



Strategic Conflict Management using Recurrent Multi-agent Reinforcement Learning for Urban Air Mobility Operations Considering Uncertainties

Cheng Huang¹ · Ivan Petrunin¹ · Antonios Tsourdos¹

Received: 30 March 2022 / Accepted: 29 November 2022 / Published online: 26 January 2023
© The Author(s) 2023

Abstract

The rapidly evolving urban air mobility (UAM) develops the heavy demand for public air transport tasks and poses great challenges to safe and efficient operation in low-altitude urban airspace. In this paper, the operation conflict is managed in the strategic phase with multi-agent reinforcement learning (MARL) in dynamic environments. To enable efficient operation, the aircraft flight performance is integrated into the process of multi-resolution airspace design, trajectory generation, conflict management, and MARL learning. The demand and capacity balancing (DCB) issue, separation conflict, and block unavailability introduced by wind turbulence are resolved by the proposed the multi-agent asynchronous advantage actor-critic (MAA3C) framework, in which the recurrent actor-critic networks allow the automatic action selection between ground delay, speed adjustment, and flight cancellation. The learned parameters in MAA3C are replaced with random values to compare the performance of trained models. Simulated training and test experiments performed on a small urban prototype and various combined use cases suggest the superiority of the MAA3C solution in resolving conflicts with complicated wind fields. And the generalization, scalability, and stability of the model are also demonstrated while applying the model to complex environments.

Keywords Strategic conflict management · Multi-agent reinforcement learning · Urban air mobility · Wind turbulence

1 Introduction

Urban air mobility enables emerging airborne missions including last-mile delivery, scheduled public air metro, and on-demand air taxis in low-altitude metropolitan regions. In addition, the rapidly iterative process of electrical vertical takeoff and landing (eVTOL) aircraft makes it soon feasible and convenient for urban air transports. However, frequent

flight activities in low-altitude urban airspace must be regulated with safety and security considerations, as the crowded traffic flow with potential separation conflict will represent dangers to ground users. In the meanwhile, the complex urban building morphology changes the micro-climate and wind field apparently, which directly affects the fleet operation in dense-distribution regions. As a consequence, it is crucial to manage the traffic flow efficiently and resolve all conflicts under uncertain circumstances.

Conventionally, the air traffic management (ATM) services (ATS), e.g. advisory, alerting, and preventing collision avoidance, are provided by human air traffic controllers (ATCOs) [1]. Therefore, traffic efficiency is restricted by the controllers' workload in addition to the airspace structure and weather conditions. Similarly, such responsibilities in Unmanned Aircraft System (UAS) Traffic Management (UTM) are undertaken by the UAS Service Supplier (USS) [2, 3]. The dynamic capacity management module in UTM further liberates the potential of dense traffic operation. And not coincidentally, resembling the architecture for UTM,

✉ Cheng Huang
cheng-huang.huang@cranfield.ac.uk

Ivan Petrunin
i.petrunin@cranfield.ac.uk

Antonios Tsourdos
a.tsourdos@cranfield.ac.uk

¹ School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, MK43 0AL, Bedfordshire, UK

the Provider of Services to UAM (PSU) performs ATS in the UAM Operations Environment (UOE) [4, 5]. It means that both in ATM, UTM as well as UAM, similar roles and responsibilities are defined for delivering necessary services.

To achieve conflict-free and efficient traffic management, ATS is offered by the aforementioned service providers in three stages: strategic conflict management, tactical separation provision, and collision avoidance [1]. Although the structure is proposed in ATM, it also applies to UAM and UTM with some modifications. The strategic conflict management service, which is carried out several days or months in advance of the actual operation in UAM or UTM [6] compared with the typical half-year in ATM, works as the first layer and aims to reduce the pressure to the next phases. If there is any conflict left from the previous stage, the separation provision is responsible for keeping the separation minima between the pairwise aircraft. Collision avoidance is an individual component and the last resort on board [1].

In this paper, we focus on conflict management at the strategic level. In detail, strategic conflict resolution is achieved through 3 components: (1) airspace organization and management, (2) demand and capacity balancing (DCB), and (3) traffic synchronization [1]. The airspace organization and management provide the regulations and procedures to flexibly accommodate various types of air activities, facilitate the seamless handling of dynamic flight trajectories and adjust airspace boundaries. The second DCB component mitigates conflicts where the demand exceeds its capacity for the effective use of airspace resources. The final traffic synchronization is fully

integrated with conflict management and DCB to negotiate conflict-free and orderly trajectories.

Prior to the management stage, the flight intents including aircraft information, flight path, and scheduled departure/arrival times must be submitted to PSUs from fleet operators. The PSU will provide the modification suggestions to the submitted plan after coordinating the aforementioned three strategic components.

With the defined airspace management module above, the prescribed intents are firstly reorganized to ensure global operational efficiency in UOE. The airspace structure in low-altitude urban areas differs from the regional ATM airspace and sectors. And the airspace in UAM is expected to be given flexibility, which can be offered by the grid-based solution [7]. The high-density stacked blocks in AirMatrix [8] are filled everywhere in the city. Similarly, the air corridor can also be built over buildings or ground roads. The AirMatrix is then resampled to rarefy blocks in high altitudes and fewer population areas and condenses the blocks around buildings [9], as the number of blocks significantly affects the complexity. As the result, the multi-resolution structure contributes to high-freedom operations.

The flexible airspace structure allows two strategic conflict management solutions. One is directly generating conflict-free trajectories using A^* method [10] with conflict factors involved, however, the underlying problem is that the efficiency is not promised as the frequency of requiring the entire trajectories to be regenerated maintains a high level in the dynamic UOE. The other follows a conventional way to generate trajectories from the submitted flight intents and then eliminate all conflicts. Contrary to the first approach,

Table 1 Comparison with other approaches

Methods	Actions	Learning-based	Uncertainties	Pros	Cons
Bertsimas and Patterson [11]	delay	No	—	<ul style="list-style-type: none"> ● large-scale 	<ul style="list-style-type: none"> ● static airspace configuration
Zhang et al. [12]	rerouting	No	—	<ul style="list-style-type: none"> ● distributed 	<ul style="list-style-type: none"> ● no uncertainties ● static airspace configuration
Spatharis [13]	ground delay	Yes	—	<ul style="list-style-type: none"> ● TBO 	<ul style="list-style-type: none"> ● no uncertainties ● static airspace configuration
Xie et al. [16]	speed change	Yes	weather	<ul style="list-style-type: none"> ● involving wide-range weather 	<ul style="list-style-type: none"> ● no uncertainties ● static airspace configuration
Ours	cancellation; ground delay; speed change	Yes	wind turbulence	<ul style="list-style-type: none"> ● dynamic airspace configuration ● refurbished components for UAM ● involving uncertainties ● including aircraft performance 	<ul style="list-style-type: none"> ● no aircraft performance ● complexity of the multi-resolution airspace structure

it allows for fine-tuning parameters of trajectories, which is closer to the initial intents of the fleet operator.

Besides the airspace and trajectory, the crucial parts of strategic conflict management are the DCB issue and separation conflict, where the DCB issue is caused by the regional demand and capacity imbalance, whereas the separation conflict results from the loss of separation minima between pairwise aircraft. While dealing with the DCB issue, the main focus is reducing the demand or increasing the capacity in the hotspot. Conventional approaches are compared in Table 1. The mathematical models, e.g. 0-1 Integer Programming (IP) [11] and Eulerian-Lagrangian [12], offer options like ground delay, rerouting, or airborne delay to mitigate the crowded traffic in static sectors. But for dynamically merging and dividing airspace blocks, intelligently selecting the appropriate action from ground delay, speed adjustment, and flight cancellation, obviously, those approaches for conventional air traffic flow management (ATFM) lack the ability to deal with those issues in highly dynamic environments. The issue is then investigated with the emerging multi-agent reinforcement learning (MARL) methodologies, which interact with the changing environments to learn generalized solutions. In cooperating with the trajectory-based operation (TBO) concept, the trajectories can be managed effectively. For instance, the agents in the multi-agent system can be defined as “peers” if their trajectories have interaction, and are then connected to form a network [13]. The edge-based MARL and agent-based MARL utilize the information propagated through the graph to solve the DCB issue. To promote the collaboration between agents, the unsupervised clustering and ranking are integrated with MARL to select the agent with priority [14]. Other approaches such as hierarchical MARL also provide insight into DCB resolution for manned air traffic [15]. And for unmanned aircraft in UTM or UAM, the deep Q-Learning network (DQN) with genetic algorithm (GA) is a good attempt to resolve the congestion problem in urban low-altitude airspace [16] consider weather changes. Because of the difference in airspace structure and operational notion, the hybrid operation of manned and unmanned flight, scheduled and on-demand flight, both pose a huge challenge for UAM.

Solving the DCB issue can only alleviate the regional traffic pressure, the problem of losing separation distance in the sector remains to be resolved. The separation matter is always studied in the tactical phase, where the instant heading change, speed adjustment, and flight level change are typical maneuvers. It becomes much easier to involve uncertain factors in the short-time stage, due to this, the single-agent deep deterministic policy

gradient (DDPG) [17] and multi-agent version [18] can cope with the conflicts between pairwise or multiple aircraft considering environmental uncertainties. It is acceptable to simply send the heading change command with the specific modification degree in the wide-range airspace, whereas this operation becomes risky in UOE, as (1) it requires much more high-frequency command exchange to guide the flight, which increases the pressure to the system and (2) the flights in UOE must be strictly planned and monitored to avoid safety issues. For these reasons, elaborated information for trajectory modification should be designed for the secure and efficient operation of UAM. In this paper, as also described in Table 1, components for UAM operational concept are refurbished considering urban uncertainties and aircraft performance. To achieve this, the flexible airspace structure and refined strategy for trajectory management collaborate with MARL for better strategic conflict management. The flight performances such as aircraft type and flight ability are determined as the strict constraints in the process of defining those elements. The airspace structure is firstly adapted for UAM with dense blocks, meanwhile, the multi-resolution blocks are merged or split in accordance with the aircraft dimension. To support precise operation, the Gaussian Mixture Model (GMM) is leveraged to model the speed profile of the flight, consequently, the conflict can be resolved by adjusting the parameters of GMM. The wind model is integrated into the environment and built through Dryden Gust Model and computational fluid dynamics (CFD). The strategic conflict in this paper consists of the DCB issue, separation conflict, and unavailable block status due to wind intensity. After carefully designing the UAM architecture, the Multi-Agent Asynchronous Advantage Actor-Critic (MAA3C) framework with recurrent actor-critic networks is proposed to train a generalized model and eliminate all conflicts in any dynamic environments, where both ground delay, speed adjustment, and cancellation serve as candidate actions.

Overall, the key contributions of the paper can be concluded as follows:

- Components in strategic conflict management are especially refurbished for urban air mobility, in which the aircraft flight performance is integrated into the multi-resolution airspace organization and spatiotemporal trajectory generation. Especially, the maximum wind speed resistance level of each aircraft is considered for flight safety.
- The Dryden gust model and computational fluid dynamics (CFD) simulation are utilized to model the wind turbulence in a low-altitude urban area and provide regional and detailed wind profiles as the stochastic factors in the UOE.

- Cancellation, ground delay, and airborne speed adjustment comprise three types of actions for the constructed Multi-Agent Asynchronous Advantage Actor-Critic (MAA3C) solution. The importance of each action is dynamically adjusted by the recurrent structure of MAA3C. In particular, the cancellation action is emphasized to enable stable and effective performance in presence of wind turbulence.

The rest of the paper is outlined as follows: Section 2 defines the problem and essential components of strategic conflict management in addition to wind turbulence models. The customized multi-agent reinforcement learning framework and characterized networks are presented in Section 3. In Section 4, we develop a small prototype for UAM simulation and analyze the training and test results under kinds of circumstances. Finally, the paper is concluded in Section 5.

