



# Decentralized Coordination of Autonomous Traffic Through Advanced Air Mobility Corridors

Jasmine Jerry Aloor\* and Hamsa Balakrishnan†  
*Massachusetts Institute of Technology, Cambridge, MA 02139, USA*

The use of dedicated corridors for Advanced Air Mobility (AAM) traffic is one of the most commonly proposed pathways to integrating them into existing airspace operations. Most prior research has focused on the design of networks of AAM corridors and conflict resolution for aircraft within corridors. It is also generally believed that while attractive from an implementation perspective, corridor-based operations may be inefficient, especially in the absence of centralized traffic management.

In this paper, we show that contrary to this belief, it is possible for autonomous aircraft to learn to self-organize into corridor flows in decentralized settings. We illustrate our approach using scenarios in which fixed-wing aircraft need to safely and efficiently traverse (1) a single corridor with metering after the exit, (2) a sequence of two consecutive corridors, and (3) a corridor that splits into two. We find that in decentralized settings with only local information, the aircraft are able to conform to the corridor boundaries more than 94% of the time and reach their goal in a relatively efficient manner. Furthermore, tactical interventions to handle violations of the separation minimum are needed only infrequently in low- and medium-density settings. However, such tactical interventions become more frequently necessary only when traffic density is high.

## I. Nomenclature

$N$	=	number of agents
$D$	=	dimension of the state
$S$	=	state space of the environment
$p$	=	position of agents
$\theta$	=	heading of agent
$v$	=	speed of agent
$\omega$	=	angular velocity of agent
$a$	=	longitudinal acceleration
$O$	=	neighborhood observations
$\mathcal{A}$	=	action space for agents
$\mathcal{G}$	=	graph network formed by the entities in the environment
$P(s' s, A)$	=	transition probability from $s$ to $s'$ given the joint action $A$
$\mathcal{R}$	=	joint reward function
$C$	=	penalty applied
$\gamma$	=	discount factor

## II. Introduction

CORRIDOR-BASED operations are widely regarded as one of the most promising approaches to integrating Advanced Air Mobility (AAM) traffic into existing airspace systems [1–5]. Existing air traffic management procedures cannot scale as the number of AAM operations grows, motivating the development of dedicated air corridors equipped with technologies that can support the new entrants [6]. The increased adoption of the AAM corridor concept has motivated work on the design, analysis, and even logistics of corridor-based AAM operations [7–9].

\*Ph.D. Candidate, Department of Aeronautics and Astronautics, and AIAA Student Member.

†Associate Dean of Engineering, William E. Leonhard (1940) Professor, Department of Aeronautics and Astronautics, and AIAA Fellow.

The development of more automated approaches to air traffic management has been a long-standing area of research due to the growth in air traffic demand over the past few decades and the resulting increase in ATC workload [10, 11]. Recently, the increase in onboard sensing and computing resources, the expectation that AAM aircraft will be highly autonomous, and advances in artificial intelligence and machine learning (AI/ML) technologies have driven the study of learning-based methods for AAM traffic management. In particular, the challenge of operating multiple autonomous aircraft in the airspace has led to the consideration of Multi-agent Reinforcement Learning (MARL) as a solution methodology in the AAM context. Prior work has used MARL approaches for inter-aircraft conflict resolution [12–14], trajectory tracking [15], and demand-capacity balancing [16].

The projected increase in demand has also motivated the development of strategic air traffic management algorithms to improve the efficiency of operations. While the desire to provide more flexibility to operators has motivated decentralized or distributed solution approaches, the approaches have largely relied on the optimization of less-constrained 4D trajectory-based operations [17, 18]. The rationale behind these works has been that the constraints imposed on trajectories by AAM corridors can lead to very inefficient use of scarce airspace resources [17]. However, a potential way to improve the efficiency of an aircraft flow through a corridor is through convoy formation, where vehicles collectively maintain an optimal separation with the vehicles in front of and behind them, as they traverse the corridor. Convoy formation or platooning has been studied in the context of railways [19], roadways [20, 21], and even AAM [22], because of its potential to considerably reduce operator workload [23] and fuel [24].

The long-term vision is that Air Navigation Service Providers (e.g., the FAA) will not be responsible for providing air traffic control (ATC) services within AAM corridors [1]. This vision raises the question of whether multiple aircraft can autonomously navigate through dedicated corridor structures simultaneously in a safe and efficient manner, without centralized coordination. To the best of our knowledge, there has been very limited work on the decentralized coordination of traffic through AAM corridors. In [25], the authors proposed a Multi-Attribute Decision-Making scheme for aircraft to maintain self-separation within a corridor. A Model Predictive Control (MPC) based centralized approach to structured airspace networks was proposed in [26]. By contrast, [27] and [28] considered the problems of coordination in merges and intersections, using MARL and a merge-assist strategy based on road traffic, respectively. Finally, very recent work has developed a hybrid transformer-based MARL architecture for traffic coordination within air corridors [29]; however, this work assumes simplified eVTOL dynamics (albeit in 3D) and that the aircraft within a corridor traverse parallel to each other. In other words, inter-aircraft separation of the traffic passing through the corridor is not considered.

