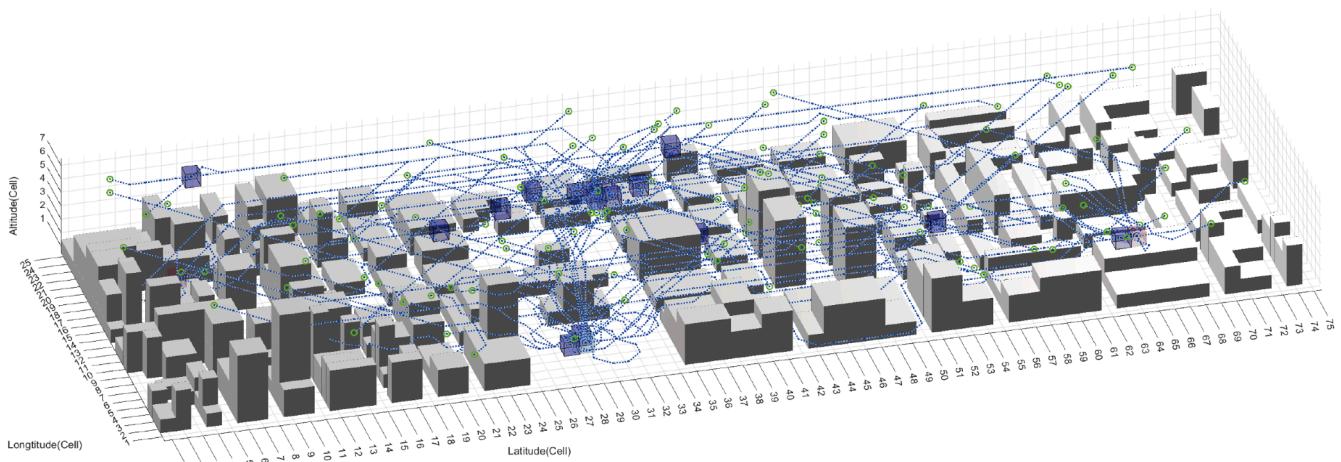


Fig. 20. Overall state of simulated airspace including all aircraft.

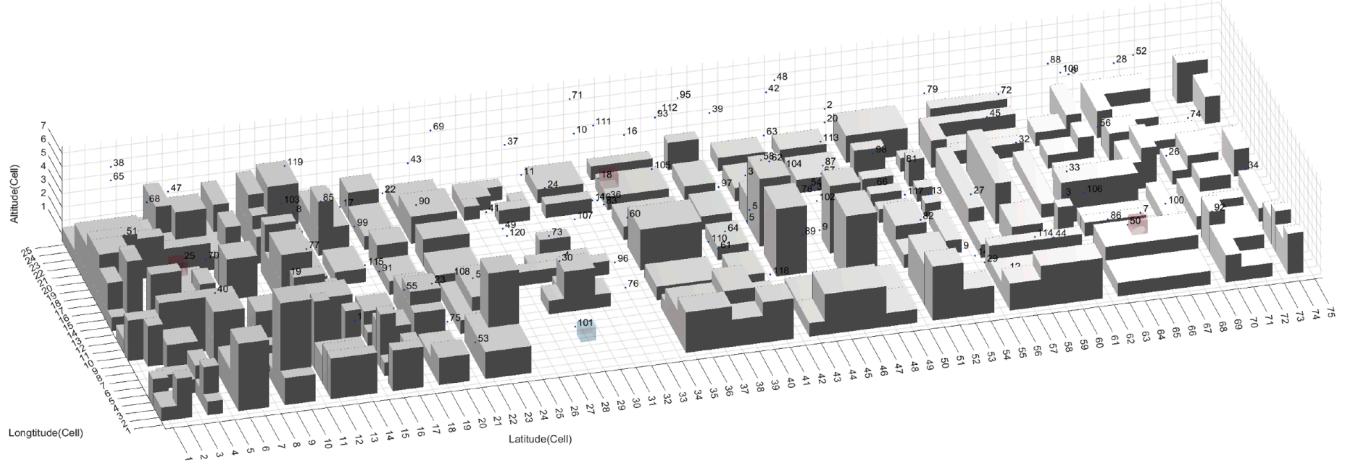


Fig. 21. Overall state of simulated airspace in the 1st second.