2 Operational Concept Components

Compared with conventional ATM, the operational environment is narrowed from large-scale high-altitude spaces to low-altitude high-density areas. Besides, the urban morphology creates additional static obstacles and dynamic micro-climate, which impact flight safety and security. Moreover, the high-density flights and uncertainties increase the airspace demand and conflict possibilities, which make it challenging to solve the DCB and conflict issues in urban regions. As a result, this paper works on a proper solution for UAM services, especially for solving the strategic DCB issue and separation conflicts in dynamic urban environments.

In this section, essential components of conflict management are introduced step by step to construct the problem. As depicted in Fig. 1, the dynamic low-altitude airspace

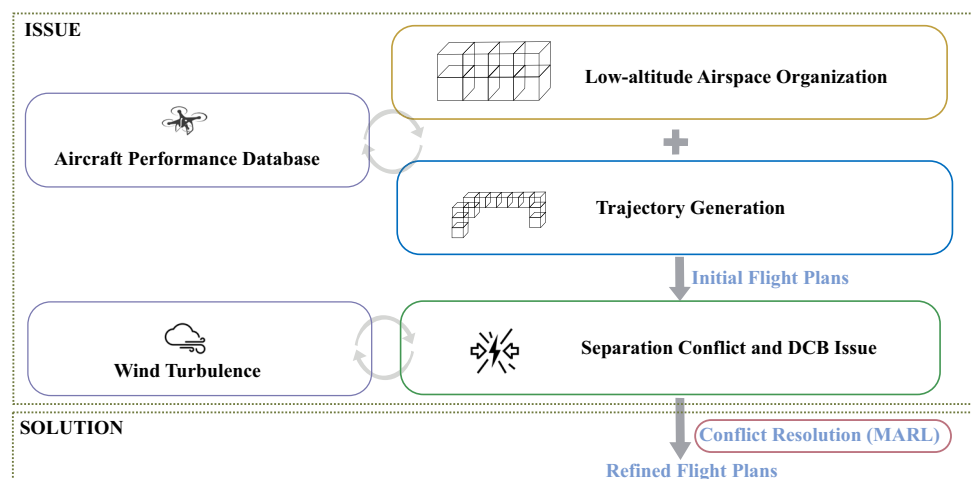
and spatiotemporal trajectory are generated at first considering aircraft performance. These two components produce the initial flight schedule and analyze potential conflicts between trajectories in complicated environments after involving wind turbulence. Then the issue will be solved by the MARL solution considering wind uncertainties to generate final conflict-free flight plans. Details of each component describing the problem are explained as follows. And the conflict-free flight plan is refined by the MARL solution which will be described in Section 3.

2.1 Airspace Organization and Management

UAM operations environment (UOE) defines the airspace structure in which PSU provides necessary services to aircraft for safety and efficiency. As the densely distributed buildings pose challenges to secure flight, dynamic and adaptive organization is required for various air activities in low-altitude airspace. The multi-layered grids can formulate a flexible structure, in which the airspace is discretized into stacked blocks. The safety separation from buildings can be achieved by placing blocks far from the static obstacles. For instance, the AirMatrix [8] or 4D grids [7] over buildings or roads can provide sufficient flexibility for airspace management. Whereas, to serve accurate navigation, the increase of block resolution must lead to the exploding number of controlled blocks. Although the multi-resolution AirMatrix at different areas and altitudes can decrease the burden to some extent [9], the static architecture is lack operational freedom.

In the convenience of conflict management, we define that the aircraft can only occupy one operation block. To further increase the flexibility while achieving high-resolution operation, we place basic blocks and dynamically merge them into larger blocks according to the size of the aircraft. As in Fig. 2, 8 basic blocks construct a

Fig. 1 Workflow for strategic conflict management



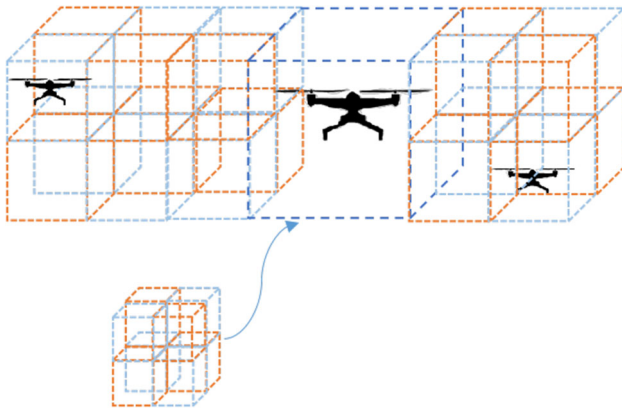


Fig. 2 Mixed-resolution

large block as an actual operation block for the large-size aircraft, and the small aircraft remain operating in the basic blocks. In this case, the entire structure keeps dynamic and varies over time. All basic blocks constitute a static graph $G_g(V, E)$, in which the discrete blocks are regarded as the set of nodes V , and the adjacency between pairwise nodes constitutes the edge set E . And the practical operation graph is mathematically represented as a graph $G(V, E)$, of which the original basic nodes are deleted and replaced with the new merging nodes as shown in Fig. 3. Besides, children nodes are also stored in G for block retrieval and recovery. Since the graph G changes over time, the merging and recovery process is critical, we summarized the detailed process in Algorithm 1.

2.2 Trajectory Generation

The trajectory is typically defined by a list of key information including latitude, longitude, altitude, and time. With detailed graph, the trajectory is formulated by $traj_f = \{(t_i, node_i, v_i)\}_{i=0}^{N_f-1}$, where $node_i \in \mathbb{R}^3$ is the node in the graph G which contains 3D position, $v_i \in \mathbb{R}_+$ and N_f is the number of traversed blocks of flight f .

We design a trajectory formulation module for strategic conflict management with aircraft performance taken into account. The time-based management (TBM) is the kernel part of the trajectory-based operation (TBO), by assigning

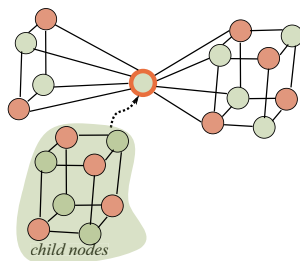


Fig. 3 Nodes connection in graph view

Require: Global graph $G_g(V, E)$ and position pos of aircraft.

```

1: while  $pos \neq 0$  do
2:   Get basic blocks  $v_i$  ( $i \in \{0, 1, \dots, 7\}$ ) near the
   position  $pos$  and merge them into one block  $v_l$ .
3:   Put  $v_i$  into the child nodes queue  $C$ .
4:   Duplicate the graph  $G_g$  to  $G'_g$ 
5:   Delete nodes  $v_i$ , add node  $v_l$ 
6:   Connect near nodes with  $v_l$ .
7:   if  $len(C) > 8$  then
8:     Remove  $len(C) - 8$  child nodes based on first-
     in-first-out (FIFO) principle.
9:     Recover the connection of removal nodes.
10:  end if
11: end while

```

Algorithm 1 Dynamic merging and recovering of nodes.

specific time at points on the spatial trajectory for strategic planning. In this way, to enable efficient TBM, two components including spatial points design and time assignment will be implemented in detail as below.

2.2.1 Spatial Trajectory

The spatial information of a trajectory is composed of discrete 3-dimensional points. The center of blocks $node_i \in \mathbb{R}^3$, represents the specific point along a trajectory. The objective is to collect the list of consecutive nodes between the departure and arrival locations. Merging nodes happens in the process of trajectory generation. The aircraft performance database $AC = \{ac_i\}_{i \in N_a}$ and vertiport database VT are firstly collected. Dijkstra algorithm is utilized to find the shortest path between the origin vertiport Ori and destination vertiport Des on graph $G_g(V, E)$. The detailed process is described in Algorithm 2.

2.2.2 Time Assignment

With the prescribed spatial points, efficient traffic flow is then achieved by the TBM. Under the condition that the

```

1: for each aircraft do
2:   Choose one aircraft  $ac_i$  from  $AC = \{ac_i\}_{i \in N_a}$ ;
3:   Random vertiport selection for  $Ori$  and  $Des$  from
    $VT$ ;
4:   Assign a cruise level  $FL$ ;
5:   Generate shortest path:  $\{node_i\}_{i=0}^{N_f-1} =$ 
    $Dijkstra(Ori, Des, G(V, E))$ .
6: end for

```

Algorithm 2 Trajectory generation.

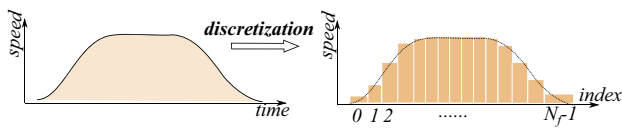


Fig. 4 Discretized time-speed graph

waypoints are not revised, one option is the ground delay (GD) and the other one is adjusting the time when reaching certain waypoints.

Electric vertical take-off and landing (eVTOL) aircraft or small drones have more advanced performance than fixed-wing aircraft in the urban environment, because of flexible maneuvers and airborne hovering abilities. To accurately revise the arrival time at a specific location to avoid potential conflict, we construct a time-speed graph based on the Gaussian Mixture Model (GMM) as the en-route speed profiles of aircraft. The GMM is composed of K independent normal distribution $\mathcal{N}$, and the probabilistic density function (PDF) is:

$$\begin{cases} p(x) = \sum_{k=1}^K \phi_k \mathcal{N}(x | \mu_k, \sigma_k) \\ \mathcal{N}(x | \mu_k, \sigma_k) = \frac{1}{\sigma_k \sqrt{2\pi}} \exp\left(-\frac{(x-\mu_k)^2}{2\sigma_k^2}\right) \\ \sum_{k=1}^K \phi_k = 1 \end{cases} \quad (1)$$

where μ_k , σ_k and ϕ_k are the mean value, variance and weight of the k^{th} distribution, respectively.

Since the GMM curve is continuous, it needs to be discretized into counterparts as the number of traversed blocks as in Fig. 4, supposing that the speed is constant inside the block and changes at the border of two adjacency blocks. All aircraft need to follow the advised speed at each kernel point. With traversing blocks $\{node_i\}_{i=0}^{N_f-1}$, speed profile $\{v_i\}_{i=0}^{N_f-1}$ and initial departure time t_0 given, the crossing time at next blocks $\{t_i\}_{i=0}^{N_f-1}$ can be obtained by $t_i = t_{i-1} + \dim_{block}/v_i$ ($1 \leq i \leq N_f - 1$), where $\dim_{block}$ is the dimension of the operation block. If the new speed profile v'_i is refined for conflict resolution, the new trajectory $\{(t'_i, node_i, v'_i)\}_{i=0}^{N_f-1}$ is then updated.

In addition to airborne adjustment, the GD program directly shifts the entire trajectory in the temporal dimension by adding the delay time δt and fresh plan $\{(t_i + \delta t, node_i, v_i)\}_{i=0}^{N_f-1}$ as depicted in Fig. 5.

In some situations, both GD and speed change are applied, the new operation plan $\{(t'_i + \delta t, node_i, v'_i)\}_{i=0}^{N_f-1}$ substitutes the original one.

2.3 Wind Turbulence

Stochastic uncertainties such as wind disturbance pose a critical challenge to flight safety. Not only directly deviating from the nominal trajectory, but the strong wind also has a

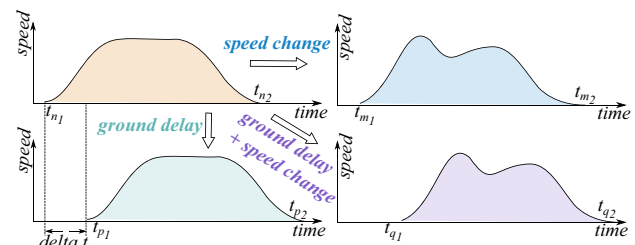


Fig. 5 Time-speed graphs based on GMM

great possibility of causing failed control or drone crash. In this strategic stage, instead of adding wind to the dynamics and kinetics model of aircraft, the wind field is used to determine if the aircraft can take off. Any aircraft, of which the maximum wind speed resistance is less than the predicted wind speed at some time and points, needs to make efforts to avoid unavailable blocks.

