

Action Robust Reinforcement Learning for Air Mobility Deconfliction against Conflict Induced Spoofing

Deepak Kumar Panda¹ and Weisi Guo¹

Abstract—Increased dynamic drone usage has increased complexity in aerial navigation and often demands distributed local deconfliction. Due to the high velocities and few landmarks, robust deconfliction relies on precise positioning and synchronization. However, intentional spoofing attacks aimed at inducing navigation conflicts threaten the reliability of conventional techniques. Here, we address these concerns by establishing a baseline on the impact of novel conflict-inducing spoofing attacks on existing geometric navigation methods. Based on the impact of the attacks on the navigation, reinforcement learning (RL) strategy is used to counter the effects of spoofing attacks. In order to counter the effect of spoofing in randomized dynamic airspace conditions, a zero-sum action-robust (ZSAR) RL based on mixed Nash equilibrium objective is used. The proposed methodology yields an improved number of conflict-free paths while reducing average conflicts compared to existing state of the art RL strategies, thus making it suitable for deploying autonomous aircrafts.

Index Terms—Spoofing, conflict resolution, reinforcement learning, UAV, adversarial network, deconfliction.

I. INTRODUCTION

ADVANCED air mobility (AAM) involves autonomous transportation of individuals and goods while incorporating innovative aircraft designs into existing and modified airspace [1]. Consequently, there is an anticipated surge in deploying unmanned aerial vehicles (UAVs) for diverse applications in co-existence with existing passenger aircrafts. Consequently, significant diversity exists in the deconfliction speed, manoeuvrability, priorities, and spatial separation requirements. In contrast to ground autonomous vehicles, airborne platforms frequently encounter challenges in utilizing visual or lidar technologies because of the greater distances involved, elevated speeds, and susceptibility to weather conditions.

Achieving complete aerial mission autonomy requires accurate positioning, which has its security concerns. For example, EUROCONTROL has outlined existing issues in the utilization of open and legacy wireless and positioning protocols and the potential cascaded impact of cyber attacks [2]. The European Aviation Safety Agency (EASA) [3] underscores the potential role of artificial intelligence (AI) in addressing these

challenges by patching system vulnerabilities and identifying preemptive threats based on behavioral patterns.

Cyber threats targeting UAVs are multifaceted, encompassing data interception [4], denial-of-service [5], jamming [6], and spoofing/manipulation (both evasion [7] and gamification [8]). Countermeasures to tackle cybersecurity threats can be broadly classified into prevention, detection, and mitigation strategies [9]. Notably, defence against intentional jamming and spoofing attacks requires urgent and practical solutions, especially when acted against the global positioning system (GPS) signals, even with known predefined trajectories and the inclusion of secondary and tertiary sensor fusion [10].

The UAV functions predominantly within the manual to full autonomous flight modes, and to alleviate the workload of air traffic controllers (ATC), the current research emphasizes the decentralized navigation mode, which requires full mission deconfliction autonomy, which depends on accurate GPS signals. Interference attacks on GPS manifests in two main categories: spoofing and jamming. Notably, the threat posed by GPS spoofing is more pronounced than jamming, as it can generate inaccurate navigation signals or even seize complete control of the trajectory of GPS-driven autonomous UAVs. However, very few works have mitigated the effects of GPS spoofing on autonomous conflict resolution (CR) strategies in dynamic air traffic management.

A. Related Works

1) Autonomous Deconfliction Algorithms

The CR strategies within unmanned airspace can be broadly categorized into centralized [11] and decentralized approaches [12, 13]. This study focuses on decentralized navigation, which can be further subdivided into three main categories: prescribed, negotiated, and reactive methods. Prescribed methods involve adhering to predefined rules for executing manoeuvres [14], although interpretable and adhering to regulations, this approach may lead to increased conflicts in dense and dynamic air traffic scenarios. In negotiated methods, CR relies on explicit communication between aircrafts to reach a mutual consensus [15], aiming for a globally optimal strategy; however, there is a risk of deadlock, rendering it impractical in dynamic airspace or for aircrafts with no hover capability.