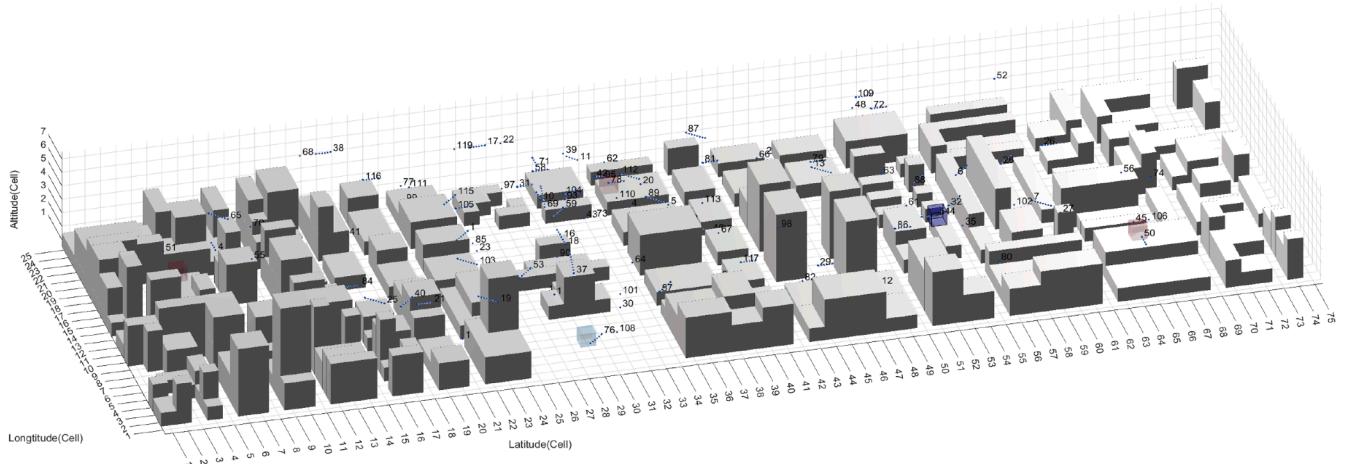


Fig. 22. Overall state of simulated airspace after 30th second.

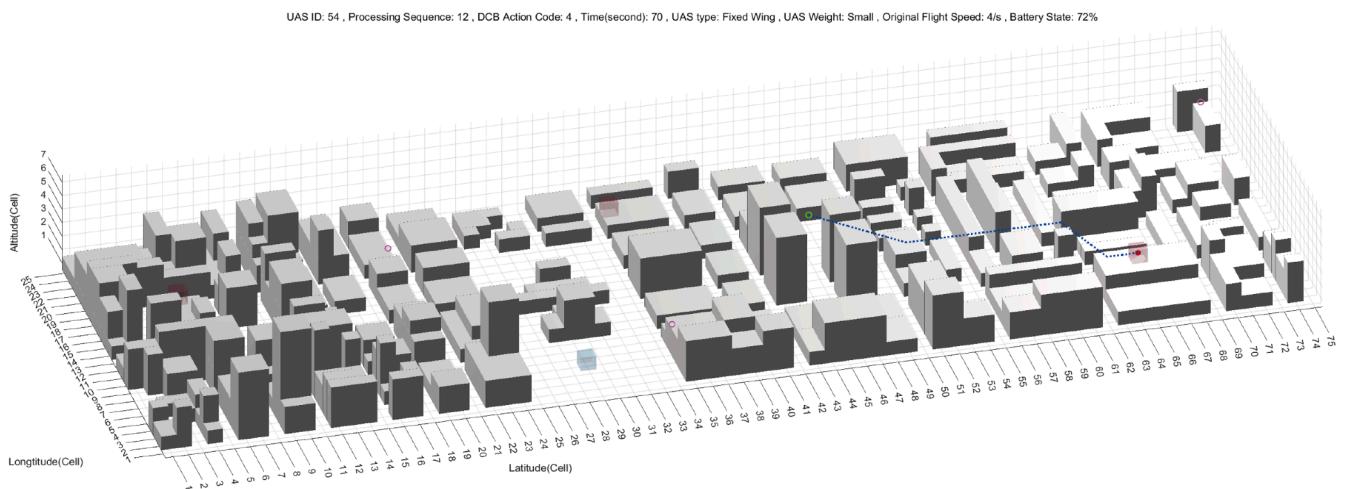


Fig. 23. Trajectory of UAS No. 54 in the simulated airspace scenario.

six-degree-of-freedom (6DOF) flight dynamics models for fixed-wing and VTOL aircraft. These simplifications were necessary due to the rather limited current maturity of existing models and some limitations in computing power. Although the simplified model does not capture all the dynamic behaviours and interactions that occur during actual

operation, it nonetheless captures the macroscopic progression of vehicles over their journeys and is sufficient for initial testing and validation of the DCB approach. Moreover, this simplification does not impact the core functionality of the genetic MA used in the research. Incorporating more accurate and detailed flight dynamics models would