Two wind models are leveraged to describe the wind information in different resolutions. The first one utilizes the Dryden model to predict the regional wind speed on each aircraft [19]. Because of the complex urban morphology, the regional weather prediction model is not sufficient for the local wind field prediction. Then the computational fluid dynamics (CFD) model is simulated to get high-resolution wind flows [20]. Although the simulated scenarios cannot be exactly the same as the frequently changing wind field, our scope is making the solution generalized and being capable of resolving any similar issues.

Dryden gust model is a commonly used mathematical model for continuous wind turbulence. And in this wind model, velocity components are defined by the stochastic processes and corresponding power spectral density. As the wind is calculated in the vehicle frame, the transfer function in the three-body frame axes $[u, v, w]$ are calculated by [21]:

$$\begin{cases} H_u(s) = \sigma_u \sqrt{\frac{2V_a}{\pi L_u}} \frac{1}{s + \frac{V_a}{L_u}} \\ H_v(s) = \sigma_v \sqrt{\frac{3V_a}{\pi L_v}} \frac{\left(s + \frac{V_a}{\sqrt{3}L_v}\right)}{\left(s + \frac{V_a}{L_v}\right)} \\ H_w(s) = \sigma_w \sqrt{\frac{3V_a}{\pi L_w}} \frac{\left(s + \frac{V_a}{\sqrt{3}L_w}\right)}{\left(s + \frac{V_a}{L_w}\right)^2} \end{cases} \quad (2)$$

where $[\sigma_u, \sigma_v, \sigma_w]$ and $[L_u, L_v, L_w]$ are the turbulence intensity and scale length, V_a is the effective airspeed of small aircraft. In [22], it suggests setting a constant speed V_{a0} for the Dryden model.

From Eq. 2, the different areas at the same altitude share the same wind value $\{W_t\}_{t=t_s}^{t_e}$, where t_s and t_e are the starts and end time of operation. But the dense distribution and morphology of buildings have a great impact on the local wind flow, such as the street-canyon effect and induced turbulence, etc. Various points at the same altitude must

have different $\{W_t\}_{t=t_s}^{t_e}$ profiles. Due to this, numerical computation using the CFD is able to solve the dynamic Navier-Stokes equations for the changing wind field.

2.4 Conflict Management

The DCB issue and the separation conflict are two matters in strategic conflict management. In ATM, capacity typically refers to the number of flights the ATCOs can process during a specific period. And because each ATCO is

responsible for a certain number of airspace sectors, the demand also represents the capacity of the sector. But in this paper, we use automatic tools to support the decision-making, in that way, and we regard the capacity of a PSU as infinite. For each block, the capacity is customized to 1, in other words, every block can only be occupied by one aircraft at each moment.

The demand is obtained through entry count. In detail, the operation period (H hours) in each day is equally split into SN fragments like in:

$$\underbrace{\{[t_s, t_s + dt]\}}_{sn=0}, \underbrace{\{[t_{s_1} + dt, t_{s_1} + 2dt]\}}_{sn=1}, \dots, \underbrace{\{[t_{s_1} + (SN - 2)dt], (t_{s_1} + (SN - 1)dt]\}}_{sn=SN-1} \quad (3)$$

where $dt = H/SN$. Any aircraft whose entry time t_i ($i \in [0, N_f - 1]$) into block j at $sn \in \{0, 1, \dots, SN - 1\}$ is added into the demand $d_{node_j, sn}$. When the demand $d_{node_j, sn}$ exceeds the capacity $c_{node_j, sn} == 1$ in any block, the DCB issue appears. For pairwise aircraft in the hotspot block, they have:

$$node_{j_1} == node_{j_2} \quad (4)$$

Where j_1 and j_2 are block indexes of different flights at the same period. This expression also meets the condition below:

$$dist(node_{j_1} - node_{j_2}) < dim_{block} \quad (5)$$

The loss of separation means the distance between pairwise aircraft is less than the safe minima. For efficient operation with aircraft performance, the dynamic separation minima are considered, where the nearest spatiotemporal distance between points $node_{j_1}$ and $node_{j_2}$ of two interactive trajectories is compared with the relative margin:

$$dist(node_{j_1} - node_{j_2}) < (\delta v_{max} \times \tau) \quad (6)$$

where δv_{max} is the maximum relative speed, and τ is the temporal margin.

From Eqs. 5 and 6, if we can keep $(v_{max} \times \tau) \gg dim_{block}$, the separation conflict and DCB issue can be merged into one meta-problem. There is no DCB issue in each block if separation conflicts are resolved.

When considering stochastic wind, we introduce a special term *availability* for the block, which determines if the aircraft can access the block. If the wind speed in specific traversing blocks is larger than the maximum wind speed resistance at some periods, the block will be unavailable for this aircraft. In addition to the separation conflict, we also define a new conflict, where the planned trajectory contains unavailable blocks.

The entire initial operation plan $\{traff = \{(t_i, node_i, v_i)\}_{i=0}^{N_f-1}\}_{f=0}^{N_a}$ is scanned to mark the conflicted flights and corresponding block index.

3 Multi-agent Reinforcement Learning Solution

In this section, the strategic conflict problem is formulated as a multi-agent system. The Partially Observable Markov Decision Process (POMDP) is described, and as an extension, the MAA3C with mask recurrent structure is implemented.

3.1 POMDP

In POMDP, the agent i ($i \in \mathcal{N}$) in the multi-agent system $\mathcal{N} = \{1, \dots, N_a\}$ only observes the local information $\mathcal{O}^i$ from global state $\mathcal{S}$. Let $\mathcal{A}^i$ be the action space of agent i , then $\{\mathcal{A}^i\}_{i \in \mathcal{N}}$ formulates the joint action space $\mathcal{A} = \mathcal{A}^1 \times \dots \times \mathcal{A}^N$. The transition probability from the previous state to the next state $\mathcal{P} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$. After executing the action, each agent obtain its reward $\mathcal{R}^i : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathcal{R}$ with a discount factor $\gamma \in [0, 1]$. As a result, the tuple $(\mathcal{N}, \mathcal{S}, \{\mathcal{A}^i\}_{i \in \mathcal{N}}, \mathcal{P}, \{\mathcal{R}^i\}_{i \in \mathcal{N}}, \gamma, \{\mathcal{O}^i\}_{i \in \mathcal{N}})$ establishes the POMDP [23]. The detailed components are described in detail as follows:

- *Agent*: Agents are flights with conflict status in simulated scenarios, for instance, losing safe separation or encountering intolerant wind. As the number of trajectories is large and varies in the operation plan, not all flights need to take actions. From this definition, we can recognize the changing number of agents at each training step because of the high-dynamic environment.

- **Observation** $o_{st}^i \in \mathcal{O}^i$: In the schedule, the global state $\mathcal{S}$ contains the expected positions and time of all flights. But in POMDP, we require that the agent i can only observe information concerning itself at each step st , and parameters of other agents remain blind for it. The observation follows the rule $\mathcal{S} \times \mathcal{N} \rightarrow \mathcal{O}$. Specifically, the observable elements in this paper consist of the traversed blocks $\{node_i\}_{i=0}^{N_f-1}$, conflict status $\{c_i\}_{i=0}^{N_f-1}$ and entry time $\{t_i\}_{i=0}^{N_f-1}$ into in each block. For each agent, the observation matrix is denoted by:

$$o_{st}^i = \begin{bmatrix} node_0 & c_0 & t_0 \\ node_1 & c_1 & t_1 \\ \vdots & \vdots & \vdots \\ node_{N_f-1} & c_{N_f-1} & t_{N_f-1} \end{bmatrix}_{(N_f \times 3)} \quad (7)$$

We can note that the dimension $N_f \times 3$ varies with the agent. And the conflict is set to $c_i = 1$ if the agent loses safe separation or the wind W_i is greater than its maximum wind speed resistance in $node_i$ at time t_i , otherwise $c_i = 0$.

- **Action** $a_{st}^i \in \mathcal{A}^i$ ($i = 1, \dots, N_a$): Each agent receives its action according to the policy $\pi^i : \mathcal{O}^i \rightarrow \mathcal{A}^i$. To resolve complex conflicts in uncertain environments, both ground delay, airborne speed adjustment, and cancellation are designed as effective candidate actions. The action a_{st}^i of every agent formulates the joint action. Agents cooperate to achieve global optimization and are viewed as homogeneous.

Although 3 types of actions are listed, just one type is allowed in each step. If the ground delay is selected as the action, the entire trajectory will shift δt in temporal dimension and be updated with $traj'_f = \{(t_i + \delta t, node_i, v_i)\}_{i=0}^{N_f-1}$. While the airborne speed is to be changed, $\delta\mu_k, \delta\sigma_k$ ($k \in [1, K]$) is the specific output for this action. Updating $\mu'_i = \mu_i + \delta\mu_i$ and $\sigma'_i = \sigma_i + \delta\sigma_i$ in GMM model gives the new speed profile $\{v'_i\}_{i=0}^{N_f-1}$, as a result, the trajectory is refreshed by $traj'_f = \{(t'_i, node_i, v'_i)\}_{i=0}^{N_f-1}$. Finally, when cancellation is the selection, the current agent will be removed from the operation schedule.

- **Reward** $\mathcal{R}_{st}^i$: All agents cooperate for conflict resolution and cost minimization. With the flight performance taken into account, the performance of executed action a_{st}^i is assessed by the previous observation o_{st}^i and updated o_{st+1}^i , the reward function is then shaping as:

$$\mathcal{R}_{st}^i(o_{st}^i, a_{st}^i, o_{st+1}^i) = r_1 + r_2 + r_3 + r_4 + r_5 + r_6 \quad (8)$$

where,

- $r_1 = \Delta t_f = t_{f2} - t_{f1}$ shows the increased en-route flight time for agent i between two iterative plans.
- $r_2 = \Delta t_a$ is the increased arrival time;
- $r_3 = \|\partial \{v_i\}_{i=0}^{N_f-1} / \partial t - dv_{max}\|$ controls the range of speed change in case of sharp gradient;
- $r_4 = c_{st+1} - c_{st}$ record the number change of conflicts;
- $r_5 = t_{max}^f - (t_{N_f-1} - t_0)$ is the maximum flight time of this aircraft and requests the en-route flight time should be less than it.
- r_6 is the cancellation cost. We always assign it with a constant.

3.2 MAA3C

With the necessary components defined for the multi-agent system, we need to consider an efficient architecture to enable stationary and scalability. In POMDP, the multi-agent system is non-stationary since other agents interact with the environment simultaneously. Some approaches can deal with the non-stationary issue such as centralized training, opponent modeling, communication, etc. [24]

In this paper, the actor-critic framework instead of value-based Q-learning is built and the centralized critic is utilized to cope with the non-stationary problem. Instead of concatenating observation matrices of all agents like a typical centralized actor, we employ parameter sharing to reduce the complexity. Each agent shares the same policy network. However, it just takes its observation as input. In this way, agents can still keep uniqueness because the various inputs give different hidden features. This procedure makes the shared policy network a generalized opponents model which can deal with any type of input.