Reactive methods consider CR manoeuvres based on the potential conflicts based on interpolated trajectories of involved aircrafts. Furthermore, they can be expanded to address multi-actor conflicts and adapt to changing airspace dynamics [16]. These conflict geometry-based manoeuvres can be classified

This work was supported by the Royal Academy of Engineering and the Office of the Chief Science Adviser for National Security under the UK Intelligence Community Postdoctoral Research Fellowship programme

¹The authors are with School of Aerospace, Transport, and Manufacturing, Cranfield University, MK43 0AL Cranfield, U.K. Deepak.Panda@cranfield.ac.uk, weisi.guo@cranfield.ac.uk

into potential-based [17] and solution space methods [18]. The former relies on simple conflict geometry, executing manoeuvres based on the closest point of approach (CPA) at some look-ahead time. At the same time, the latter seeks to find conflict-free velocity vectors by optimizing specific objectives like fuel consumption or deviation from desired velocity. The latter set of methods faces a risk of being infeasible under certain geometric constraints, reducing its reliability in diverse uncooperative air traffic scenarios. However, these methods typically address conflicts at specific time steps without considering the cumulative impact of sequential decision-making on overall mission success, highlighting the potential utility of model-free methods.

2) Deep Reinforcement Learning for Autonomous Deconfliction

Deep reinforcement learning (DRL) is a prevalent technique utilized for CR [19], with algorithms taking decisions in discrete spaces such as deep Q-networks (DQN) and double deep Q-networks (DDQN) [20], as well as continuous spaces like proximal policy optimization (PPO) [21], deep deterministic policy gradient (DDPG) [4, 22], and multi-agent deep deterministic policy gradient (MADDPG) [23]. Air traffic state representation has been achieved using long short-term memory (LSTM) [24] and attention networks [21]. However, challenges persist in deploying DRL-based navigation [25], such as issues related to validation and generalizability across diverse traffic scenarios/tasks, thereby limiting the trustworthiness of these models for real-world applications. Hybrid approaches combining autonomous reactive CR strategies with DRL have been explored in [26] to determine the adaptive look-ahead time to execute the CR manoeuvre. However, none have definitively resolved navigation concerns in dense airspace under adversarial conditions. Adversarial AI methods have shown effectiveness in developing robust algorithms capable of adapting quickly to changing environments. Unlike traditional AI approaches, which are vulnerable to perturbations in input data [27], adversarial AI holds promise for ensuring safe and resistant navigation in dynamic air traffic scenarios. These methods are resilient against input manipulations, such as spoofing attacks that introduce false navigation data and air traffic information, making them valuable for enhancing security in evolving airspace contexts. Adversarial AI has been utilized in both single-agent [28, 29] and multi-agent [30] UAVs for deep learning (DL)-based navigation and control in the face of adversarial challenges. These attacks often come in two forms: black-box attacks, where attackers have limited model knowledge and manipulate pixel inputs based on gradients of loss functions, and white-box attacks, where adversaries inject attacks based on specific parameter of UAV navigation [7]. Defensive strategies against these attacks include creating robust adversarial samples to strengthen navigation resilience [29] and using SHAP values for spoof attack detection [28]. However, the application of adversarial methods to counter spoofing attacks in rapidly evolving air traffic environments remains an area yet to be thoroughly explored, which is the focus of this paper.