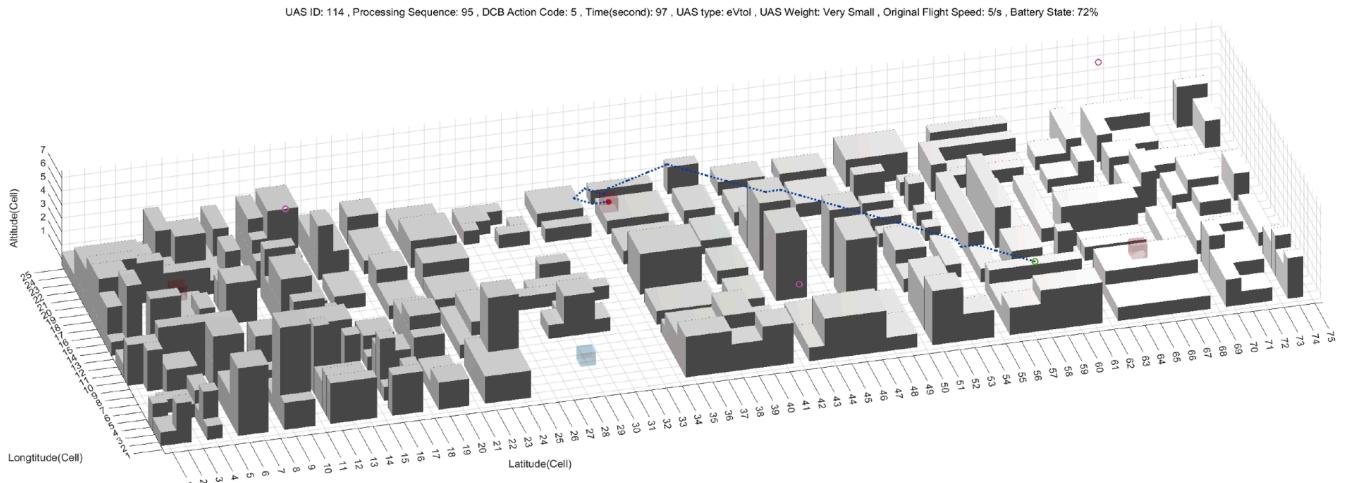


Fig. 24. Trajectory of UAS No. 114 in the simulated airspace scenario.

enhance the realism and precision of the simulation results, but it does not affect the effectiveness of the DCB method.

This study also assumed parameters related to occupancy and capacity, including the quantification of the impact of weather on airspace capacity, the quantification of the impact of CNS performance on airspace capacity, and the quantification of CNS performance on the occupancy of UAM vehicles in the airspace. These assumptions, while necessary for modelling purposes, may not accurately reflect the real-world dynamics where the impact of weather and CNS performance can be highly variable and context-dependent. Furthermore, the model does not fully account for the effects of variable weather and atmospheric boundary layer (ABL) phenomena such as local gusts, building wakes, and turbulence, which can significantly impact the safe access to certain airspace cells. These ABL phenomena introduce localized and transient disruptions in the airspace, creating unpredictable variations in capacity and vehicle performance. For instance, local gusts involve sudden changes in wind speed and direction, which in conjunction with building wakes and turbulence can create hazardous conditions that are not uniformly distributed across the airspace. The degradation in CNS performance might not uniformly affect all areas within the airspace, and the impact of severe weather conditions could be more localized or transient than the model assumes. Additionally, the study assumes that the occupancy rate of UAM vehicles is directly influenced by CNS performance, increasing by a fixed percentage under poor CNS conditions. The relationship between CNS performance and vehicle occupancy is likely to be more complex, influenced by factors such as the spectral band and type of technology used, the level of redundancy of CNS systems, and the operational protocols of individual UAM vehicles. These simplified assumptions could lead to both overestimation or underestimation of airspace capacity and vehicle occupancy in certain scenarios, affecting the accuracy and robustness of the UTM system's performance predictions. It should be noted that these assumptions result in empirical quantitative values, the accuracy of which can significantly affect the performance and effectiveness of the UTM system, but the assumed values themselves do not affect the functionality of the DCB method.

6. Conclusions

In this article, we proposed an intelligent UTM system based on a hybrid AI algorithm, which supports high-density operation of UAM vehicles in future urban low-altitude airspace through suitable DCB functionalities. The proposed method combines 4D trajectory planning based on three-dimensional space and time descriptors and realizes the balance between demand and capacity in the airspace, so as to ensure the safety of flight operations in low altitude airspace. This is achieved

by integrating both metaheuristic (genetic) and machine learning algorithms, which evaluate a large array of possible solutions and therefore offer a very significant potential for addressing the complexity of all UAM environments. To perform a verification in representative conditions, we developed a virtual environment and carried out a certain set of simulation case studies in different weather and CNS system performance conditions. The results demonstrate that the system can provide DCB solutions without capacity overload or flight path violations for low-altitude airspace environments across various operating densities within a limited number of iterations on an experimental scale. Notably, even in high-density operating environments, the DCB solution effectively reduces airspace overload by at least 93.15 %. Future research will aim to gradually eliminate some of the current method's limitations, further refine the quantitative parameters assumed in the system, and develop corresponding components to accurately analyse real-life data once available. This will ensure a higher fidelity in terms of weather conditions, CNS performance, and other key parameters which were empirically assumed in this study. In addition, further research should address the following aspects, which were intentionally omitted in this initial study:

- the effect of wake turbulence separation for both fixed-wing and VTOL vehicles: while simplified models for fixed-wing aircraft wakes have been in existence for a few decades, the modelling of downwash by multirotor aircraft is still subject to investigation and therefore it is not yet mature for integration [67].
- airspace cell partitioning with different and time-variable dimensions, matching variable CNS performance and local environmental conditions: while proposed and conceptualised, the full mathematical development and practical realization of this concept is still ongoing [66].
- the cognitive performance of human UTM operators and UAS pilots, which may reduce the overall capacity even if their involvement is less tactical than in conventional ATM: research on this issue is ongoing and the greatest promise is held by cognitive-adaptive interfaces and interactions such as CHMI2 [68].