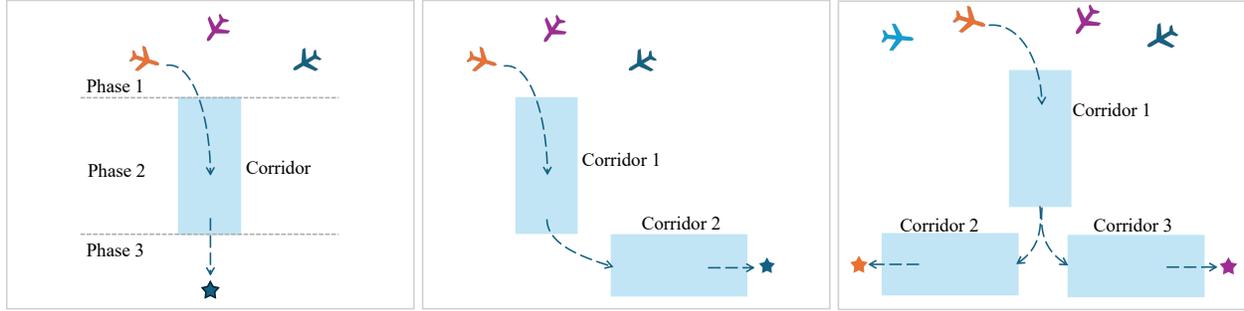
Recently, we proposed InforMARL, a MARL architecture based on Graph Neural Networks (GNNs) that enables multi-agent navigation and control in limited-information settings [30]. We also extended this approach to show that agents could learn to balance fairness (load balancing between agents) and efficiency in a number of contexts, including task coverage and formation [31]. In this paper, we address the following question: **Can autonomous fixed-wing aircraft learn to self-organize in a decentralized manner while traversing through AAM corridors?**

We show that **the answer to this question is yes**, by developing a centralized training/decentralized execution approach based on our prior work. We consider three canonical scenarios, shown in Fig. 1: (a) A single corridor with a post-exit metering point that all aircraft must enter, traverse and exit; (b) two consecutive corridors, where the aircraft must enter then exit each one in succession; and (c) a split, where the traffic enters and travels through a corridor (Corridor 1 in Fig. 1c), exits it and then splits, with some aircraft entering Corridor 2 and the rest, Corridor 3. We choose these configurations as they form the building blocks of general networks of air corridors.

Our simulations show that in all three corridor configurations, aircraft are able to conform to corridor boundaries more than 94% of the time and reach their final goals in a timely manner, even in highly-congested environments. As our approach does not set hard constraints on inter-aircraft separation minima, tactical interventions may be needed when two aircraft approach within this specified value. Our experiments show that such tactical interventions are needed less than 8% of the time in low- and medium-density environments, and about 17% of the time in highly-congested environments with split corridors.

## A. Outline

The remainder of this paper is organized as follows. Sec. III presents the models of agents, aircraft kinematics, and environment. Sec. IV describes the MARL training setup, including the reward functions used. Sec. V presents our experiments and discusses the key results, and we conclude in Sec. VI.



(a) Single corridor with post-exit metering (b) Two consecutive corridors placed one after another. (c) Split corridor, aircraft entering a single corridor then splitting into two streams.

**Fig. 1** Illustrations of corridor-based scenarios.

### III. Models of environment and agents

In this section, we describe the environment and dynamics used in our MARL training setup.

#### A. Preliminaries

Our environment consists of aircraft modeled as agents, goals, and obstacles, following the InforMARL [30] framework.

We model our system as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) defined by the tuple  $\langle N, S, O, \mathcal{A}, \mathcal{G}, P, R, \gamma \rangle$ , where:

- $N$  is the number of agents
- $s \in S = \mathbb{R}^{N \times D}$  is the state space of the environment, with  $D$  as the dimension of the state
- $o^{(i)} = O(s^{(i)}) \in \mathbb{R}^d$  is the neighborhood observation for agent  $i$
- $a^{(i)} \in \mathcal{A}$  is the action space for agent  $i$ .
- $g^{(i)} \in \mathcal{G}(s; i)$  is the graph network formed by the entities in the environment with respect to agent  $i$
- $P(s'|s, A)$  is the transition probability from  $s$  to  $s'$  given the joint action  $A$
- $R(s, A)$  is the joint reward function
- $\gamma \in [0, 1)$  is the discount factor

The objective is to find a policy  $\Pi = (\pi^{(1)}, \dots, \pi^{(N)})$ , where  $\pi_{\theta}^{(i)}(a^{(i)} | o^{(i)}, g^{(i)})$  for agent  $i$  selects action  $a^{(i)}$  based on its graph network  $g^{(i)}$  and the neighborhood observation  $o^{(i)}$ .

#### 1. Agent dynamics

We represent the aircraft dynamics using the mathematical model of a fixed-wing aircraft:

$$\dot{x} = v \cos \theta, \quad \dot{y} = v \sin \theta, \quad \dot{\theta} = \omega, \quad \dot{v} = a, \quad (1)$$

We define the agent state as  $s^{(i)} = [x; y; \theta; v]$ , which are the x- and y-positions, heading, and speed of the agent, respectively. The action space represents the angular velocity and longitudinal acceleration  $a^{(i)} = [\omega, a]$ . Each agent's speed is limited to the range of  $[v_{\min}, v_{\max}]$ . Here,  $v_{\max} > v_{\min} > 0$ , and the agents cannot move in reverse. The action space is defined as  $\mathcal{A} = [-\omega_{\max}, \omega_{\max}] \times [a_{\min}, a_{\max}]$ . When an agent reaches a goal, it is marked as 'done', and the agent is assumed to have stopped. When all agents arrive at their goals and are 'done', we end the episode and start the next one.