B. Gaps & Contribution in this paper

While existing literature has predominantly focused on effective DRL-based CR strategies against dynamic air traffic, limited focus has been given to exploring targeted attack strategies on UAV navigation [7, 29]. Based on the previous DRL algorithms for CR, the efficiency diminishes with the spoofed input signals to instigate conflicts with nearby aircraft. In addressing this challenge, geometric methods around obstacles have been integrated with DDPG for attack detection, as evidenced in [28]. However, the impact of such attacks and the subsequent correction in navigation strategies have not been discussed. A notable gap exists in the previous works while investigating the cascaded impact due to targeted attacks within dynamic air traffic scenarios, and the impact should aid in developing a robust defence strategy against spoofing attacks within dynamic air traffic scenarios. Despite the notable achievements of RL in various practical domains such as games [31] and robotics [32], its results are vulnerable when transitioning from training to real-world environment, particularly when encountering differences between the training and testing scenarios. This paper introduces an enhancement to CR by incorporating geometric methods with action-robust Deep Deterministic Policy Gradient (DDPG). The goal is to optimize navigational efficiency while ensuring resilience against dynamic conflict-induced spoofing amid dynamic air traffic. The design of the algorithm is guided by reducing the maximum impact of conflict-induced spoofing. The integration of geometric navigation methods with action-robust reinforcement learning (RL) enhances navigation by removing the reliance on sparse rewards, allowing a focus on mission completion and defence against spoofing attacks. This approach contrasts with the method in [33], where both strategies are managed simultaneously, reducing the generalizability of the algorithm. Adversarial training, as discussed in [34–36], is limited in its robustness to specific types of white-box or black-box attacks and fails to generalize across a broader range of unpredictable spoofing attack patterns. Therefore, action-robust RL improves the algorithm's adaptability to a wider and diverse set of environment-based attacks.

In this study, we introduce a RL-based strategy aiding maximum voltage potential (MVP) for UAV navigation to effectively mitigate the impact of conflict-induced spoofing attacks. To address spoofing in randomized and dynamic airspace scenarios, we propose an action-robust DDPG approach formulated as a zero-sum game between the agent and an adversary, aiming to solve a mixed Nash equilibrium objective. This approach is analogous to enhancing intrusion detection in cyber-physical systems using generative adversarial networks (GAN) and non-zero-sum game between the actor and the adversary to formulate non-zero-sum action robust (NZSAR) RL, as described in [37] and [38] respectively. The analysis encompasses conflict-free paths, the number of conflicts, and overall steps to complete the mission. This study employs a distributional analysis similar to our prior work in [39], considering the stochastic nature of attacks, missions, and the number of aircraft. The results are compared with the geometric MVP CR method and state-of-the-art (SOTA)

geometry-augmented DDPG used in scenarios involving static [28] and dynamic obstacles [26], and NZSAR RL formulated in [38]. The contributions can be summarized as:

- **Development of novel spoofing attack strategy:** Utilize the novel randomized conflict-induced spoofing attack to analyze the effect on UAV navigation in dynamic air traffic.
- **Design of zero-sum action robust RL against spoofing attack:** Development of a defensive strategy with MVP augmented zero-sum action robust RL against conflict-induced spoofing attack for a given UAV in dense airspace of non-cooperative centralized aircraft.
- **Comparison of the metrics with the state-of-the-art:** RL Validation and analysis of the ZSAR RL against SOTA RL algorithms like DDPG and NZSAR comparing the metrics like the number of conflict-free paths, conflicts and steps required for mission completion.

II. CONFLICT RESOLUTION UNDER INDUCED SPOOFING ATTACK

In this section, we describe the MVP method used for conflict resolution, where an ownship UAV is tasked with the mission to move from the start point $(x_{\text{start}}, y_{\text{start}})$ to the goal point $(x_{\text{goal}}, y_{\text{goal}})$ in the 2D airspace, which is shared with other aircraft under centralized control, each pursuing its designated mission.

Due to their centralized navigation, these aircraft are anticipated to follow direct trajectories facilitated by cooperative data exchange through, automatic dependent surveillance–broadcast (ADS-B) while following a non-cooperative CR strategy [39]. The airspace operations used in this study as per the taxonomy defined in [11, 12]:

- **Surveillance:** Independent surveillance techniques are used, with each aircraft expected to cooperate through ADS-B. Unmanned airspace manager (UAM) can monitor the ADS-B signals exchanged between aircraft.
- **Trajectory:** The centralized aircraft are assumed to be moving in a straight line.
- **Coordination:** ADS-B data (i.e. position and velocity) communicated amongst aircraft and UAV.
- **Manoeuvre:** Reactive method as ownship UAV has to perform the CR manoeuvre, with centralized aircraft moving across a pre-programmed path.
- **Avoidance Planning:** UAV has to detect the aircraft within certain radius with finite probability of conflict, to plan escape avoidance.
- **Avoidance manoeuvre:** Speed and Heading.