In terms of algorithmic improvements, we will further explore the most promising reinforcement learning (RL) techniques to develop faster and more efficient UTM decision-making functions and exploit larger amounts of labelled data generated by the MA to support the learning and training of the RL algorithm to improve overall system performance.

CRediT authorship contribution statement

Yibing Xie: Conceptualization, Data curation, Investigation, Methodology, Software, Visualization, Writing – original draft. **Alessandro Gardi:** Conceptualization, Methodology, Supervision, Writing – review & editing. **Man Liang:** Supervision, Writing – review & editing. **Roberto Sabatini:** Conceptualization, Methodology, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Appendix A. Aircraft Movement Matrixes

Table A.1 and Table A.2

Table A.1

Full list of possible movements in 3 dimensions for fixed-wing aircraft.

Previous Direction of Movement	Movable Direction									
1-1	1-4	1-1	1-2	2-4	2-1	2-2	N/A	N/A	N/A	N/A
1-2	1-1	1-2	1-3	2-1	2-2	2-3	N/A	N/A	N/A	N/A
1-3	1-2	1-3	1-5	2-2	2-3	2-5	N/A	N/A	N/A	N/A
1-5	1-3	1-5	1-8	2-3	2-5	2-8	N/A	N/A	N/A	N/A
1-8	1-5	1-8	1-7	2-5	2-8	2-7	N/A	N/A	N/A	N/A
1-7	1-8	1-7	1-6	2-8	2-7	2-6	N/A	N/A	N/A	N/A
1-6	1-7	1-6	1-4	2-7	2-6	2-4	N/A	N/A	N/A	N/A
1-4	1-6	1-4	1-1	2-6	2-4	2-1	N/A	N/A	N/A	N/A
2-1	2-4	2-1	2-2	1-4	1-1	1-2	3-4	3-1	3-2	3-2
2-2	2-1	2-2	2-3	1-1	1-2	1-3	3-1	3-2	3-3	3-3
2-3	2-2	2-3	2-5	1-2	1-3	1-5	3-2	3-3	3-5	3-5
2-5	2-3	2-5	2-8	1-3	1-5	1-8	3-3	3-5	3-8	3-8
2-8	2-5	2-8	2-7	1-5	1-8	1-7	3-5	3-8	3-7	3-7
2-7	2-8	2-7	2-6	1-8	1-7	1-6	3-8	3-7	3-6	3-6
2-6	2-7	2-6	2-4	1-7	1-6	1-4	3-7	3-6	3-4	3-4
2-4	2-6	2-4	2-1	1-6	1-4	1-1	3-6	3-4	3-1	3-1
3-2	3-4	3-1	3-2	2-4	2-1	2-2	N/A	N/A	N/A	N/A
3-2	3-1	3-2	3-3	2-1	2-2	2-3	N/A	N/A	N/A	N/A
3-3	3-2	3-3	3-5	2-2	2-3	2-5	N/A	N/A	N/A	N/A
3-5	3-3	3-5	3-8	2-3	2-5	2-8	N/A	N/A	N/A	N/A
3-8	3-5	3-8	3-7	2-5	2-8	2-7	N/A	N/A	N/A	N/A
3-7	3-8	3-7	3-6	2-8	2-7	2-6	N/A	N/A	N/A	N/A
3-6	3-7	3-6	3-4	2-7	2-6	2-4	N/A	N/A	N/A	N/A
3-4	3-6	3-4	3-1	2-6	2-4	2-1	N/A	N/A	N/A	N/A

Table A.2

Full list of possible movements in 3 dimensions for VTOL aircraft.