#### B. Corridor discretization phases

We denote the corridors by rectangular lanes in the environment with length  $L$ , width  $w$ , and alignment angle  $\theta^{\text{corridor}}$ . We divide the corridor navigation process into three phases for ease of representation.

- 1) **Phase 1:** Pre-corridor phase. Agents approach the corridor while maintaining inter-agent spacing and merge as needed to enter the corridor.
- 2) **Phase 2:** In-corridor phase. Agents maintain the line formation and spacing within the corridor.

3) **Phase 3:** Post-corridor phase. Agents exit and disperse toward their respective goals.

## IV. Methodology

We investigate a structured environment MARL navigation problem. The agents are tasked with navigating from one end of the environment to the other end. There exist corridors in the middle of the environment that act as a channel for the agents to use while navigating from one half of the environment to the other. The agents must traverse these corridors while maintaining a desired inter-agent separation. Once they cross the corridor, they have to arrive at the goal with a specified metering rate. An example is shown in Fig. 1a. We train our models using three agents per episode.

### A. Agent observations

Each agent’s local observation vector  $o^{(i)}$  captures essential information about its own state, nearby agents, goals, and its designated corridor. The agent’s heading,  $\theta$ , and speed,  $v$ , are recorded in a global frame of reference. The relative positions of the assigned goal and the two nearest neighboring agents are provided in the agent’s body-fixed reference frame, which is aligned with its heading direction. We incorporate the details of the corridor, such as the agent’s normalized longitudinal distance from the entrance  $d^{\text{entrance}}/L$  and exit  $d^{\text{exit}}/L$ , the lateral offset from the centerline of the corridor  $d^{\text{center}}/w$ , the alignment of agent’s heading to the corridor  $\Delta\theta = \theta^{\text{agent}} - \theta^{\text{corridor}}$ , and the agent’s current phase  $\phi \in \{1, 2, 3\}$  within the corridor. The final ego observation vector  $o^{(i)}$  can be represented as  $o^{(i)} = [\theta_i, v_i, p_i^{\text{goal}_1}, p_i^{\text{n}_1}, p_i^{\text{n}_2}, d_i^{\text{entrance}}/L, d_i^{\text{exit}}/L, d_i^{\text{center}}/w, \Delta\theta, \phi_i]$ .

Each agent also has its neighborhood information aggregated into a graph observation vector  $x_j$ , which is then processed by a graph neural network (GNN). Through graph message passing, the GNN can encode and incorporate information about agents and goals that lie beyond the agent’s direct sensing range, and allows the framework to scale to any number of agents. For each agent  $i$ ,  $x_j^{(i)} = [p_i^j, v_i^j, p_i^{\text{goal}_1, j}, \text{entity\_type}(j)]$  where  $p_i^j, v_i^j, p_i^{\text{goal}_1, j}$  are the *relative* position, velocity, and position of the nearest goal of the entity at node  $j$  with respect to agent  $i$ , respectively. The variable  $\text{entity\_type}(j) \in \{\text{agent}, \text{goal}\}$  specifies the type of entity at node  $j$ . When an agent is marked as ‘done’, we remove the agent from the graph to prevent it from being considered in any potential conflict calculations.

The combined input of the ego observation vector and the GNN-encoded output of neighborhood entities is provided to the policy to produce an action.

### B. Reward function design

During training time, each agent is given a reward at every time step. The reward function is designed based on the following desired behaviors.

#### 1. Separation maintenance

*Reward for maintaining proper inter-agent spacing.* To encourage agents to maintain an adequate inter-agent distance, we measure how close an agent is to its nearest neighbors located in front and behind it and compare it to the desired separation value ( $d_s$ ). Any inter-agent distance that is lower than this desired spacing value is penalized.

$$\mathcal{R}_{\text{separation}} = \sum_{j \in \{\text{front}, \text{back}\}} \max(0, d_s - \|p^{(i)} - p^{(j)}\|) \quad (2)$$

#### 2. Phase transitions

*Reward encouraging agents to smoothly move between phases.* At the start of the episode, each agent is in Phase 1 and is provided with information about its relative position with respect to the entrance of its current corridor. The agent is provided with a continuous reward that is based on the distance to the corridor’s entrance, which incentivizes them to move to the corridor. A small reward is added when the agent is near the corridor entrance to encourage heading angle alignment. When the agent transitions into Phase 2, we provide a bonus reward for completing the transition and shift the reward computation reference to the exit of the corridor. This incentivizes the agent to move toward the exit of the corridor. We provide a similar bonus phase transition reward once an agent crosses the corridor. Once an agent is in Phase 2, we provide it with a goal-reaching reward that is proportional to the distance to the agent’s chosen goal. At each phase  $\phi$  the reward can be computed as

$$\mathcal{R}_{\text{phase}(\phi)} = -\|p^{\text{corridor}} - p^{(i)}\| + \kappa \mathcal{R}_{\text{transition}} \quad (3)$$

$\kappa$  is set to 1 if the agent made the phase transition for the first time. To prevent agents from skipping the correct phase transition order or trying to move in the reverse phase order, we penalize the agent with an incorrect phase transition penalty  $-C_{\text{phase}}$ .

### 3. Conflict avoidance

*Penalty for approaching too close to other agents or obstacles.* We also penalize agents colliding with other agents in the environment using a conflict penalty  $-C_{\text{conflict}}$ . This penalty is applied when the inter-agent separation becomes lower than the desired minimum separation distance  $d_s$ .