The conflict detection strategy using MVP is explained in the next subsection.

A. Modified Voltage Potential

Modified voltage potential (MVP) is a potential field methodology to mitigate conflicts arising from dynamic obstacles. Within this algorithmic framework, aircraft are visualized as charged particles, with their CR manoeuvre effectively serving as repulsive forces, deflecting particles away from one another based on their CPA [17]. MVP stands out for its inherent

simplicity relative to alternative CR strategies, which accounts for the potential constraints using geometric representations.

We have considered the motion of the ownship UAV at position $\mathbf{P}_{\text{own}} = (x_{\text{own}}, y_{\text{own}})$ moving in the airspace at speed magnitude v_{own} and heading angle ϕ_{head} as a point mass model as described in (1) [39].

$$\begin{aligned}\dot{x}_{\text{own}} &= v_{\text{own}} \cos \phi_{\text{head}}, \\ \dot{y}_{\text{own}} &= v_{\text{own}} \sin \phi_{\text{head}}.\end{aligned}\quad (1)$$

As we see in Figure 1a, in an unspoofed system, ownship UAV is potentially in conflict with aircraft 1, in some point of time, while there is no potential conflict with aircraft 2. Hence, CR manoeuvre with MVP will be executed with respect to aircraft 1. Hence, we can consider the position of ownship at time t_0 as $\mathbf{P}_{\text{own}}(t_0)$ and the position of the nearest aircraft as $\mathbf{P}_{\text{int}}(t_0)$, and the velocities are defined as $\mathbf{V}_{\text{own}}$ and $\mathbf{V}_{\text{int}}$ respectively, considering its magnitude and heading angle. Let us consider the relative distance between the ownship and the intruder aircraft at time t as $\mathbf{P}_{\text{rel}}(t)$, defined as:

$$\begin{aligned}\mathbf{P}_{\text{rel}}(t) &= \mathbf{P}_{\text{int}}(t_0) + (t - t_0) * \mathbf{V}_{\text{int}} - \mathbf{P}_{\text{own}}(t_0) - (t - t_0) * \mathbf{V}_{\text{own}}, \\ r_{\text{rel}} &= \sqrt{\mathbf{P}_{\text{rel}}^T \cdot \mathbf{P}_{\text{rel}}}.\end{aligned}\quad (2)$$

In order to find the time at CPA, we need to find $\min_t \mathbf{P}_{\text{rel}}(t)$ to obtain t^* which is obtained from (2) as:

$$t^* = t_0 - \frac{(\mathbf{V}_{\text{int}} - \mathbf{V}_{\text{own}})^T \cdot (\mathbf{P}_{\text{int}}(t_0) - \mathbf{P}_{\text{own}}(t_0))}{\|\mathbf{V}_{\text{int}} - \mathbf{V}_{\text{own}}\|^2}. \quad (3)$$

We can refer to (3) intuitively in the frame of the nearby intruding aircraft, where its velocity is subtracted from the ownship while considering its position to be constant. We consider the distance d_{min} as the conflict radius of the aircraft as shown in Figure 1. If we consider $\Delta \mathbf{P} = \mathbf{P}_{\text{int}}(t_0) - \mathbf{P}_{\text{own}}(t_0)$ and $\Delta \mathbf{V} = \mathbf{V}_{\text{int}} - \mathbf{V}_{\text{own}}$, and accordingly substitute the expression (3) in (2), then we can write the expression for the minimum distance between the intruder and the ownship as,

$$p_{\text{min}} = \left\| \Delta \mathbf{P} + \Delta \mathbf{V} \cdot \left(t_0 - \frac{\Delta \mathbf{V} \cdot (\Delta \mathbf{P} + \mathbf{V}_{\text{A}} \cdot t)}{\|\Delta \mathbf{V}\|^2} + \mathbf{V}_{\text{A}} \cdot t \right) \right\|. \quad (4)$$