Previous Direction of Movement	Movable Direction															
1-1	1-4	1-1	1-2	N/A	N/A	2-6	2-4	2-1	2-2	2-3	0	N/A	3-9	N/A	N/A	N/A
1-2	1-1	1-2	1-3	N/A	N/A	2-4	2-1	2-2	2-3	2-5	0	N/A	3-9	N/A	N/A	N/A
1-3	1-2	1-3	1-5	N/A	N/A	2-1	2-2	2-3	2-5	28	0	N/A	3-9	N/A	N/A	N/A
1-5	1-3	1-5	1-8	N/A	N/A	2-2	2-3	2-5	2-8	27	0	N/A	3-9	N/A	N/A	N/A
1-8	1-5	1-8	1-7	N/A	N/A	2-3	2-5	2-8	2-7	26	0	N/A	3-9	N/A	N/A	N/A
1-7	1-8	1-7	1-6	N/A	N/A	2-5	2-8	2-7	2-6	24	0	N/A	3-9	N/A	N/A	N/A
1-6	1-7	1-6	1-4	N/A	N/A	2-8	2-7	2-6	2-4	21	0	N/A	3-9	N/A	N/A	N/A
1-4	1-6	1-4	1-1	N/A	N/A	2-7	2-6	2-4	2-1	22	0	N/A	3-9	N/A	N/A	N/A
2-1	2-6	2-4	2-1	2-2	2-3	N/A	1-4	1-1	1-2	3-4	3-1	3-2	1-9	3-9	N/A	N/A
2-2	2-4	2-1	2-2	2-3	2-5	N/A	1-1	1-2	1-3	3-1	3-2	3-3	1-9	3-9	N/A	N/A
2-3	2-1	2-2	2-3	2-5	28	N/A	1-2	1-3	1-5	3-2	3-3	3-5	1-9	3-9	N/A	N/A
2-5	2-2	2-3	2-5	2-8	27	N/A	1-3	1-5	1-8	3-3	3-5	3-8	1-9	3-9	N/A	N/A
2-8	2-3	2-5	2-8	2-7	26	N/A	1-5	1-8	1-7	3-5	3-8	3-7	1-9	3-9	N/A	N/A
2-7	2-5	2-8	2-7	2-6	24	N/A	1-8	1-7	1-6	3-8	3-7	3-6	1-9	3-9	N/A	N/A
2-6	2-8	2-7	2-6	2-4	21	N/A	1-7	1-6	1-4	3-7	3-6	3-4	1-9	3-9	N/A	N/A

(continued on next page)

Table A.2 (continued)

Previous Direction of Movement	Movable Direction																
2-4	2-7	2-6	2-4	2-1	22	N/A	1-6	1-4	1-1	3-6	3-4	3-1	1-9	3-9	N/A	N/A	N/A
3-2	3-4	3-1	3-2	N/A	N/A	2-6	2-4	2-1	2-2	2-3	N/A	N/A	1-9	N/A	N/A	N/A	N/A
3-2	3-1	3-2	3-3	N/A	N/A	2-4	2-1	2-2	2-3	2-5	N/A	N/A	1-9	N/A	N/A	N/A	N/A
3-3	3-2	3-3	3-5	N/A	N/A	2-1	2-2	2-3	2-5	28	N/A	N/A	1-9	N/A	N/A	N/A	N/A
3-5	3-3	3-5	3-8	N/A	N/A	2-2	2-3	2-5	2-8	27	N/A	N/A	1-9	N/A	N/A	N/A	N/A
3-8	3-5	3-8	3-7	N/A	N/A	2-3	2-5	2-8	2-7	26	N/A	N/A	1-9	N/A	N/A	N/A	N/A
3-7	3-8	3-7	3-6	N/A	N/A	2-5	2-8	2-7	2-6	24	N/A	N/A	1-9	N/A	N/A	N/A	N/A
3-6	3-7	3-6	3-4	N/A	N/A	2-8	2-7	2-6	2-4	21	N/A	N/A	1-9	N/A	N/A	N/A	N/A
3-4	3-6	3-4	3-1	N/A	N/A	2-7	2-6	2-4	2-1	22	N/A	N/A	1-9	N/A	N/A	N/A	N/A
1-9	2-1	2-2	2-3	2-4	2-5	2-6	2-7	2-8	1-1	1-2	1-3	1-4	1-5	1-6	1-7	1-8	1-9
3-9	2-1	2-2	2-3	2-4	2-5	2-6	2-7	2-8	3-9	3-2	3-3	3-4	3-5	3-6	3-7	3-8	3-9

Appendix B. Metaheuristic Algorithm Steps

Step 1: Initial population

Set up the number of iteration i as the algebra, and each iteration generates a set of iteration group IG . The generated initial group NP_1 is initialized based on the parameter number of populations. In each individual, the number of aircraft randomly generates DCB tactical instructions.

Step 2: Fitness

The equation B.1 presents each iteration of each iteration requires the calculation of the adaptive value function set NP_F based on the DCB solution.

$$NP_F = \sum_{i=1}^{NP} P_f \quad (i = 1, 2, 3, \dots, NP) \quad (B.1)$$

Fitness value function set consists of three evaluation functions, which shows on equation B.2.

$$P_f = (FV_1, FV_2, FV_3) \quad (B.2)$$

The calculation equation of FV_1 is as follows, where $F(n)$ is the equation in the A*algorithm section B.3.

$$FV_1 = \sum_{i=1}^{N_UAS} F(n) \quad (i = 1, 2, 3, \dots, N_Aircraft) \quad (B.3)$$

Step 3: Selection

In the Selection step of the algorithm, appropriate data is selected based on the fitness value set P_f in NP_F , which will become the chromosome basis for inheritance to the next generation IG . The algorithm first pre-processes the fitness value using the Min-Max Normalization method, which is given by equation B.4:

$$FV' = \frac{FV - \min(FV)}{\max(FV) - \min(FV)} \quad (B.4)$$

Based on the number of valid NPs in the previous fitness calculation, select the weight calculation mode and K-means clustering mode

Step 4: Crossover

All DCB tactical decision-making actions of both parents are exchanged starting from the intersection gene point to form the new NP solutions for the next iteration. This process is repeated until the number of newly generated NPs is equal to the value of number of populations in the parameter, and the newly generated NP solution must not have been calculated before.