The asynchronous advantage actor-critic algorithm (A3C) [25] is selected as the basic architecture and extended for the multi-agent system (MAA3C). The single-agent A3C assesses the action with the advantage function $A^\pi(\mathcal{S}_{st}, a_{st}) = Q^\pi(\mathcal{S}, a) - V^\pi(\mathcal{S}) = \sum_{j=0}^{k-1} \gamma^j \mathcal{R}_{st+k}^j + \gamma^k V^\pi(\mathcal{S}_{st+k}) - V^\pi(\mathcal{S}_{st})$, where $Q^\pi(\mathcal{S}, a) > V^\pi(\mathcal{S})$ of action a indicates the better return of the current policy. To enable the multi-agent extension, each agent shares the common advantage function $A^\pi(\mathcal{S}_{st}, a_{st})$. Because of the parameter sharing, the average advantage value $A_{avg}^\pi = \left\{ \sum_{n=0}^{N_a} A_n^\pi(\mathcal{S}_{st}^n, a_{st}^n) \right\}$ contributes to the policy loss $J(\pi_{\phi'})$ and critic loss $L(\phi_v)$, as the consequence, updating the policy network ϕ and the critic network ϕ_v .

$$J(\pi_{\phi'}) = \mathbb{E} \left[\sum_{st=0}^T \left\{ \log \pi_{\phi'}(a_{st}^n | \mathcal{S}_{st}^n) \right\}_{avg} \cdot A_{avg}^\pi \right] \quad (9)$$

$$L(\phi_v) = \mathbb{E} \left[\frac{1}{2} \sum_{st=0}^T \left(\{U - V(S_{st}^n; \phi_v')\}_{avg} \right)^2 \right] \quad (10)$$

where $\{ \log \pi_{\phi'}(a_{st}^n | S_{st}^n) \}_{avg}$ is the average $\log$ value of all agents with policy $\pi_{\phi'}$. And $\{U - V(S_{st}^n; \phi_v')\}_{avg}$ denotes the average advantage value A_{avg}^π , U is the accumulative returns and V is the critic value for state S_{st}^n . $\mathbb{E}[\cdot]$ refers to the expected value.

3.2.1 Recurrent Actor-Critic

The learning procedure is formulated as a multi-agent system. However, two issues remained to be processed. One is the various number N_f of traversed blocks for each aircraft, which leads to the dimension misalignment. The other one is the scalability of the system which would accept various agent numbers. Those problems are addressed by the recurrent actor-critic networks as depicted in Fig. 6. Instead of feeding the input one by one, the observations of all agents are stacked. Specifically, the number of agents is treated as batch size. The first layer concatenates the information of all traversed blocks, and it can process flexible input length thanks to the encoding ability of recurrent neural networks. Different components in the second layer generate the speed adjustment parameter $[\delta\mu_k, \delta\sigma_k]$ ($k \in [1, K]$, $K = 2$), ground delay δt , specially designed variant *mask*, cancellation value *Ca* and the critic value $V^\pi(S)$. As only one action is expected to be selected at each step, a two-value mask $mask = 0$ or 1 is proposed. It determines the action between ground delay and speed refinement at first, but the cancellation *Ca* has the highest priority, if it outputs the cancellation decision $Ca = 1$, this agent and related parameters would be removed from the flight schedule. The action determination at each step is given by:

$$a_{st}^i = [mask \times [\delta\mu_k, \delta\sigma_k], (1 - mask) \times \delta t] \times Ca \quad (11)$$

Fig. 6 Recurrent neural networks

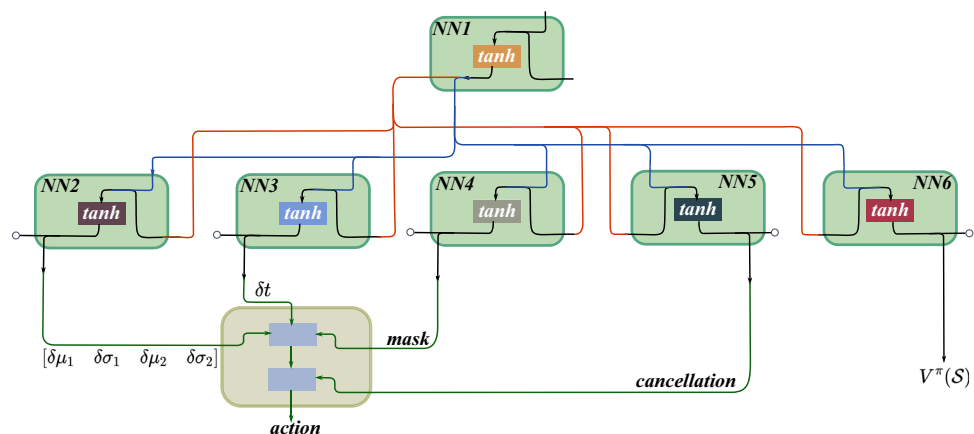
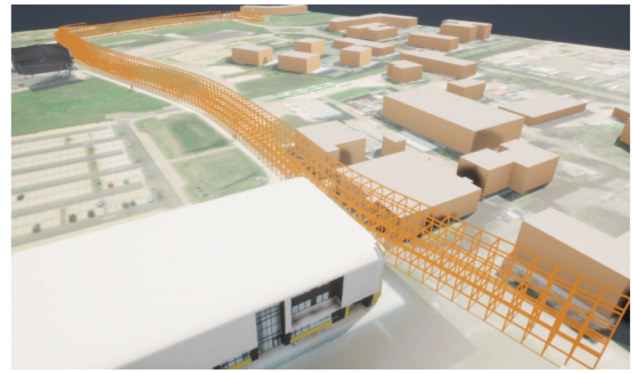


Fig. 7 Perspective view

The two-layer structure can deal with the dynamic number of traversed blocks and various agent numbers at the same time. In this way, the scalability of the multi-agent solution can be guaranteed. Training and testing on different dimensional operation plan naturally become acceptable. Combined with the multi-agent learning procedure, the recurrent actor-critic structure is integrated into the MAA3C framework as in Algorithm 3.

4 Use Cases

In this section, we establish a small-scale urban environment to accomplish the last-mile delivery task with hundreds of small UAVs. And workflows for low-altitude airspace organization, operation plan generation, aircraft performance database, and wind models are presented exhaustively. The proposed MAA3C solution is then applied to the prototype to evaluate its effectiveness.

4.1 Flight Performance Dataset

To enable efficient performance-based operation, flight performance plays a critical role in the process of airspace

management, trajectory generation, conflict management, and multi-agent systems. Flight performances of 8 drones are collected and listed in Table 2, in which those small aircraft have the capacity to carry lightweight payloads. its body frame, enables the airspace management module to merge and recover blocks accordingly. We outline that the operation block ought to be a minimum of double the size of the aircraft. In this case, drones such as DJI AGRAS T30 and T60 can only operate in the merged operation block. On the contrary, other types of drones can only fly in basic $4m$ -dimension blocks. The maximum hovering time with a full or empty payload is the constraint for flight time. And the speed range is the maneuverability of drones and requires the multi-agent system to follow

Require: parameters ϕ and ϕ_v for global thread, ϕ' and ϕ'_v for child threads, $st \leftarrow 1$ for step counter;

- 1: **while** $MaxSteps$ **do**
- 2: $d\phi \leftarrow 0$ and $d\phi_v \leftarrow 0$
- 3: Synchronize parameters from child thread: $\phi' = \phi$ and $\phi'_v = \phi_v$
- 4: $st_{start} = st$
- 5: Determine the valid agent list from the up-to-date schedule based on entry count
- 6: Get all agents' observations $\{\mathcal{O}^i\}_{i \in \mathcal{N}}$
- 7: **while** terminal s_{st} or $st - st_{start} == st_{max}$ **do**
- 8: Determine the effective action among speed change, ground delay and cancellation with Eq. 11 for each agent i
- 9: Execute action $a_{st} = \{a_{st}^i\}_{i \in \mathcal{N}}$ according to policy π
- 10: Get reward $\mathcal{R}_{st}$ and update the schedule for new state s_{st+1}
- 11: $st \leftarrow st + 1$
- 12: $U = \begin{cases} 0 & \text{for terminal } s_{st} \\ V(s_{st}, \phi'_v) & \text{for non-terminal } s_{st} \end{cases}$
- 13: **end while**
- 14: **for** $i \in \{st - 1, \dots, st_{start}\}$ **do**
- 15: $U \leftarrow \mathcal{R}_i + \gamma U$
- 16: $\phi': d\phi \leftarrow d\phi + \nabla_{\phi'} \{ \log \pi(a_i | s_i; \phi') \}_{avg} \cdot \{U - V(s_i; \phi'_v)\}_{avg}$
- 17: $\phi'_v: d\phi_v \leftarrow d\phi_v + \partial \left(\{U - V(s_i; \phi'_v)\}_{avg} \right)^2 / \partial \phi'_v$
- 18: **end for**
- 19: Update of ϕ and ϕ_v using $d\phi$ and $d\phi_v$
- 20: **end while**

Algorithm 3 Pseudocode Code for the MAA3C algorithm.



Fig. 8 Top view

the restriction when generating actions. The maximum wind speed resistance is the crucial component for safe flight in an uncertain environment. Any flight encountering excessive wind speed is not permitted.

4.2 Urban Environment

4.2.1 Low-altitude Air Corridor

To enable flexible airspace management, the dense grid is a great preference. Therefore, an air corridor with multi-layer blocks is constructed over Multi-User Environment for Autonomous Vehicle Innovation (MUEAVI) road built at Cranfield University, as depicted in Fig. 7. The dimension of each block is set to $4m \times 4m \times 4m$. Ground infrastructures such as cameras, lidar, and radar have great potential to achieve resilient surveillance in urban GPS-degraded areas. We implement this low-altitude air corridor in Carla Simulator [26] for corridor and trajectory visualization.

As displayed in Fig. 8, 5 vertiports are simulated in the environment for UAVs taking off and landing. In

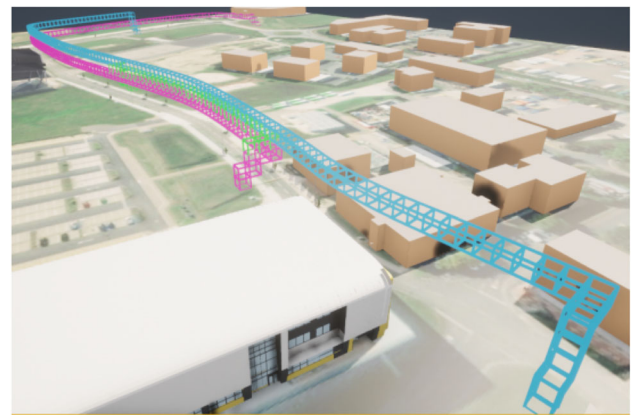


Fig. 9 Trajectories

Table 2 Small aircraft performance data

Aircraft Type	Size (m)	Max Hovering Time (min) [Full Payload]	Max Hovering Time (min) [Empty Payload]	Speed Range (m/s)	Max Wind Speed Resistance (m/s)
DJI Matrice 600	1.668	16	35	[-18,18]	8
DJI Matrice 100	0.65	13	22	[-22, 22]	10
DJI Matrice 200	0.887	27	38	[-23, 23]	12
DJI Mavic pro	0.887	24	24	[-18, 18]	9
Horsefly Gen 5.3	1.1	25	50	[-22, 22]	8
DJI MG-1S	1.47	10	22	[-15, 15]	7
DJI AGRAS T30	2.858	7.5	20.5	[-10, 10]	8
DJI AGRAS T16	2.509	10	18	[-10, 10]	8

Section 2.1, we explain the graph construction process for dense grids, all 1900 blocks comprise the nodes set V of the graph $G(V, E)$.

4.2.2 Operation Plan Generation

With the flexible airspace management strategy, we are now able to generate batches of trajectories, which then formulate the entire operation plan. In the training phase, there are 100 trajectories simulated considering the flight performance and air corridor structure according to Algorithm 2. Some selected trajectories are drawn in Fig. 9. The valid operation time starts from 8:00 to 18:00, and it also means the time assignment should be limited in this range. The operation timeline is then divided into 8640 snapshots, where the entry demand is counted in each 5s-period snapshot. As depicted in Fig. 10, the spatiotemporal information of all trajectories is projected into the 2D map. Each point shows the existence of the aircraft in the block at each period and formulates consecutive trajectories when indices of traversing blocks are sequential. It is clear that some overlapped trajectories lead to separation conflicts.

We count and then visualize the traffic flow in the 3D air corridor structure as in Fig. 11, combined with the conflict map in Fig. 12, it is convenient to observe the crowded traffic in the vicinity of vertiports and nearby airspace. There are 10 failed separations accumulated near blocks above vertiports.