### 4. Goal reaching reward

*Reward encouraging a quick approach to the goal.* When an agent successfully reaches its goal (indicated by  $\rho$ ), it receives a one-time goal-reaching reward  $\mathcal{R}_{\text{goal}}$ . The indicator variable  $\rho$  is 1 if the agent reached the assigned goal and was previously not at the goal; otherwise, 0.

The overall reward provided to each agent at every time step then becomes,

$$\mathcal{R}_{\text{total}} = \mathcal{R}_{\text{separation}} + \mathcal{R}_{\text{phase}(\phi)} + \rho\mathcal{R}_{\text{goal}} - C_{\text{phase}} - C_{\text{conflict}} \quad (4)$$

This reward design ensures that the agents learn to maintain a desired separation while being able to follow the various phases of the corridor and reach the desired goal.

## V. Experiments and results

### A. Parameter values

We consider the following parameters for the environment and aircraft kinematics. The parameter values are chosen to broadly align with industry standards and proposed corridor designs [32–35]. We note that these particular parameter values are chosen to merely demonstrate our methodology and can be changed as technology and standards evolve. Additionally, the minimum inter-aircraft separation could be adjusted to a fuel-optimal (or some other preferred) separation to achieve convoy formation in high-demand scenarios.

- 1) Minimum groundspeed,  $v_{\text{min}}$ : 60 knots/s = 111 km/h
- 2) Maximum groundspeed,  $v_{\text{max}}$ : 175 knots/s = 324 km/h
- 3) Minimum acceleration,  $a_{\text{min}}$ :  $-1 \text{ m/s}^2$
- 4) Maximum acceleration,  $a_{\text{max}}$ :  $2 \text{ m/s}^2$
- 5) Maximum angular velocity,  $\omega_{\text{max}}$ : 0.075 rad/s
- 6) Length of air corridor: between 2 km to 1.2 km
- 7) Width of air corridor: 0.6 km
- 8) Minimum inter-aircraft separation: 0.3 km
- 9) Goal threshold distance: 0.2 km
- 10) Metering rate at goal: 10 aircraft/min
- 11) Episode length: 150 s

### B. Experimental setup

To evaluate our approach, we define the following test scenarios, each designed to assess different aspects of corridor-based aircraft navigation.

#### 1. Single corridor with post-exit metering point

In this scenario, illustrated in Fig. 1a, aircraft follow a structured, metered entry into a single corridor. The objective is to evaluate how well the method regulates the flow of aircraft, ensuring smooth entry and maintaining safe separation between them. This is the simplest scenario and serves as a baseline to verify that autonomously operated aircraft can traverse a corridor and satisfy metering requirements at a point downstream.

For each episode, we define the air corridor to be at the center of the environment. We vary the initial position of each aircraft for better generalization. Each aircraft is required to maintain the desired inter-aircraft separation minimum

relative to all other aircraft. The vehicles are tasked to go to the goal at a fixed metering rate. Once the aircraft reaches its goal, it is stopped and not considered for any further actions for that episode.

### 2. *Two consecutive corridors*

This scenario, as shown in Fig. 1b, increases complexity by requiring agents to traverse multiple consecutive corridors. The agents are tasked with going through Corridor 1 first before entering the horizontal Corridor 2.

In each episode, each aircraft is provided with information about Corridor 1 via its local observations. Once the aircraft traverses the first corridor, we provide information about the second corridor. This allows the multi-agent system to be scaled up to multiple corridors through a simple repetition of the corridor setup. Once the aircraft crosses the second corridor, it proceeds to its destination and comes to a stop there.

### 3. *Split corridor*

This scenario, depicted in Fig. 1c, adds to the previous scenario by dynamically allocating a second corridor to each aircraft once it traverses Corridor 1. This scenario demonstrates the ability of our method to accommodate forks in the route. In this scenario, every aircraft has a common goal of traversing through Corridor 1. As it exits the corridor, each aircraft is randomly assigned to either Corridor 2 or Corridor 3. The aircraft stops when it reaches the goal at the end of its assigned corridor.

## C. Performance metrics

We run evaluations for each scenario for 100 randomly generated episodes. For each scenario, we consider traffic flows with 3, 5, and 10 aircraft. We note that the corridors are either 1.6 or 3.2 km in length; consequently, the aircraft counts correspond to low, medium, and very high density operations. We calculate the following performance metrics for each scenario in our experiments.

### 1. *Conformance to corridor boundaries ( $C\%$ )*

This metric measures how well aircraft adhere to the designated corridor boundaries during their traversal. Conformance is evaluated based on the deviation of an aircraft's trajectory from the corridor boundaries, ensuring the aircraft remains within the allowable corridor width.  $C\%$  is calculated as the average percentage of time that each aircraft stays within the corridor boundaries while traversing through it, averaged over all aircraft and 100 episodes. High conformance to corridor boundaries is equivalent to the required navigation performance and reflects the effectiveness of the AAM corridor definition.