Step 5: Mutation

After the crossover step, the new NP solution set generated is subjected to random DCB tactical action changes of random individual genes based on the value of P_m in the parameter. The change of DCB tactical action is based on two principles: the selection of DCB tactical action is based on the aircraft type and the current aircraft state. The model then generates a new NP to replace the original NP , and the newly generated NP solution must not have been calculated before.

Step 6: Determination

In this step, the optimal solution in the current iterative solution set IG is saved and compared with the optimal solution of the previous generation to keep track of the global optimal solution. The algorithm then determines whether the current iteration satisfies the output condition based on the output condition. Let $i = i + 1$, and there are two judgment conditions. The optimal solution is found when i does not meet the set maximum number of iterations, and when the optimal solution is not found, i reaches the set maximum number of iterations. When one of the conditions is met, the al-

gorithm stops and outputs the global optimal solution. Otherwise, it returns to Step 2 and iterates again.

Step 7: Result

The final step is to output the best feasible plan that satisfies the output conditions, including aircraft flight routes, airspace, sector and cell states, aircraft flight missions, and DCB tactical action sets.

References

- [1] O. Cheikhrouhou, A. Koubaa, A. Zarrad, A cloud based disaster management system, *J. Sensor Actuat. Networks* 9 (2020) 6.
- [2] F. Cugurullo, Urban artificial intelligence: from automation to autonomy in the smart city, *Frontiers in Sustainable Cities* 2 (2020).
- [3] M.A. Ruiz Estrada, A. Ndorma, The uses of unmanned aerial vehicles –UAV's- (or drones) in social logistic: natural disasters response and humanitarian relief aid", in: *Procedia Computer Science*, 149, 2019, p. 375.
- [4] J. Rios, *UAS Traffic Management (UTM) Project Strategic Deconfliction: system Requirements Final Report*, 2018.
- [5] SESAR, European ATM Master Plan 2020, Single European ATM Research, 2019.
- [6] FAA, UTM Concept of Operations Version 2.0 (UTM ConOps v2.0), US Dept of Transportation, Office of NextGen, Washington, DC, 2022.
- [7] EUROCONTROL, Think Paper #12 - Aviation under Attack from a Wave of Cybercrime, 2021.
- [8] SESAR, Drone DCB Concept and Process, 2021. DACUS, SESAR-ER4-31-2019-U-space.
- [9] N. Pongsakornthien, A. Gardi, S. Bijjahalli, A multi-criteria clustering method for UAS traffic management and urban air mobility, in: *IEEE/AIAA 40th Digital Avionics Systems Conference*, DASC 2021, San Antonio, TX, USA, 2021.
- [10] N. Pongsakornthien, A. Gardi, R. Sabatini, Interpretable human-machine interactions for UAS traffic management, in: *AIAA Aviation and Aeronautics Forum and Exposition, AVIATION2021*, Virtual Conference, 2021.
- [11] M. Syroeras, V. Vega, ATFCM User Manual, EUROCONTROL, Internet, 2023.
- [12] A. Gardi, R. Sabatini, T. Kistan, Multiobjective 4D trajectory optimization for integrated avionics and air traffic management systems, *IEEE Trans. Aerosp. Electron. Syst.* 55 (1) (2019) 170–181.
- [13] T. Kistan, A. Gardi, R. Sabatini, An evolutionary outlook of air traffic flow management techniques, *Prog. Aerosp. Sci.* 88 (2017) 15.
- [14] T. Prevot, J. Rios, P. Kopardekar, UAS traffic management (UTM) concept of operations to safely enable low altitude flight operations, in: *16th AIAA Aviation Technology, Integration, and Operations Conference*, 2016, p. 3292.
- [15] C. Chin, K. Gopalakrishnan, H. Balakrishnan, Tradeoffs between efficiency and fairness in unmanned aircraft systems traffic management, in: *9th International Conference on Research in Air Transportation, ICRAT*, 2020, 2020.
- [16] Y. Xie, A. Gardi, R. Sabatini, Reinforcement learning-based flow management techniques for urban air mobility and dense low-altitude air traffic operations, in: *IEEE/AIAA 40th Digital Avionics Systems Conference*, DASC 2021, San Antonio, TX, USA, 2021.
- [17] ICAO, Annex 11 to the convention on international civil aviation - Air traffic services, in: International Civil Aviation Organization (ICAO), Montreal, QC, Canada, 2001.
- [18] ICAO, Doc. 4444 - Procedures for air navigation services - Air traffic management, in: The International Civil Aviation Organization (ICAO), Montreal, Canada, 2007.
- [19] A.R. Odoni, L. Bianco, G. Szegö, *Flow Control of Congested Networks*, Nato ASI Subseries F: Computer and Systems Sciences, Springer Berlin Heidelberg, Berlin, Heidelberg, 1987, 38, 1st ed. 1987. ed.
- [20] X. Diao, C.H. Chen, A sequence model for air traffic flow management rerouting problem, *Transportation Research Part E: Logistics and Transportation Review* 110 (2018) 15.
- [21] M. Ozgur, A. Cavcar, 0-1 integer programming model for procedural separation of aircraft by ground holding in ATFM, *Aerosp. Sci. Technol.* 33 (2014) 1.
- [22] V. Dal Sasso, F. Djemou Fomeni, G. Lulli, Planning efficient 4D trajectories in Air Traffic Flow Management, *Eur. J. Oper. Res.* 276 (2019) 676.
- [23] A. Mukherjee, M. Hansen, A dynamic rerouting model for air traffic flow management, *Transportat. Res. Part B: Methodol.* 43 (2009) 159.
- [24] N. Ivanov, F. Netjasov, R. Jovanović, Air Traffic Flow Management slot allocation to minimize propagated delay and improve airport slot adherence, *Transportation Research Part A: Policy and Practice* 95 (2017) 183.
- [25] C. Mannino, A. Nakkerud, G. Sartor, Air traffic flow management with layered workload constraints, *Comput. Operat. Res.* 127 (2021).
- [26] E. Aydogan, C. Cetek, Point merge concept for en route air traffic flow management, *AIAA J. Aircraft* 55 (2018) 2203.
- [27] FAA, NextGen – SESAR State of Harmonisation, US Federal Aviation Administration, Washington, DC, USA, 2018.