The MVP strategy is to find the closest point of approach, i.e. $\mathbf{O}$ at Figure 1, and from this point finding the closest distance out from the conflict zone, as defined in [17, 40]:

$$\mathbf{OO}' = \kappa * d_{\text{min}} \left| \cos \left(\arcsin \left(\frac{d_{\text{min}}}{\Delta \mathbf{P}} \right) - \arcsin \left(\frac{p_{\text{min}}}{d_{\text{min}}} \right) \right) \right|. \quad (5)$$

Here κ refers to the protection radius amount which will steer the UAV from the conflict with the neighbourhood aircraft. In this study κ is considered to be 1.1. Hence, considering (5), we can find the corrected velocity manoeuvre $\Delta \mathbf{V}_{\text{cor}}$ as shown in Figure 2,

$$\mathbf{V}_{\text{own}}^{\text{uncon}} = \Delta \mathbf{V}_{\text{cor}} + \mathbf{V}_{\text{own}} = \frac{\mathbf{OO}'}{t^*} + \mathbf{V}_{\text{own}}. \quad (6)$$

where, $\mathbf{V}_{\text{own}}^{\text{uncon}}$ deconflicted velocity of the ownship UAV after the manoeuvre $\Delta \mathbf{V}_{\text{cor}}$ has taken place. In the case of multiple aircraft, the net velocity manoeuvre is the summation of the manoeuvres considered for all the neighbourhood aircrafts.

given airspace along with the ownship UAV, with the position $\mathbf{P}_{\text{int}} = \{x_{\text{int}_i}, y_{\text{int}_i}\}_{i \in N}$. We define the position of the nearest neighbor as $\mathbf{P}_{\text{nn}} = (x_{\text{nn}}, y_{\text{nn}})$, defined by,

$$P_{\text{nn}} = \underset{(x_{\text{nn}}, y_{\text{nn}})}{\operatorname{argmin}} \left\| (x_{\text{own}}, y_{\text{own}}) - (x_{\text{int}_i}, y_{\text{int}_i}) \right\|_2^2 \quad (10)$$

Utilizing the position of the nearest aircraft, the spoofer with the spoofing radius r_{sp} injects the spoofing signal as shown in Algorithm 1. at a probability p_{att} .

C. Practical Defence Against Conflict Induced Spoofing

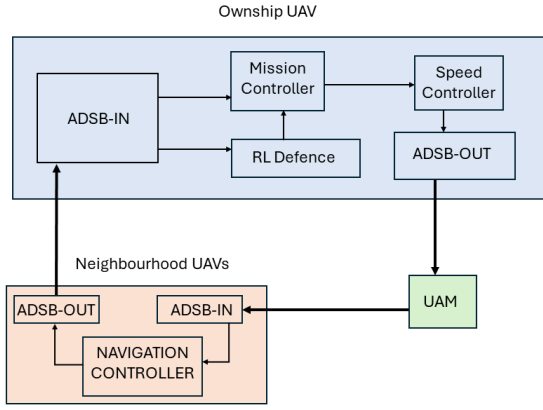


Fig. 3. Schematic of the RL used as a secondary system for aiding UAV navigation while communication data from neighbourhood UAVs and UAM via ADS-B.

Although practical anti-spoofing techniques exist to detect such spoofing attacks [9], they often fail to provide corrective measures necessary for mission completion. Therefore, a defence system is employed alongside the primary navigation strategy to ensure redundancy and validation, enabling the UAV to successfully complete its mission [41]. Inertial measurement unit (IMU) sensors can serve as a backup to GPS, offering an alternative when signals are jammed or spoofed. However, IMUs operate independently of external inputs and are prone to drift issues [42], making their navigation output unreliable. To address these challenges, RL is integrated as a secondary system to support the navigation controller, as depicted in Figure 3, ensuring the UAV can complete its mission even under spoofing attacks. As shown in Figure 3, even though its not mandatory for the UAVs to be equipped with ADS-B IN capabilities, however as specified in [43], ADS-B IN improves the safety operations of the UAV considerably. The strategy is implemented through the following steps:

- **Communication between Ownship UAV and UAM:** The ownship UAV's ADS-B OUT transmits its position signal to the centralized UAM, as illustrated in Figure 2.
- **Communication between UAM and Neighboring UAVs:** The centralized UAM requests neighboring UAVs within a certain radius of the ownship UAV, via their ADS-B IN, to send their navigational information to the ownship UAV.
- **Communication between Neighboring UAVs and Ownship UAV:** The neighboring UAVs transmit their position and velocity data through their ADS-B OUT, which is received by the ownship UAV's ADS-B IN.