### 2. *Success rate ( $S\%$ )*

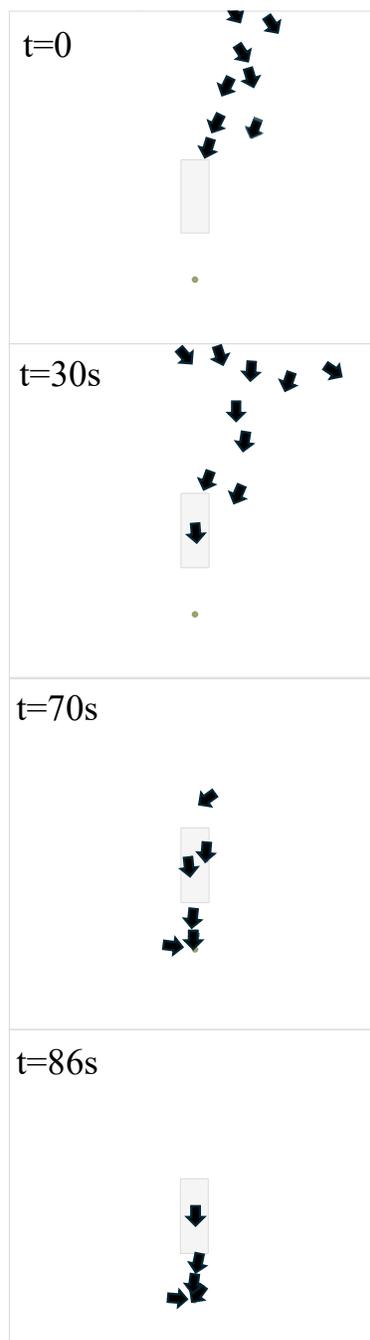
Success rate represents the percentage of aircraft that successfully pass through the corridors and reach their assigned goal within the length of the episode, i.e., 150 s. A high value of the success rate indicates better performance, as it implies that the aircraft flies efficient routes.

### 3. *Completion time per episode ( $T$ , s)*

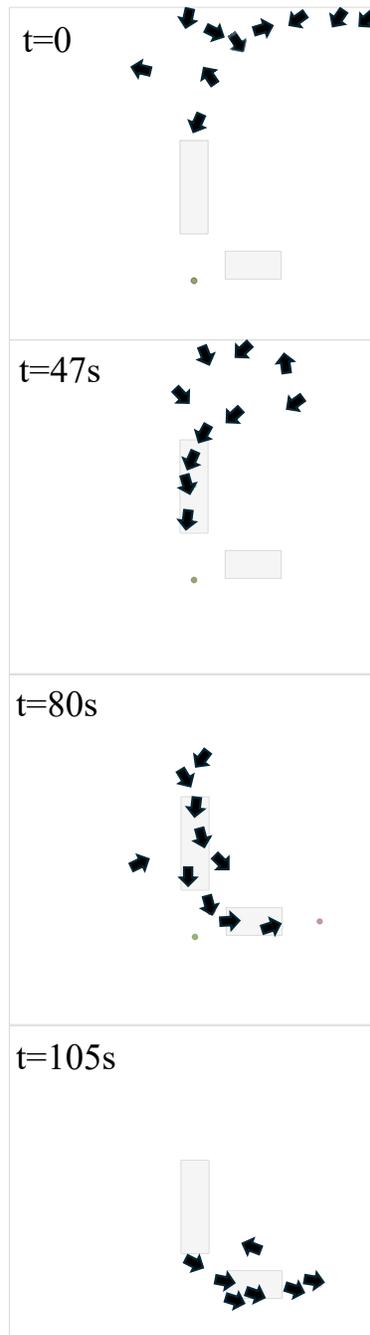
The time taken per episode is calculated based on how long it takes for all the aircraft in a scenario to reach their goals. A lower value of  $T$  is reflective of efficient navigation by all aircraft without unnecessary delays.

### 4. *Violation of separation minimum ( $\Delta d$ , m)*

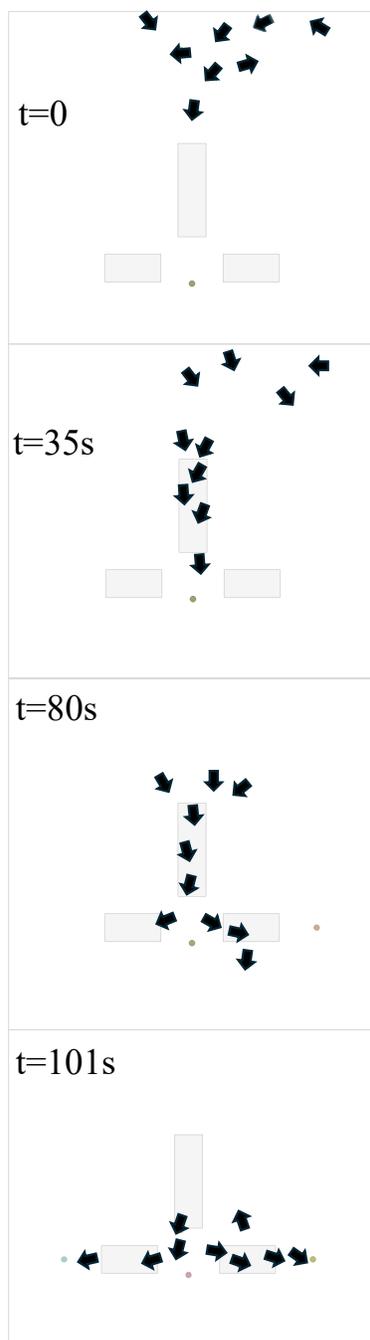
$\Delta d$  is calculated as the amount by which the inter-aircraft separation minimum, which is assumed to be 300 m, is violated. It is only considered when two aircraft come closer than the separation minimum; in these situations,  $\Delta d$  is given by the difference between the minimum and actual separations. For example, if two aircraft are 290 m apart, then  $\Delta d = 10$  m. This metric quantifies how well aircraft satisfy the separation minima while traveling through corridors. We report the mean and standard deviation of  $\Delta d$  in meters, noting that the values are only averaged over instances when a violation occurs. A lower value of  $\Delta d$  is preferred. We note that in all our experiments, the inter-aircraft separation never became zero, i.e., there were no collisions even without tactical deconfliction.