4.3 Model Training

As the wind turbulence is integrated into the environment, the conflict number must change once the wind effect is activated. To involve specific wind profiles, the Dryden gust model and CFD model are simulated to get regional and local wind information, respectively. The Dryden gust data for training (Dryden-I) is generated as shown in Fig. 13. To

make the information condensed, the directional velocities are projected into a wind-rose plot in Fig. 14 to observe the frequency of magnitude and direction. We can notice the predominant wind towards east (0°) and south-east (315°) directions. But in the strategic phase, the magnitude poses greater threats to the operation safety than the direction. From the wind-rose diagram, the intensity over 8 m/s keeps a high frequency and makes the training challenging. As the blocks with excessive wind speed are closed and unavailable for certain aircraft, consequently, the inaccessibility to blocks enlarges the conflict number from 10 to 66.

As the Dryden gust model only provides the view from wide-range regions, the fragmented wind field between urban buildings cannot be presented. Therefore, the CFD simulation at the building scale works perfectly as the complementary source to train a generalized solution. All numerical computations are accomplished with OpenFOAM [27]. As in Fig. 15, one snapshot of the CFD simulation result intuitively displays the complex wind flow, especially the street-canyon effect between buildings, where the wind speed increases or decreases dramatically. We select several points on the map to visualize the detailed data as in Fig. 16. Compared with the standard inlet velocities, the points near buildings, e.g. p_2 and p_3 , always have high fluctuations. And the point p_1 behind the building keeps lower intensity than the inlet face, which further pinpoints the complicated environment in the urban area. The point p_4 in the open area, in contrast, shares the same tendency as inlet velocity due to no shelter. As a consequence, the conflict number increases sharply from 10 to 182 when considering the high-resolution turbulence.

The conflict heat maps in Figs. 17 and 18 reveal the severe conditions after the Dryden wind and CFD wind model involved. The regional wind leads to discrete points, whereas the CFD model only affects flights in specific areas. To eliminate all conflicts and ensure a safe flight, the proposed MAA3C framework is applied to train with the Dryden model and CFD model, respectively.

Fig. 10 Spatio-temporal trajectory information associated with traversed sectors

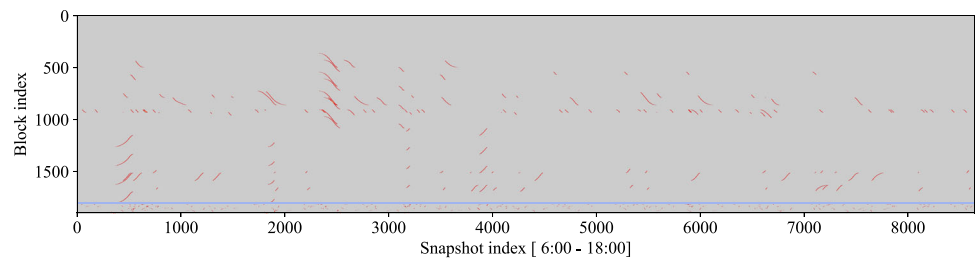


Fig. 11 3D traffic flow heat map

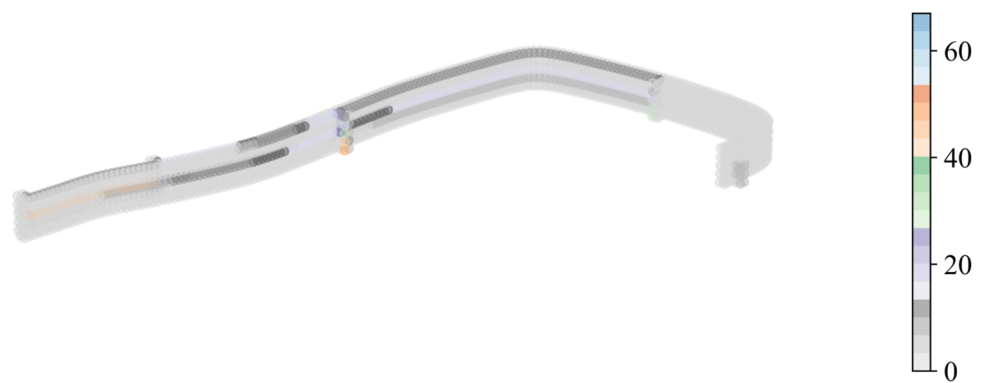
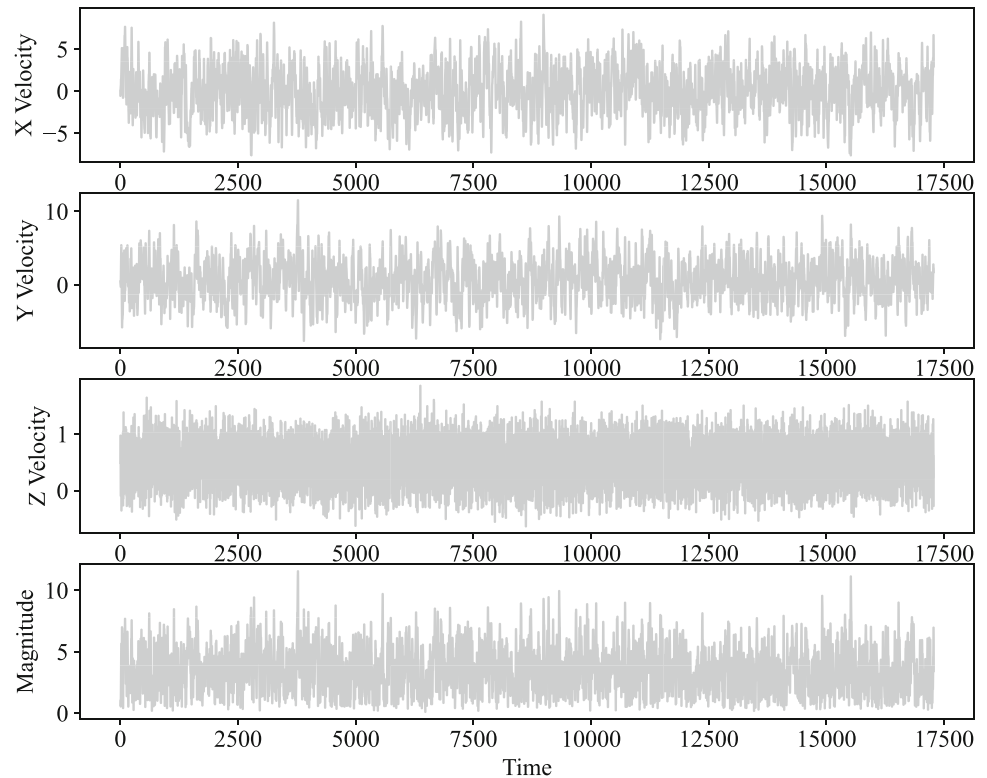


Fig. 12 Conflict heat map without wind



Fig. 13 Dryden gust velocities (Dryden-I)

The proposed architecture is implemented in PyTorch on Ubuntu 18.04 and Intel Xeon W-2123 CPU @3.60GHz×8. Two models are effectively trained with the same initial operation plan and different-resolution wind models until

convergence. And all parameters of 6 recurrent neural networks in Fig. 6 are initialized as in Table 3, where the block number is set to 700. The flight number can be reset from 100 to other numbers for various scenarios. In the training process, all cores are utilized to train the model asynchronously, and learning parameters in child threads are set to synchronize with the parent thread every 100 steps.

The return curve in Fig. 19 finally converges to a relatively stable value with all conflicts resolved in the Dryden-I model. Meanwhile, the fast converging curve in Fig. 20 also shows the ability of MAA3C to learn effective parameters for solving the conflict issue in complex turbulence.

4.4 Model Test

Although the performance of trained models shows the capability in the process of training, it is not adequate to directly announce the generalization of the MARL framework. As a comparison with the MAA3C solution, we replace all neural networks with continuous uniform distributions to assess learned parameters. The comparison model is entitled RANDOM, of which the architecture in Fig. 21 resembles the structure in Fig. 6. The cumulative distribution function of each *random* component is formulated with a uniform distribution as in Eq. 12.

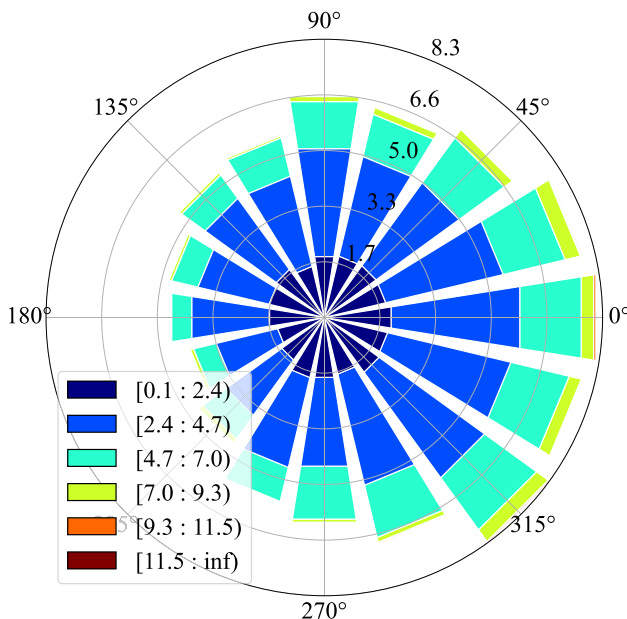
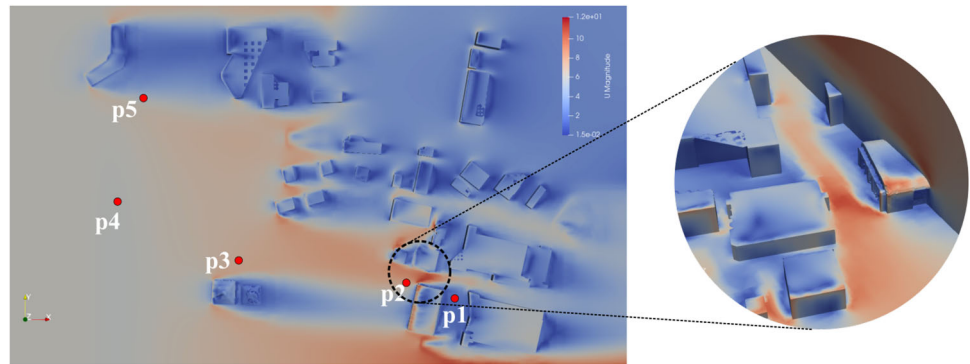
**Fig. 14** Windrose plot (Dryden-I)

Fig. 15 Simulated wind (CFD-wind-I) for training



Therefore, the ground delay, speed change, and cancellation parameters $[\delta\mu_k, \delta\sigma_k, \delta t, Ca]$ can be generated with Eq. 13.

$$f(x) = \begin{cases} \frac{1}{b-a} & a \leq x \leq b, \\ 0 & x < a \text{ or } x > b \end{cases} \quad (a = 0, b = 1) \quad (12)$$

$$[\delta\mu_k, \delta\sigma_k, \delta t, Ca] = f(x), \quad x \in [0, 1] \quad (13)$$

Especially, only those agents whose Ca value is larger than the average of all agents are canceled. In the meanwhile, the *mask* component in RANDOM is produced with the Bernoulli distribution $p(\text{mask} = 0) + p(\text{mask} = 1) = 1$ to make a selection between ground delay and speed change.