- **Execution of the Defence Strategy using RL:** The mission controller implements the MVP algorithm to avoid conflicts with neighboring UAVs. Additionally, the RL algorithm uses information from the ADS-B IN to dynamically adjust the UAV's speed, in conjunction with the speed recommended by the MVP algorithm. The computed speed reference is then provided to the speed controller, which dynamically adjusts the velocity, ensuring the mission is completed even in the presence of spoofing attacks.

III. ACTION ROBUST RL AGAINST CONFLICT INDUCED SPOOFING

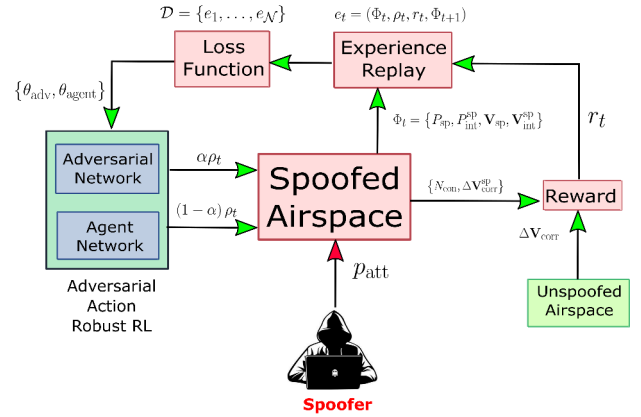


Fig. 4. Schematic of the training of adversarial action robust RL against conflict induced spoofing attack

As shown in Figure 4, action robust RL is used as a secondary system alongside MVP to defend against the random conflict-induced spoofing attack explained in the previous subsection, as per the cyber-secure architecture against spoofing defined in [41]. Action robust RL has two component networks, i.e. adversarial and agent, which contribute to the action space, with α being the proportion of the action contributed by the adversary. Both networks are parameterized as (ω, θ) for adversary and agent, respectively. DDPG is used as the action space of the RL is continuous. The states are represented by Φ , based on which the action ρ is taken, thus obtaining the reward r and the spoofed airspace state transition to Φ' . The deep neural network representing the agent and the adversary are simultaneously trained utilizing the state transition tuples $e_n = \{\Phi, \rho, r, \Phi'\}$ from the experience replay buffer consisting of $\mathcal{N}$ experiences as $\mathcal{D} = \{e_1, \dots, e_{\mathcal{N}}\}$. The loss function is computed by sampling batches of experience tuples from the replay buffer $\mathcal{D}$ to obtain the neural network parameters (θ, ω) for the adversary and the agent, respectively. The training is conducted considering adversarial samples generated by spoofing the UAV position at the probability of p_{att} . The reward r obtained from the CR manoeuvre from the unspoofed airspace and the corresponding conflicts and collisions occurring in the spoofed airspace. In the testing phase, the trained action robust DDPG will compensate for the damage in the spoofed airspace by controlling the speed magnitude and heading accordingly.

A. States

The state space Φ consists of information which includes the ownship and the nearest aircrafts from the spoofed position given as,

$$\Phi = \{ \{ \Delta \mathbf{P}_i^{\text{sp}}, \Delta \mathbf{V}_i^{\text{sp}} : i \in N(o) \}, \Delta \mathbf{P}_{\text{rel}}^{\text{sp}}, \mathbf{V}_{\text{own}}^{\text{sp}} \} \quad (11)$$

where,

$$\begin{aligned} \Delta \mathbf{P}_i^{\text{sp}} &= \mathbf{P}_{\text{int}}^{\text{sp}} - \mathbf{P}_{\text{sp}}, \\ \Delta \mathbf{V}_i^{\text{sp}} &= \mathbf{V}_{\text{int}}^{\text{sp}} - \mathbf{V}_{\text{sp}}, \\ \Delta \mathbf{P}_{\text{rel}}^{\text{sp}} &= \mathbf{P}_{\text{sp}} - \mathbf{P}_{\text{goal}}. \end{aligned} \quad (12)$$

Here $N(o)$ represents set of 2 free flight UAVs nearest to the UAVs spoofed position $\mathbf{P}_{\text{sp}}$.