**Fig. 2 Visualization of behaviors of 10 aircraft in the simple metered corridor scenario. The aircraft start from the upper half of the environment and are tasked to navigate through the corridor and reach a point. The length of the vertical corridor is 1.6 km. The arrows indicate aircraft positions and are not to scale.**



**Fig. 3 Visualization of behaviors of 10 aircraft with the two consecutive corridors scenario. The aircraft start from the upper half of the environment and are tasked to navigate through both corridors. The length of the first vertical corridor is 2.0 km. The length of the horizontal corridor is 1.2 km. The arrows indicate aircraft positions and are not to scale.**



**Fig. 4** Visualization of behaviors of 10 aircraft in the split scenario where all aircraft enter the first corridor and then split into two corridors. The length of the first vertical corridor is 2.0 km. The length of each horizontal corridor is 1.2 km. The arrows indicate aircraft positions and are not to scale.

### 5. Need for tactical intervention for deconfliction ( $I\%$ )

In practice, a tactical intervention will be activated any time that two aircraft approach each other closer than the separation minimum. We therefore also calculate the need for such tactical intervention,  $I$ , as the average fraction of time when aircraft within corridors have a separation violation relative to the total time they spend in corridors. As in the case of  $\Delta d$ , a lower value of  $I$  is preferred.

Table 1 shows these metrics for the three corridor scenarios described in Fig. 1.

**Table 1 Performance metrics for the three corridor scenarios illustrated in Fig. 1. The metrics presented are the conformance to corridor boundaries ( $C\%$ , higher is better), the success rate ( $S\%$ , higher is better), the average completion time  $T$  per episode in seconds (lower is better), the mean and standard deviation of the separation distances when the minimum separation is violated ( $\Delta d$  m, lower is better), and the fraction of time that a tactical intervention for deconfliction is needed ( $I\%$ , lower is better). These performance metrics are defined in Sec. V C. We consider environments with 3, 5, and 10 aircraft, representing low-, medium-, and high-density environments.**

Scenario	# agents	Conformance, $C$ (%)	Success, $S$ (%)	Completion, $T$ (s)	$\Delta d$ (m)	Need for Tactical Deconfliction, $I$ (%)
Single	3	99%	99%	60.5	0±36	1%
	5	95%	94%	63.9	1±52	8%
	10	94%	92%	86.1	97±50	14%
Consecutive	3	98%	98%	96.8	0±41	1%
	5	95%	97%	99.9	8±56	6%
	10	94%	83%	102.5	97±42	17%
Split	3	95%	90%	87.8	0±48	2%
	5	95%	88%	91.5	31±53	7%
	10	96%	87%	109.9	41±127	12%

## D. Discussion

In all three scenarios, aircraft maintain high conformance to the corridor even when the total number of agents in the system is modified. The single corridor with the post-exit metering point achieves high success rates while also resulting in a lower violation of the separation minimum and the need for tactical intervention for deconfliction. Owing to the simple environment structure, it also takes the shortest total time. As we increase the number of aircraft from what was used in training to 5 and then 10 aircraft, as shown in Fig. 2, we observe a decrease in the corridor conformance and increased violations of the separation minimum.

As we increase the complexity of the scenario, such as the consecutive and split corridor experiments, we observe a lower success rate. Additionally, with an increased number of aircraft, the need for tactical intervention increases due to congestion in the environment. Fig. 3 shows a series of time steps of an episode involving the consecutive corridor scenario in a congested environment with ten aircraft. With the split corridor scenario (Fig. 4), the aircraft have multiple options and have to follow their recommended corridors, which further increases the complexity. This impacts the performance, as seen in the lower success rates across all numbers of agents.

## VI. Conclusions and future work

In this work, we presented a method that enables autonomous fixed-wing aircraft to learn to self-organize while traversing through AAM corridors in a decentralized manner. We modeled three scenarios involving aircraft traveling through corridors with increasing levels of complexity. The simplest scenario involved aircraft traversing through a single corridor with a post-exit metering point. We added consecutive corridors and split corridors and performed experiments with varying numbers of aircraft to represent levels of congestion. Our results showed that aircraft are able to conform to the corridor boundaries within the specified episode time. We also showed the requirements of tactical interventions when aircraft violate the desired spacing minima.

Future work includes extending the corridor-based navigation to coverage tasks after traversing the corridor and

considering a greater number of agents. Other interesting directions are generalizations to unseen environments and 3D vehicle dynamics with dynamic corridor generation.

## Acknowledgments

The authors would like to thank the MIT SuperCloud [36] and the Lincoln Laboratory Supercomputing Center for providing high-performance computing resources that have contributed to the research results reported within this paper. We also thank Ruhundaka Ejilemele and Aadarsh Govada for helpful discussions. J. Aloor was supported in part by a Mathworks Fellowship. This work was supported by NASA under grant #80NSSC23M0220, but this article solely reflects the opinions and conclusions of its authors and not any NASA entity.