4.4.1 Test with Dryden Model

Different scenarios are simulated to evaluate the trained models. Firstly, another two Dryden wind profiles (Dryden-II and Dryden-III) are simulated with different average intensities in addition to Dryden-I. As exhibited in Fig. 22, the average magnitude of Dryden-II is lower than Dryden-I which is used for training. The wind direction in the north (90°) and south-east (300°) appears frequently during the

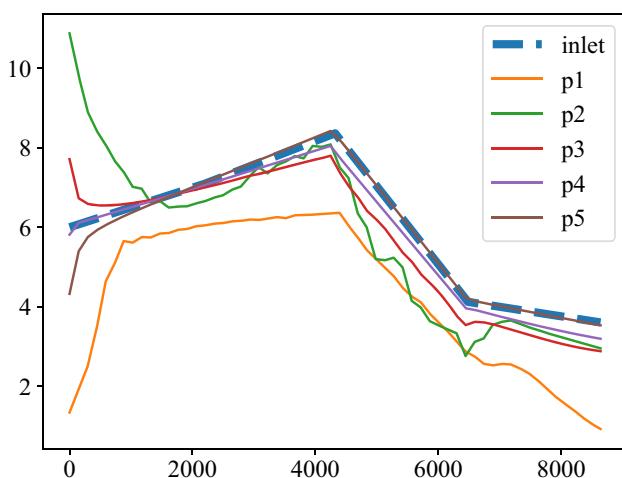


Fig. 16 Wind profiles (CFD-wind-I)

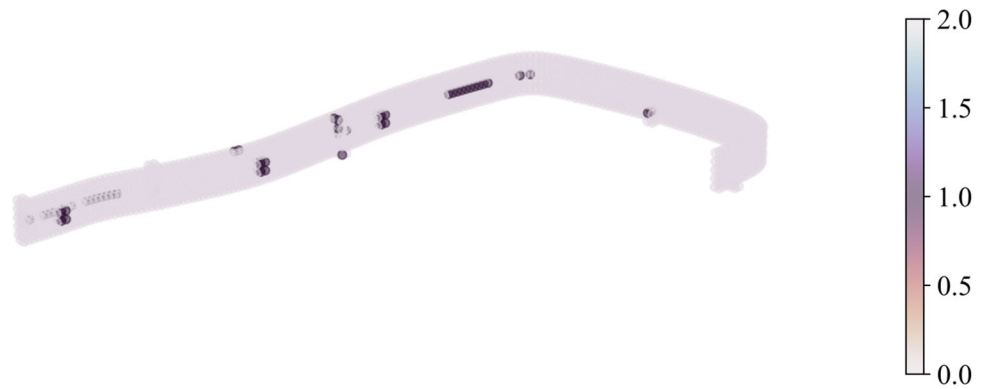
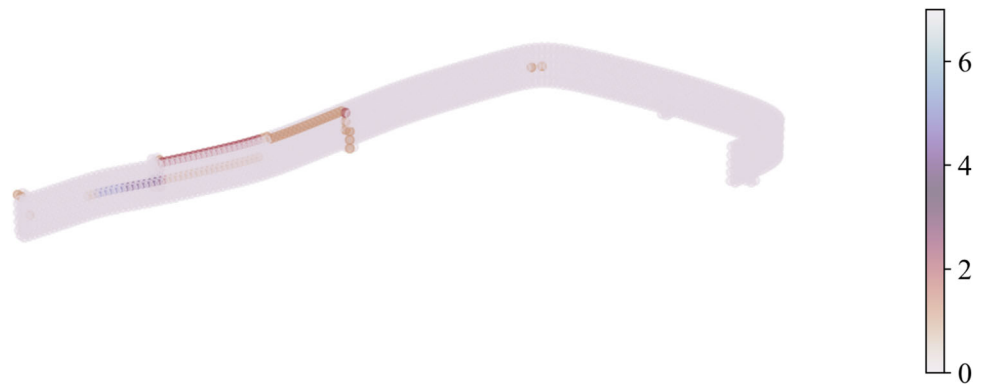
whole day according to the direction and intensity frequency statistics in Fig. 23. On the contrary, the Dryden-III presents greater average intensity and is always beyond 8 m/s in the northeast direction as depicted in Figs. 24 and 25.

Besides various wind data and test solutions, operation plans with 200, 300, and 400 flights are also produced to validate the scalability of our solution. Collecting all scenarios with different Dryden wind information, methods, and operation plan, 24 combinations are generated as test cases as listed in Table 4.

Cases 1-6 test the model only with new operation plans and without wind effect. The MAA3C and RANDOM will be applied to the same operation plan such as Cases 1 and 2, 3 and 4, etc. We can observe the test return values from Fig. 26. All cases with MAA3C obtain stable return values, in contrast to the lower mean values and larger variances when employing RANDOM. In particular, Cases 1 and 2 get 0 reward values because they are not necessary for deconfliction with no conflict.

For Cases 7-12, 13-18, and 19-24, three different Dryden wind profiles (Dryden-I, Dryden-II, and Dryden-III) are integrated into the environment, respectively. The total conflict number changes obviously with the wind influence. If the Dryden-I in Case 13-18 is regarded as the baseline, the Dryden-II decreases the environment complexity and gets small conflict numbers in Case 7-12. However, the Dryden-III makes the situation complicated as conflict numbers in Cases 19-24 are nearly doubled. Under those challenging circumstances, the return values in all test cases embody the stabilization of the MAA3C solution. With the same wind input in cases 7-12, 13-18, or 19-24, the growth in the flight number significantly degrades the return values. In this trend, the MAA3C solution can still stabilize the output in multiple trials. Another evident finding is that the low-intensity wind contributes to a high return value for the same operation plan, for instance, Case 3, 9, 15, and 21.

To analyze the detailed reason behind this, the specific delay and cancellation numbers are plotted in Fig. 27. The discovery is that number of flights to perform actions increases when the environment becomes much more

Fig. 17 Conflict heat map with Dryden-I**Fig. 18** Conflict heat map with CFD-wind-I**Table 3** Parameter setting of recurrent networks for 100 flights

Network Modules	Input Dimension	Output Dimension	Layer Num	Notes
NN1	[100, 700, 3]	[100, 700, 1]	2	
NN2	[100, 1, 700]	[100, 1, 4]	2	$[\delta\mu_1, \delta\sigma_1, \delta\mu_2, \delta\sigma_2]$
NN3	[100, 1, 700]	[100, 1, 1]	2	δt
NN4	[100, 1, 700]	[100, 1, 1]	2	<i>mask</i>
NN5	[100, 1, 700]	[100, 1, 1]	2	<i>cancellation</i>
NN6	[100, 1, 700]	[100, 1, 1]	2	<i>critic value</i>

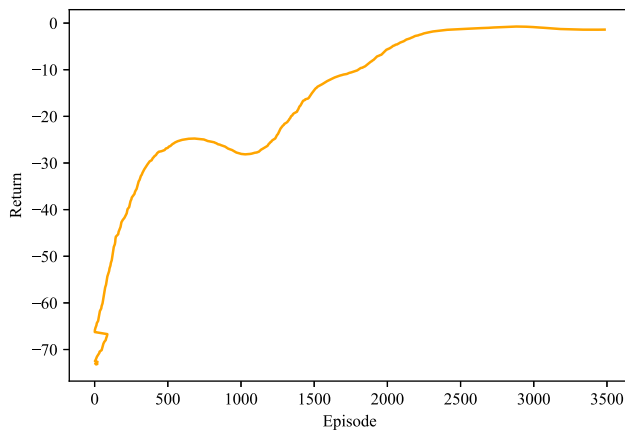


Fig. 19 Training curve with Dryden-I

stressful. For instance, for Cases 1, 3, and 5 in the same wind field, the number of flights to perform delay, speed change, and cancellation grows with the total flight number. Similarly, Cases like 5, 11, 17, and 23 with the same operation tend to process much more flights to cope with high-magnitude wind. We notice that the solution is inclined to select the cancellation when wind turbulence is complicated. From Case 13 to 24, the cancellation percentage ranges between 12%–26%.

We need to mention that RANDOM can also resolve all conflicts because of the involvement of random cancellations. The competitive results sometimes owe to the overall architecture which will ensure all conflicts are eliminated through iterative steps. We can note that the RANDOM can balance the selection between delay and cancellation since the generated random values can always keep a certain percentage for these two parts.

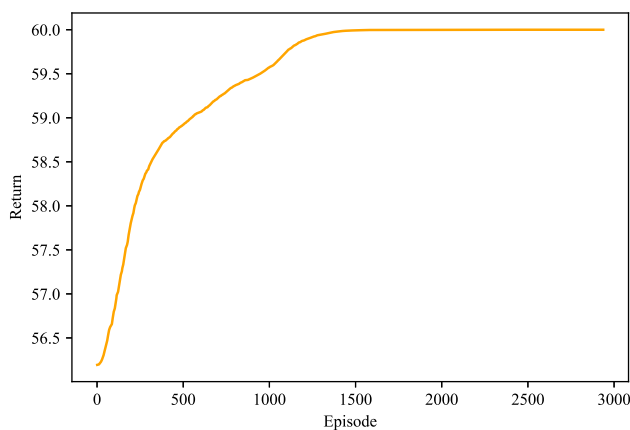


Fig. 20 Training curve with CFD-wind-I

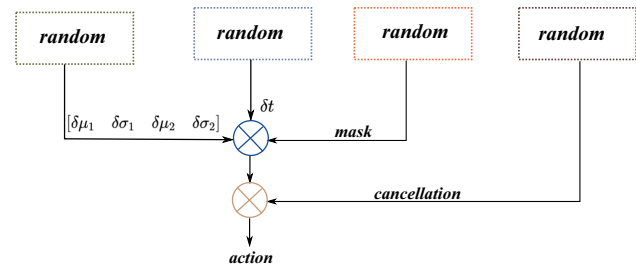


Fig. 21 Structure of RANDOM

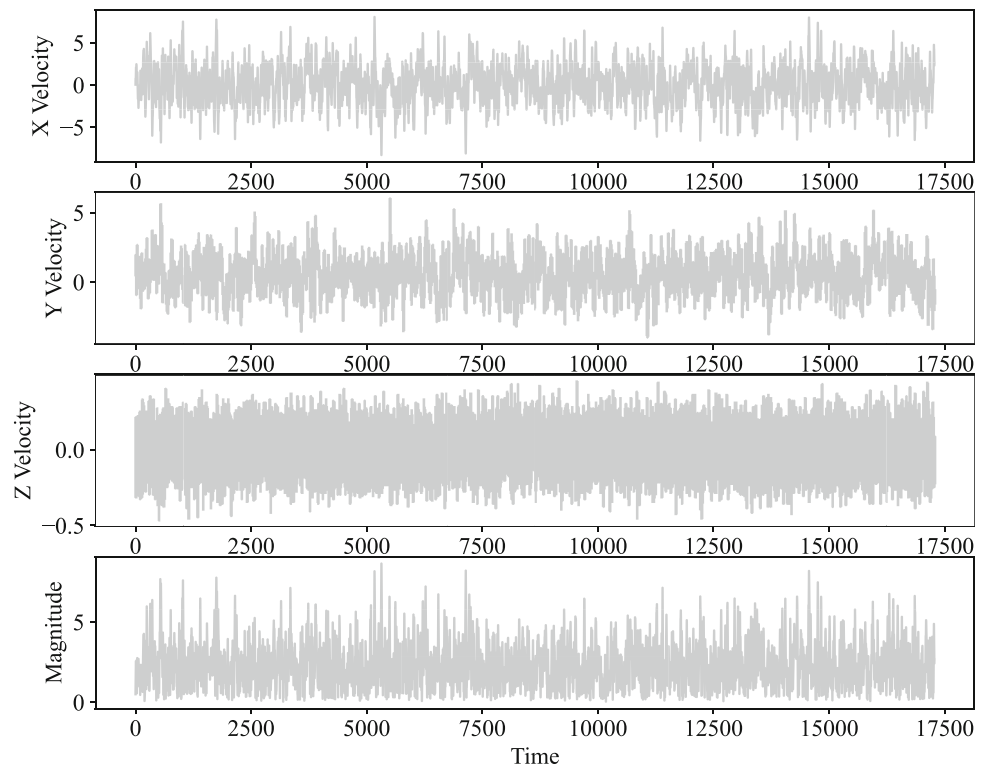
4.4.2 Test with CFD Simulation Model

Not only testing regional wind model but the simulated wind fields with CFD are also merged into the environment for refined urban operation. Along with the numerical model CFD-wind-I used for training, another new CFD model (CFD-wind-II) is simulated as in Fig. 28. The speed curves of the picked 5 points show the complex changing trends along with the dynamic inlet velocities. In comparison with the speed magnitude in Fig. 16, the wind intensity drops a lot as in Fig. 29.