B. Action

The action ρ represents the velocity magnitude regulation η of the UAV along with the respective heading correction ϕ_{corr} to avoid conflicts under spoofed conditions.

C. Reward

The episodic reward R_{ep} is the penalty imposed on the UAV which consists of three parts i.e. the summation of the UAV steps across the difference between the velocity correction manoeuvre in spoofed and unspoofed conditions i.e. $-\sum_{t=1}^{N_{\text{steps}}} |\Delta \mathbf{V}_{\text{cor}} - \Delta \mathbf{V}_{\text{cor}}^{\text{sp}}|$. The summation provides a step penalty for the total number of steps N_{steps} utilized by the UAV. At the same time, the absolute difference tries to compensate for the change in velocity manoeuvre due to a spoofing attack. The second component penalizes the total number of conflicts during the entire episode N_{con} with each conflict contributing to the penalty of r_{col} . The third component is the heavy penalty r_{col} imposed on the agent if it collides with the neighbouring aircraft. The three penalties are weighted respectively using $\{w_1, w_2, w_3\}$. Hence, the total episodic reward can be written as:

$$R_{\text{ep}} = -w_1 \cdot \sum_{t=1}^{N_{\text{steps}}} |\Delta \mathbf{V}_{\text{cor}} - \Delta \mathbf{V}_{\text{cor}}^{\text{sp}}| - w_2 \cdot N_{\text{con}} \cdot r_{\text{con}} - w_3 \cdot r_{\text{col}}. \quad (13)$$

The algorithm to maximize the robust reward against the random number of aircraft and randomized spoofing attack is explained in the next subsection.

D. Zero Sum Action Robust Reinforcement Learning (ZSAR RL)

In order to tackle the rapidly changing environmental dynamics in the context of RL, adversarial perturbation is employed where a two-player max-min problem emerges involving the agent and the adversary. The agent strives to fulfil given missions while concurrently remaining robust against the descriptions generated by the adversary.

Two methods for modelling these adversaries are robust adversarial reinforcement learning (RARL) [44] and noisy robust Markov decision process (NR-MDP) [45]. However, training robust RL objectives poses a challenge, as they prove sub-optimal when employing deterministic policy gradient steps to address the max-min objectives outlined in [45]. Moreover, the inherent difficulty stems from the non-convex-concave nature of the objectives in robust RL, where pure Nash

equilibrium (NE) may not be guaranteed for such problems. However, this challenge can be encountered by adopting a sampling perspective from the mixed NE, which provides an approximate and accurate solution to the max-min optimization problem [46]. This approach can be harnessed to develop robust reinforcement learning algorithms [47].

Let us consider the two-player (i.e. agent and adversary) zero-sum Markov game [48], where at each time instant both the players choose an action simultaneously, and the reward depends on the joint action space of both the players. This can be described in the form of MDP as $\mathcal{M} = (\Phi, \mathcal{A}_{\text{agent}}, \mathcal{A}_{\text{adv}}, T_2, \gamma, R_2, P_0)$, where $\mathcal{A}_{\text{agent}}$ and $\mathcal{A}_{\text{adv}}$ represent the continuous set of action space for agent and adversary, respectively, $T_2 : \Phi \times \mathcal{A}_{\text{agent}} \times \mathcal{A}_{\text{adv}} \times \Phi \rightarrow \mathbb{R}$ represents the state transition probability and $R_2 : \Phi \times \mathcal{A}_{\text{agent}} \times \mathcal{A}_{\text{adv}}$ is the reward for both the players. We assume that the agent executes the policy $\pi : \Phi \rightarrow \mathcal{A}_{\text{agent}}$ and adversary executing the policy $\nu : \Phi \rightarrow \mathcal{A}_{\text{adv}}$ in the environment $\mathcal{M}$. Thus we can say that for each time step t , both agents and adversary observe state Φ_t and take actions ρ_{agent} and ρ_{adv} respectively, getting the reward r and $-r$ respectively. Using the two-player Markov game, the following objective can be designed [44, 45].