## References

- [1] Federal Aviation Administration, “Urban Air Mobility (UAM) Concept of Operations Version 2.0,” Tech. rep., Federal Aviation Administration, Washington, DC, April 2023. 1, 2
- [2] SESAR Joint Undertaking, “U-space ConOps and architecture (edition 4),” Tech. rep., EUROCONTROL, July 2023.
- [3] Reim, G., “UAE Begins Mapping Air Corridors For Air Taxis, Cargo Drones,” *Aviation Week*, 2025. URL <http://aviationweek.com/aerospace/advanced-air-mobility/uae-begins-mapping-air-corridors-air-taxis-cargo-drones>.
- [4] World Economic Forum, “Skyways to the Future: Operational Concepts for Advanced Air Mobility in India,” , 2024. [http://reports.weforum.org/docs/WEF\\_Skyways\\_to\\_the\\_Future\\_2024.pdf](http://reports.weforum.org/docs/WEF_Skyways_to_the_Future_2024.pdf).
- [5] UNICEF, “Malawi’s Unique Drone Corridor,” , 2017. <https://www.unicef.org/innovation/drones/malawi-unique-drone-corridor>. 1
- [6] Federal Aviation Administration, “Advanced Air Mobility (AAM) Implementation Plan,” Tech. rep., Federal Aviation Administration, Washington, DC, July 2023. 1
- [7] Verma, S., Dulchinos, V., Wood, R. D., Farrahi, A., Mogford, R., Shyr, M., and Ghatas, R., “Design and Analysis of Corridors for UAM Operations,” *2022 IEEE/AIAA 41st Digital Avionics Systems Conference (DASC)*, 2022, pp. 1–10. <https://doi.org/10.1109/DASC55683.2022.9925820>. 1
- [8] Tony, L. A., Ratnoo, A., and Ghose, D., “Lane Geometry, Compliance Levels, and Adaptive Geo-fencing in CORRIDRONE Architecture for Urban Mobility,” *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2021, pp. 1611–1617. <https://doi.org/10.1109/ICUAS51884.2021.9476745>.
- [9] He, X., Li, L., Mo, Y., Sun, Z., and Qin, S. J., “Air Corridor Planning for Urban Drone Delivery: Complexity Analysis and Comparison via Multi-Commodity Network Flow and Graph Search,” *Transportation Research Part E: Logistics and Transportation Review*, Vol. 193, 2025, p. 103859. <https://doi.org/https://doi.org/10.1016/j.tre.2024.103859>, URL <https://www.sciencedirect.com/science/article/pii/S1366554524004502>. 1
- [10] Tomlin, C., Pappas, G., Lygeros, J., Godbole, D., Sastry, S., and Meyer, G., “Hybrid Control in Air Traffic Management Systems,” *IFAC Proceedings Volumes*, Vol. 29, No. 1, 1996, pp. 5512–5517. URL <https://www.sciencedirect.com/science/article/pii/S1474667017585596>. 2
- [11] Cruck, E., and Lygeros, J., “Subliminal air traffic control: Human friendly control of a multi-agent system,” *2007 American Control Conference*, 2007, pp. 462–467. <https://doi.org/10.1109/ACC.2007.4282641>. 2
- [12] Isufaj, R., Aranega Sebastia, D., and Angel Piera, M., “Toward conflict resolution with deep multi-agent reinforcement learning,” *Journal of Air Transportation*, Vol. 30, No. 3, 2022, pp. 71–80. 2
- [13] Brittain, M. W., and Wei, P., “One to any: Distributed conflict resolution with deep multi-agent reinforcement learning and long short-term memory,” *AIAA Scitech 2021 Forum*, 2021, p. 1952.
- [14] Ghosh, S., Laguna, S., Lim, S. H., Wynter, L., and Poonawala, H., “A deep ensemble method for multi-agent reinforcement learning: A case study on air traffic control,” *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 31, 2021, pp. 468–476. 2
- [15] Buelta, A., Olivares, A., and Staffetti, E., “Towards multi-aircraft transfer learning for trajectory tracking,” *Proceedings of the Fifteenth USA/Europe Air Traffic Management Research and Development Seminar*, 2023. 2