Combining all possible tuples of wind input and study methods, there are different 18 cases indexed in Table 5 to validate the MAA3C solution in precise wind fields. Cases 25–30 are evaluating models on operation plans without wind as Cases 1–6 in Table 4. Cases 31–36 are totally brand-new scenarios with the wind model CFD-wind-II, and meanwhile, we can observe fewer conflicts compared with Cases 37–42.

The same as before, Cases 25, 26, 31, and 32 are not included since there is no conflict. The reward output from MAA3C presents a stable performance in multiple test episodes as displayed in Fig. 30. The mean and variance values are relatively better than RANDOM. And RANDOM fluctuates obviously in presence of the large conflict number contrary to MARL results. The number of flights that need to take action increases with the total conflict as plotted in Fig. 31. An obvious observation is that all cases with MAA3C tend to execute cancellation instead of delay or speed adjustment. The cancellation proportion keeps a low level which is less than 10% for Case 25–38, but for Case 39–42, it is floating from 10% to 19%.

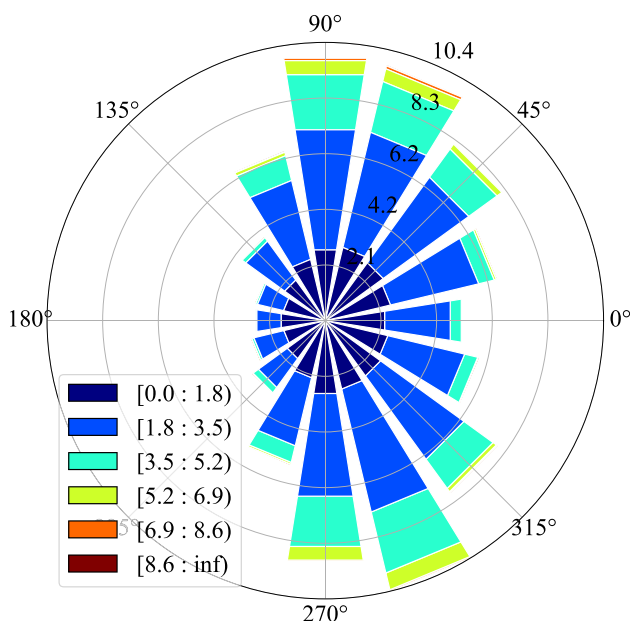
Those observations imply the critical role of the weight of each component in the reward function. If the cancellation is underestimated, conflicts in complicated scenarios can not be resolved. In this paper, conflict resolution is the priority, therefore, the cancellation is put to the decisive location in the architecture. Accordingly, delay and cancellation exhibit similar selection possibilities in simple scenarios. But with the environment being

Fig. 22 Dryden gust velocities (Dryden-II)

complicated, the MAA3C shifts the emphasis to only cancellation with the expectation to solve all conflicts. This learning ability is not RANDOM can achieve when coping with dynamic environments.

We need to mention that the simulated wind will not be the same as in the realistic and cannot cover all situations.

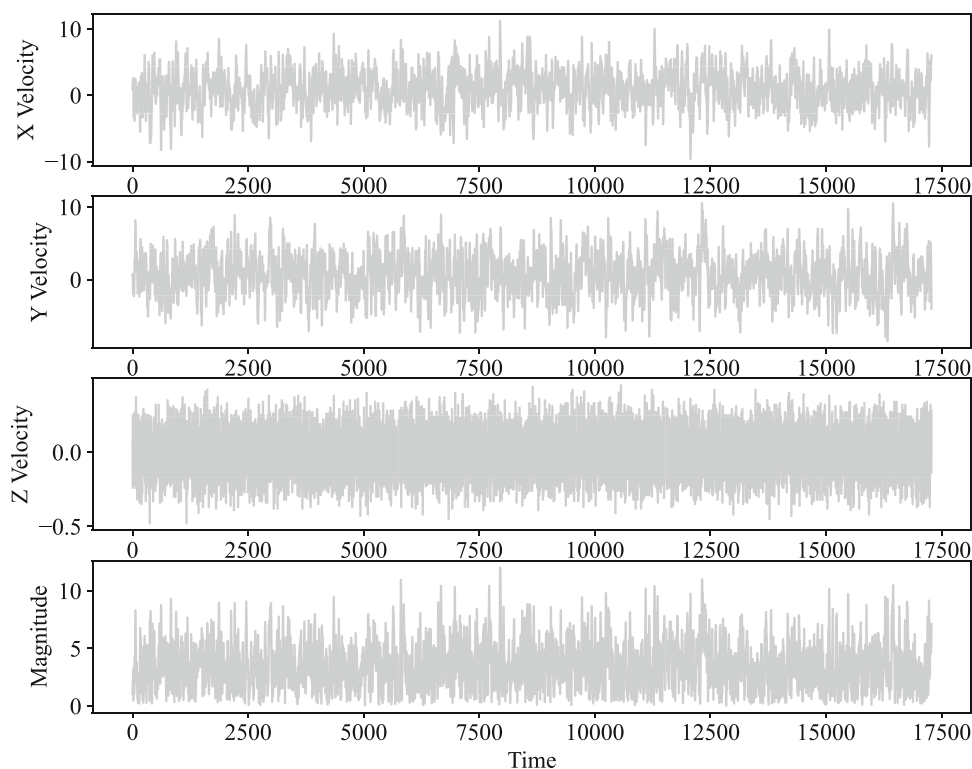
However, the objective of our solution is to be generalized for any operation schedule and wind input. Finally, the test results with different Dryden gusts and CFD wind models demonstrate the stability and robustness of the proposed MAA3C framework in presence of dynamic-wind turbulence. At the same time, the trained model can be flexibly applied to different operation plans with various flight numbers, which achieves scalable, resilient, and efficient operations for UAM.

**Fig. 23** Windrose for Dryden-II

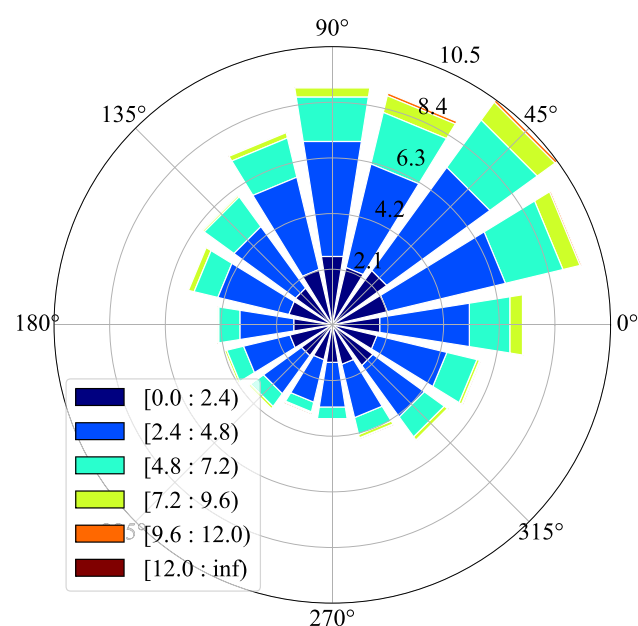
5 Conclusion

In this paper, the strategic conflict management problem, which comprises multi-resolution airspace organization, elaborated trajectory generation, and performance-based operation, is resolved by the MARL solution in presence of wind turbulence. The definitions of flexible airspace, adjustable trajectory with the GMM model, Dryden gust model, and CFD simulation model enable the MAA3C to learn effective parameters. Especially, the recurrent actor-critic networks in MAA3C allow scalable input with the various number of flights and airspace blocks. The crucial performance-related concept is integrated into each component of strategic conflict resolution to achieve efficient operations in urban regions.

This paper explores applying MARL for operations in high-dynamic environments. Conflict management follows

Fig. 24 Dryden gust velocities (Dryden-II)

a similar procedure in ATM or UTM and then involves some concepts in UAM ConOps. But the problem lay with focusing on the initial stage with air corridor structure rather than completely free flight, due to the lack of effective surveillance measures in low-altitude urban areas. It is possible to extend the proposed solution to broad airspace with more blocks. In reality, the wind flow changes

**Fig. 25** Windrose for Dryden-II**Table 4** Test Cases with Dryden Gust

Case	Wind Model	Method	Flight Num	Conflict Num
1	No Wind	MAA3C	50	0
2	No Wind	RANDOM	50	0
3	No Wind	MAA3C	100	2
4	No Wind	RANDOM	100	2
5	No Wind	MAA3C	200	29
6	No Wind	RANDOM	200	29
7	Dryden-II	MAA3C	50	1
8	Dryden-II	RANDOM	50	1
9	Dryden-II	MAA3C	100	19
10	Dryden-II	RANDOM	100	19
11	Dryden-II	MAA3C	200	35
12	Dryden-II	RANDOM	200	35
13	Dryden-I	MAA3C	50	25
14	Dryden-I	RANDOM	50	25
15	Dryden-I	MAA3C	100	51
16	Dryden-I	RANDOM	100	51
17	Dryden-I	MAA3C	200	186
18	Dryden-I	RANDOM	200	186
19	Dryden-III	MAA3C	50	61
20	Dryden-III	RANDOM	50	61
21	Dryden-III	MAA3C	100	227
22	Dryden-III	RANDOM	100	227
23	Dryden-III	MAA3C	200	439
24	Dryden-III	RANDOM	200	439

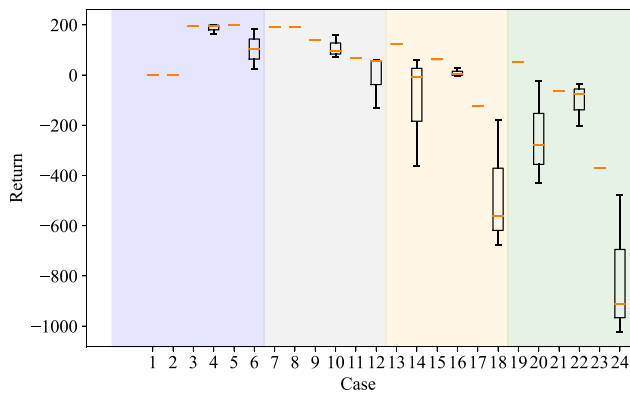


Fig. 26 Test results with Dryden model

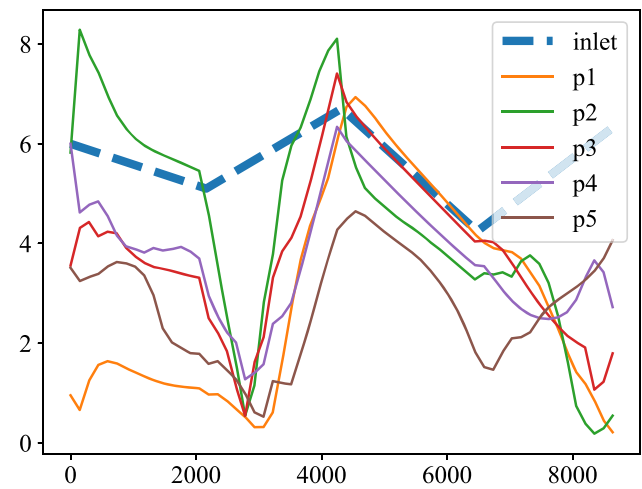


Fig. 29 Wind profiles for CFD-wind-II

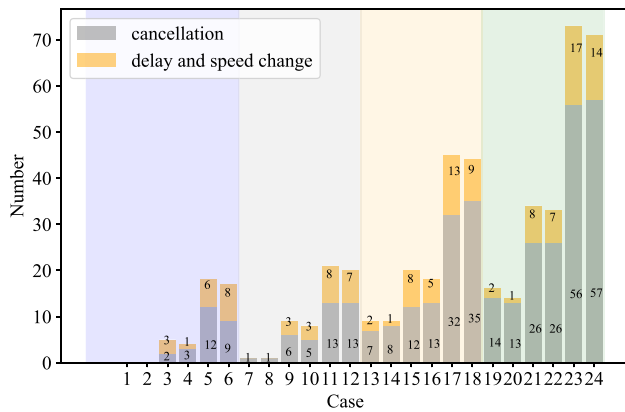


Fig. 27 Number of flights to take action

Table 5 Test Cases with CFD Wind

Case	Wind Model	Method	Flight Num	Conflict Num
25	No Wind	MAA3C	50	0
26	No Wind	RANDOM	50	0
27	No Wind	MAA3C	100	2
28	No Wind	RANDOM	100	2
29	No Wind	MAA3C	200	29
30	No Wind	RANDOM	200	29
31	CFD-wind-II	MAA3C	50	0
32	CFD-wind-II	RANDOM	50	0
33	CFD-wind-II	MAA3C	100	2
34	CFD-wind-II	RANDOM	100	2
35	CFD-wind-II	MAA3C	200	145
36	CFD-wind-II	RANDOM	200	145
37	CFD-wind-I	MAA3C	50	84
38	CFD-wind-I	RANDOM	50	84
39	CFD-wind-I	MAA3C	100	445
40	CFD-wind-I	RANDOM	100	445
41	CFD-wind-I	MAA3C	200	918
42	CFD-wind-I	RANDOM	200	918