- [28] J. Zhang, J. Liu, R. Hu, Online four dimensional trajectory prediction method based on aircraft intent updating, *Aerosp. Sci. Technol.* 77 (2018) 774.
- [29] X. Tang, P. Chen, Y. Zhang, 4D trajectory estimation based on nominal flight profile extraction and airway meteorological forecast revision, *Aerosp. Sci. Technol.* 45 (2015) 387.
- [30] S. Bijjahalli, R. Sabatini, A. Gardi, Advances in intelligent and autonomous navigation systems for small UAS, *Prog. Aerosp. Sci.* 115 (2020) 100617.
- [31] D. Sacharny, T.C. Henderson, A lane-based approach for large-scale strategic conflict management for UAS service suppliers, in: *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2019, pp. 937–945.
- [32] E. Sunil, J. Hoekstra, J. Ellerbroek, Metropolis: relating airspace structure and capacity for extreme traffic densities, in: *ATM seminar2015*, 11th USA/EUROPE Air Traffic Management R&D Seminar, 2015.
- [33] D.-S. Jang, C.A. Ippolito, S. Sankararaman, Concepts of airspace structures and system analysis for uas traffic flows for urban areas. *AIAA Infotech@Aerospace*, I@A 2017, Grapevine, TX, USA, 2017, p. 0449.
- [34] G.F. Newell, A simplified car-following theory: a lower order model, *Transportation Research Part B: Methodological* 36 (2002) 195.
- [35] V. Duchamp, L. Sedov, V. Polishchuk, Density-adapting layers towards PBN for UTM, in: *13th USA/Europe Air Traffic Management Research and Development Seminar*, 2019, p. 2019.
- [36] Z. Wu, Y. Zhang, Integrated network design and demand forecast for on-demand urban air mobility, *Engineering* (2021).
- [37] L. Sedov, V. Polishchuk, Centralized and distributed UTM in layered airspace, in: *2018 8th International Conference on Research in Air Transportation (ICRAT)*, Catalonia, Spain, 2018.
- [38] D. Oliva, M.Abd Elaziz, S. Hinojosa, Metaheuristic Algorithms for Image Segmentation: theory and Applications. *Studies in Computational Intelligence*, 825, 1st ed., Springer International Publishing, Cham, 2019, 2019. ed.
- [39] W. Boukadiha, A. Benamor, H. Messaoud, Multi-objective design of optimal higher order sliding mode control for robust tracking of 2-DoF helicopter system based on metaheuristics, *Aerosp. Sci. Technol.* 91 (2019) 442.
- [40] P. Champasak, N. Panagant, N. Pholdee, Aircraft conceptual design using metaheuristic-based reliability optimisation, *Aerosp. Sci. Technol.* 129 (2022) 107803.
- [41] C. Ma, Y. Yang, L. Wang, Research on distribution route with time window and on-board constraint based on tabu search algorithm, *EURASIP J Wirel Commun Netw* 2019 (1) (2019).
- [42] C. Blum, A. Roli, Metaheuristics in combinatorial optimization: overview and conceptual comparison, *ACM Comput Surv* 35 (2003) 268.
- [43] Y. Wang, J. Zhang, K. Assogba, Collaboration and transportation resource sharing in multiple centers vehicle routing optimization with delivery and pickup, *Knowl Based Syst* 160 (2018) 296.
- [44] J. Jin, X. Ma, I. Kosonen, A stochastic optimization framework for road traffic controls based on evolutionary algorithms and traffic simulation, *Adv. Eng. Software* (1992) 114 (2017) 348.
- [45] S. Samsam, R. Chhabra, Multi-impulse smooth trajectory design for long-range rendezvous with an orbiting target using multi-objective non-dominated sorting genetic algorithm, *Aerosp. Sci. Technol.* 120 (2022) 107285.
- [46] D. Zhao, L. Liu, F. Yu, Chaotic random spare ant colony optimization for multi-threshold image segmentation of 2D Kapur entropy, *Knowl Based Syst* 216 (2021) 106510.
- [47] Q. Zhang, S. Xiong, Routing optimization of emergency grain distribution vehicles using the immune ant colony optimization algorithm, *Appl. Soft Comput.* 71 (2018) 917.
- [48] W. Jiang, Y. Lyu, Y. Li, UAV path planning and collision avoidance in 3D environments based on POMDP and improved grey wolf optimizer, *Aerosp. Sci. Technol.* 121 (2022) 107314.
- [49] X. Zhang, S. Xia, T. Zhang, Hybrid FWPS cooperation algorithm based unmanned aerial vehicle constrained path planning, *Aerosp. Sci. Technol.* 118 (2021) 107004.
- [50] X. Chai, Z. Zheng, J. Xiao, Multi-strategy fusion differential evolution algorithm for UAV path planning in complex environment, *Aerosp. Sci. Technol.* 121 (2022) 107287.
- [51] A. Bakdi, A. Hentout, H. Boutami, Optimal path planning and execution for mobile robots using genetic algorithm and adaptive fuzzy-logic control, *Rob Auton Syst* 89 (2017) 95.
- [52] Z. Wang, J. Cai, Probabilistic roadmap method for path-planning in radioactive environment of nuclear facilities, *Progress in Nuclear Energy (New series)* 109 (2018) 113.
- [53] J.W. Woo, J.-Y. An, M.G. Cho, Integration of path planning, trajectory generation and trajectory tracking control for aircraft mission autonomy, *Aerosp. Sci. Technol.* 118 (2021) 107014.
- [54] S. Karaman, E. Frazzoli, Sampling-based algorithms for optimal motion planning, *Int. J. of Robotics Research* 30 (2011) 846.
- [55] T. Meng, T. Yang, J. Huang, Improved hybrid a-star algorithm for path planning in autonomous parking system based on multi-stage dynamic optimization, *Int. J. Automot. Technol.* 24 (2023) 459.
- [56] C.A. Pötter Neto, G. de Carvalho Bertoli, and O. Saotome, "A-star path planning simulation for UAS Traffic Management (UTM) application", 2021.
- [57] N.P. Sebasco, H.E. Sevil, Graph-based image segmentation for road extraction from post-disaster aerial footage, *Drones* 6 (2022) 315.
- [58] A. Souto, R. Alfaia, E. Cardoso, UAV path planning optimization strategy: considerations of urban morphology, microclimate, and energy efficiency using q-learning algorithm, *Drones* 7 (2023) 123.
- [59] L. Huo, D. Jiang, S. Qi, An ai-based adaptive cognitive modeling and measurement method of network traffic for EIS, *Mobile Networks and Applicat.* 26 (2021) 575.