- [16] Spatharis, C., Kravaris, T., Vouros, G. A., Blekas, K., Chalkiadakis, G., Garcia, J. M. C., and Fernandez, E. C., “Multiagent Reinforcement Learning Methods to Resolve Demand Capacity Balance Problems,” *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*, Association for Computing Machinery, New York, NY, USA, 2018. <https://doi.org/10.1145/3200947.3201010>, URL <https://doi.org/10.1145/3200947.3201010>. 2
- [17] de Oliveira, Í. R., Neto, E. C. P., Matsumoto, T. T., and Yu, H., “Decentralized air traffic management for advanced air mobility,” *2021 Integrated Communications Navigation and Surveillance Conference (ICNS)*, IEEE, 2021, pp. 1–8. 2
- [18] Balakrishnan, H., and Chandran, B., “A Distributed Framework for Traffic Flow Management in the Presence of Unmanned Aircraft,” *USA/Europe ATM R&D Seminar*, 2017. 2
- [19] Henke, C., Frohleke, N., and Bocker, J., “Advanced convoy control strategy for autonomously driven railway vehicles,” *IEEE Intelligent Transportation Systems Conference*, 2006. 2
- [20] Heinovski, J., and Dressler, F., “Platoon formation: Optimized car to platoon assignment strategies and protocols,” *IEEE Vehicular Networking Conference (VNC)*, 2018. 2
- [21] Johansson, A., Nekouei, E., Johansson, K. H., and Mårtensson, J., “Strategic Hub-Based Platoon Coordination Under Uncertain Travel Times,” *IEEE Transactions on Intelligent Transportation Systems*, 2022. 2
- [22] Ishihara, A. K., Idris, H. R., and Xue, M., “A Framework for Sense and Follow Convoys for Collective Autonomous Mobility,” *AIAA Aviation Forum*, 2021. 2
- [23] Chottani, A., Hastings, G., Murnane, J., and Neuhaus, F., “Distraction or disruption? Autonomous trucks gain ground in US logistics,” [www.mckinsey.com/industries/travel-logistics-and-infrastructure/our-insights/distraction-or-disruption-autonomous-trucks-gain-ground-in-us-logistics](http://www.mckinsey.com/industries/travel-logistics-and-infrastructure/our-insights/distraction-or-disruption-autonomous-trucks-gain-ground-in-us-logistics), 2018. 2
- [24] Tsugawa, S., Jeschke, S., and Shladover, S. E., “A review of truck platooning projects for energy savings,” *IEEE Transactions on Intelligent Vehicles*, 2016. 2
- [25] Liu, Z., Wang, M., and Liu, Z., “Enhanced self-separation decision making for autonomous flight operations in air corridors,” *Journal of Air Transport Management*, Vol. 124, 2025, p. 102721. <https://doi.org/https://doi.org/10.1016/j.jairtraman.2024.102721>, URL <https://www.sciencedirect.com/science/article/pii/S0969699724001868>. 2
- [26] Chour, K., Razzaghi, P., Verma, D., Xue, M., Munishkin, A., and Kalyanam, K., “Analysis of Traffic Flow in Structured Urban Airspace Networks with MFD-based Feedback Control,” *AIAA AVIATION FORUM AND ASCEND 2024*, 2024, p. 4575. 2
- [27] Deniz, S., Wu, Y., Shi, Y., and Wang, Z., “A reinforcement learning approach to vehicle coordination for structured advanced air mobility,” *Green Energy and Intelligent Transportation*, Vol. 3, No. 2, 2024, p. 100157. <https://doi.org/https://doi.org/10.1016/j.geits.2024.100157>, URL <https://www.sciencedirect.com/science/article/pii/S2773153724000094>. 2
- [28] Doole, M., Ellerbroek, J., and Hoekstra, J. M., “Investigation of Merge Assist Policies to Improve Safety of Drone Traffic in a Constrained Urban Airspace,” *Aerospace*, Vol. 9, No. 3, 2022. <https://doi.org/10.3390/aerospace9030120>, URL <https://www.mdpi.com/2226-4310/9/3/120>. 2
- [29] Yu, L., Li, Z., Ansari, N., and Sun, X., “Hybrid Transformer Based Multi-Agent Reinforcement Learning for Multiple Unmanned Aerial Vehicle Coordination in Air Corridors,” *IEEE Transactions on Mobile Computing*, , No. 01, 2025, pp. 1–14. <https://doi.org/10.1109/TMC.2025.3532204>, URL <https://doi.ieeecomputersociety.org/10.1109/TMC.2025.3532204>. 2
- [30] Nayak, S., Choi, K., Ding, W., Dolan, S., Gopalakrishnan, K., and Balakrishnan, H., “Scalable Multi-Agent Reinforcement Learning through Intelligent Information Aggregation,” *Proceedings of the 40th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 202, PMLR, 2023, pp. 25817–25833. URL <https://proceedings.mlr.press/v202/nayak23a.html>. 2, 3
- [31] Aloor, J. J., Nayak, S. N., Dolan, S., and Balakrishnan, H., “Cooperation and Fairness in Multi-Agent Reinforcement Learning,” *ACM J. Auton. Transport. Syst.*, Vol. 2, No. 2, 2024. <https://doi.org/10.1145/3702012>, URL <https://doi.org/10.1145/3702012>. 2
- [32] Wisk, 2025. <https://wisk.aero/aircraft/>. 5
- [33] Joby Aviation, 2025. <https://www.jobyaviation.com/>.
- [34] Mogford, R., Peknik, D., and Zelman, J., “UAM Variable Separation,” Presentation, NASA Ames Research Center, 2020. <https://ntrs.nasa.gov/citations/20205006214>.

- [35] Choi, J. J., Aloor, J. J., Li, J., Mendoza, M. G., Balakrishnan, H., and Tomlin, C., “Resolving Conflicting Constraints in Multi-Agent Reinforcement Learning with Layered Safety,” *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, 2025. <https://doi.org/10.15607/RSS.2025.XXI.094>. 5
- [36] Reuther, A., Kepner, J., Byun, C., Samsi, S., Arcand, W., Bestor, D., Bergeron, B., Gadepally, V., Houle, M., Hubbell, M., Jones, M., Klein, A., Milechin, L., Mullen, J., Prout, A., Rosa, A., Yee, C., and Michaleas, P., “Interactive supercomputing on 40,000 cores for machine learning and data analysis,” *2018 IEEE High Performance extreme Computing Conference (HPEC)*, IEEE, 2018, pp. 1–6. 11