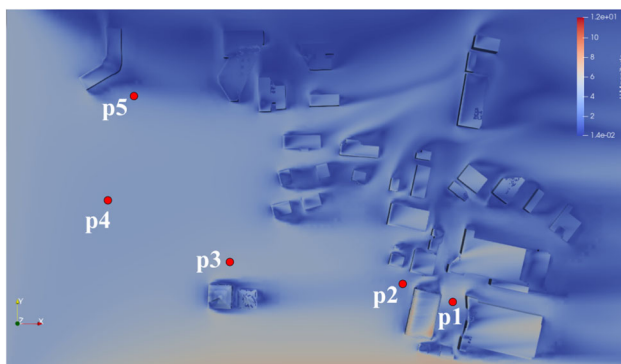


Fig. 28 Simulated wind (CFD-wind-II)

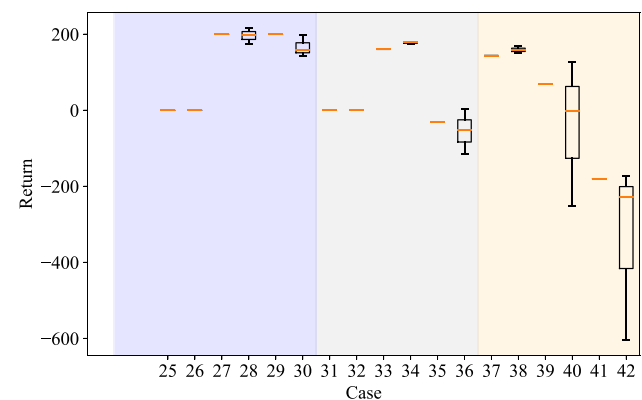


Fig. 30 Test results with CFD wind data

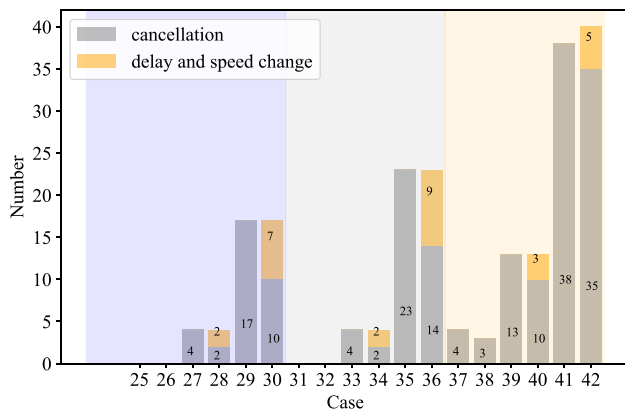


Fig. 31 Number of flights to take action

frequently and is hard to be predicted, especially between buildings. Nevertheless, our MAA3C solution still offers the applicable ability thanks to its generalization feature.

Author Contributions Cheng Huang contributed to the algorithm design, implementation, and writing of this paper; Ivan Petrunin and Antonios Tsourdos contributed to the result analysis and revision of the manuscript.

Funding This research was partially supported by grants from the Funds of China Scholarship Council (202008420248).

Availability of data and materials The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Code Availability The code is available in <https://github.com/ChengHuang-CH/uam-maa3c.wind.turbulence.git>.

Declarations

Ethics approval No applicable as this study does not contain biological applications.

Consent to participate All authors of this research paper have consented to participate in the research study.

Consent for Publication All authors of this research paper have read and approved the submitted version.

Conflict of Interests The authors have no relevant financial or non-financial interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. De l'aviation civile internationale, O.: Global Air Traffic Management Operational Concept (ICAO) (2005)
2. Unmanned aircraft system (uas) traffic management (utm) - concept of operations v2.0. Tech. Rep. 1-68 federal aviation administration (2020)
3. Kopardekar, P., et al.: Unmanned Aircraft System Traffic Management (utm) Concept of Operations, pp. 1–16. AIAA, Reston (2016)
4. Urban air mobility (uam) - concept of operations v1.0. Tech. Rep. 1-37 federal aviation administration (2020)
5. Hill, B.P., et al.: Uam vision concept of operations (conops) uam maturity level (uml) 4 (2020)
6. Drone dcB concept and process: Tech. Rep. 1-261, SESAR (2021)
7. Acevedo, J.J., Castaño, Á.R., Andrade-Pineda, J.L., Ollero, A.: A 4d Grid Based Approach for Efficient Conflict Detection in Large-Scale Multi-uav Scenarios, pp. 18–23. IEEE, Cranfield, UK (2019)
8. Mohamed Salleh, M.F.B., et al.: In Preliminary concept of adaptive urban airspace management for unmanned aircraft operations 2260 (2018)
9. Pang, B., Dai, W., Ra, T., Low, K.H.: A Concept of Airspace Configuration and Operational Rules for uas in Current Airspace, pp. 1–9. IEEE, San Antonio, TX, USA (2020)
10. Dai, W., Pang, B., Low, K.H.: Conflict-free four-dimensional path planning for urban air mobility considering airspace occupancy. Aerospace Science and Technology 107154 (2021)
11. Bertsimas, D., Patterson, S.S.: The air traffic flow management problem with enroute capacities. Oper. Res. **46**(3), 406–422 (1998)
12. Zhang, Y., Su, R., Li, Q., Cassandras, C.G., Xie, L.: Distributed flight routing and scheduling for air traffic flow management. IEEE Trans. Intell. Transp. Syst. **18**(10), 2681–2692 (2017)
13. Spatharis, C., et al.: Multiagent reinforcement learning methods to resolve demand capacity balance problems 1–9 (2018)
14. Huang, C., Xu, Y.: Integrated Frameworks of Unsupervised, Supervised and Reinforcement Learning for Solving Air Traffic Flow Management Problem, pp. 1–10. IEEE, San Antonio, TX, USA (2021)
15. Spatharis, C., et al.: Hierarchical multiagent reinforcement learning schemes for air traffic management. Neural Comput. & Applic. 1–13 (2021)
16. Xie, Y., Gardi, A., Sabatini, R.: Reinforcement Learning-Based Flow Management Techniques for Urban Air Mobility and Dense Low-Altitude Air Traffic Operations, pp. 1–10. IEEE, San Antonio, TX, USA (2021)
17. Pham, D.-T., Tran, N.P., Goh, S.K., Alam, S., Duong, V.: Reinforcement Learning for Two-Aircraft Conflict Resolution in the Presence of Uncertainty, pp. 1–6. IEEE, Danang, Vietnam (2019)
18. Tran, P.N., Pham, D.-T., Goh, S.K., Alam, S., Duong, V.: An interactive conflict solver for learning air traffic conflict resolutions. Journal of Aerospace Information Systems **17**(6), 271–277 (2020)
19. Abichandani, P., Lobo, D., Ford, G., Bucci, D., Kam, M.: Wind measurement and simulation techniques in multi-rotor small unmanned aerial vehicles. IEEE Access **8**, 54910–54927 (2020)
20. Kadaverugu, R., Purohit, V., Matli, C., Biniwale, R.: Improving accuracy in simulation of urban wind flows by dynamic downscaling wrf with openfoam. Urban Climate **38**, 100912 (2021)
21. Allison, S., Bai, H., Jayaraman, B.: In Modeling trajectory performance of quadrotors under wind disturbances. 1237 (2018)

22. Beard, R.W., McLain, T.W.: *Small Unmanned Aircraft*. Princeton University Press, Princeton (2012)
23. Zhang, K., Yang, Z., Başar, T.: Multi-agent reinforcement learning: a selective overview of theories and algorithms. *Handbook of Reinforcement Learning and Control* 321–384 (2021)
24. Papoudakis, G., Christianos, F., Rahman, A., Albrecht, S.V.: Dealing with non-stationarity in multi-agent deep reinforcement learning. arXiv:1906.04737 (2019)
25. Mnih, V., et al.: *Asynchronous Methods for Deep Reinforcement Learning*, pp. 1928–1937. PMLR, New York (2016)
26. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: CARLA: An open urban driving simulator. 1–16 (2017)
27. Jasak, H.: Openfoam: open source cfd in research and industry. *International Journal of Naval Architecture and Ocean Engineering* 1(2), 89–94 (2009)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Cheng Huang is currently a Ph.D. student in Aerospace at Cranfield University. He received his M.Sc. and B.Eng. in Guidance, Navigation and Control from Northwestern Polytechnical University. His main research interests include aerospace, urban air mobility, sensor fusion, and artificial intelligence.

Ivan Petrunin received his MSc degree in Design and Manufacturing of Electronic Equipment from National Technical University of Ukraine in 1998, PhD degree in Signal Processing for Condition Monitoring from Cranfield University, UK, in 2013. He is currently a Senior Lecturer in digital signal processing with the Centre for Autonomous and Cyber-Physical Systems, Cranfield University. His current research interests include applied signal processing for autonomous systems with particular emphasis on communications, navigation, and surveillance aspects, such as cognitive communication schemes, assured multi-sensor navigation, and remote sensing.

Antonios Tsourdos is Head of the Center of Autonomous and Cyber-Physical Systems and Director of Research – SATM. He is Chair of the IFAC Technical Committee of Aerospace and editorial member for a number of journals including IEEE Transaction of Aerospace & Electronics Systems, Aerospace Science and Technology, IMechE Journal of Aerospace Engineering, Control Engineering Practice, and International Journal of Systems Science. He has been involved in a number of projects including EPSRC, H2020, IUK, Sesar, and Clean Sky funded projects focusing on fight envelope protection, data analytics, system architectures, urban air mobility, communication and networking, cyber-physical aerospace systems, and avionics. He has published widely and has been the recipient of the UK MoD Autonomy Grand Challenge award and the IET Team Innovation award.



DSPACE

<https://dspace.org/>

Strategic conflict management using recurrent multi-agent reinforcement learning for urban air mobility operations considering uncertainties

Huang, Cheng; Petrunin, Ivan; Tsourdos, Antonios

2023-01-26

Springer

<https://doi.org/10.1007/s10846-022-01784-0>

Huang C, Petrunin I, Tsourdos A. (2023) Strategic conflict management using recurrent multi-agent reinforcement learning for urban air mobility operations considering uncertainties. Journal of Intelligent and Robotic Systems, Volume 107, February 2023, Article number 20