- [60] J. Wu, C. Sun, C. Zhang, Deep clustering variational network for helicopter regime recognition in HUMS, *Aerosp. Sci. Technol.* 124 (2022) 107553.
- [61] K.P. Sinaga, M.-S. Yang, Unsupervised K-Means Clustering Algorithm, *IEEE access* 8 (2020) 80716.
- [62] J. Hao, X. Zhang, C. Wang, Application of UAV digital photogrammetry in geological investigation and stability evaluation of high-steep mine rock slope, *Drones* 7 (2023) 198.
- [63] H. Khalil, S.U. Rahman, I. Ullah, A UAV-swarm-communication model using a machine-learning approach for search-and-rescue applications, *Drones* 6 (2022) 372.
- [64] J.-H. Chen, H.-H. Wei, C.-L. Chen, A practical approach to determining critical macroeconomic factors in air-traffic volume based on K-means clustering and decision-tree classification, *J. Air Transport Manag.* 82 (2020) 101743.
- [65] S. Bijjahalli, R. Sabatini, A. Gardi, GNSS performance modelling and augmentation for urban air mobility, *Sensors (Switzerland)* 19 (2019) 4209.
- [66] N. Pongsakornthien, S. Bijjahalli, A. Gardi, A performance-based airspace model for unmanned aircraft systems traffic management, *aerosp.* 7 (1) (2020).
- [67] M.M. Gomaa, R. Sabatini, and A. Gardi, "AAM and UAS collision avoidance in the presence of wind and wake turbulence", 2023.
- [68] N. Pongsakornthien, A. Gardi, R. Sabatini, Human-machine interactions in very-low-level UAS operations and traffic management, in: *IEEE/AIAA 39th Digital Avionics Systems Conference, DASC 2020*, San Antonio, TX, USA, 2020.