$$J(\pi, \nu) = \mathbb{E} \left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \right]. \quad (14)$$

where, $\sum_{t=1}^{\infty} \gamma^{t-1} r_t$ is the random cumulative return/. If we consider the parameterized policies for the agent and adversary as $\{\pi_{\theta} : \theta \in \Theta\}$ and $\{\nu_{\omega} : \omega \in \Omega\}$ respectively, hence the objective in (14) is written as,

$$\max_{\theta \in \Theta} \min_{\omega \in \Omega} J(\theta, \omega). \quad (15)$$

Here J is non-convex-concave in both θ_{agent} and θ_{adv} . If we have a mixed strategy formulation of (14), while considering all the probability distribution over Θ and Ω , hence optimal distribution is to be obtained that solves the following objective

$$\max_{p \in \mathcal{P}(\Theta)} \min_{q \in \mathcal{P}(\Omega)} f(p, q) := \mathbb{E}_{\theta \sim p} [\mathbb{E}_{\omega \sim q} [J(\theta, \omega)]] \quad (16)$$

The objective in (16) intuitively means that the agent has to maximize the reward while considering the worst-case impact from the adversary. As shown in Figure 4, in order to find the loss function from the experience replay, the targets ϑ have to be computed, from which the loss function is computed. It is given as:

$$\vartheta(r, \Phi', d) = r + \gamma(1-d) Q_{\phi_{\text{targ}}}(\Phi', \rho_{\text{target}}). \quad (17)$$

where $\rho_{\text{target}} = (1-\alpha)\mu_{\theta_{\text{target}}}(\Phi') + \alpha\nu_{\omega_{\text{target}}}(\Phi')$. r represents the reward from the experience replay samples, γ represents the discount factor, d represent represents the terminal status of the given experience sample. $Q(\cdot)$ represents the value function at transitioned state Φ' . $\mu_{\theta_{\text{target}}}$ and $\nu_{\omega_{\text{target}}}$ represent the policies for the agent and adversary respectively having parameters $\{\theta_{\text{target}}, \omega_{\text{target}}\}$ respectively. Now the critic is updated by one step of preconditioned descent $\nabla_{\phi} L(\phi)$ as:

$$\nabla_{\phi} L(\phi) = \frac{1}{N} \sum_{(\Phi, \rho, r, \Phi', d) \in B} (\vartheta(r, \Phi', d) - Q_{\phi}(\Phi, \bar{\rho}))^2. \quad (18)$$

Then the agent and adversary policy network has been updated by minimizing the following loss functions:

$$\nabla_{\theta} J(\widehat{\theta}, \omega_t) = \frac{1-\alpha}{N} \sum_{\Phi \in \mathcal{D}} \nabla_{\theta} \mu_{\theta}(\Phi) \nabla_{\bar{\rho}} Q_{\phi}(\Phi, \bar{\rho}). \quad (19)$$

$$\nabla_{\omega} J(\widehat{\theta}_t, \omega) = \frac{\alpha}{N} \sum_{\Phi \in \mathcal{D}} \nabla_{\omega} v_{\omega}(\Phi) \nabla_{\bar{\rho}} Q_{\phi}(\Phi, \bar{\rho}). \quad (20)$$

where, $\bar{\rho} = \alpha \cdot \rho_{\text{adv}} + (1-\alpha) \cdot \rho_{\text{agent}}$. In [47] it has been shown that, solving mixed strategy objective (16) as a sampling problem has superior performance over solving the objective in pure strategy form given in (15). In order to improve the generalizability within the DNN, stochastic gradient Langevin dynamics (SGLD) has been used, which improves the convergence rate using the positive-definite symmetric matrix C . As shown, in [49], adaptive preconditioning of the covariance matrix C using the approximate second order moment of gradient update can improve the generalizability of SGD and the first order methods used in [49]. With the abuse of notation, we can consider the gradient g to optimize the reward function loss function concerning the parameter θ and ω ,

$$g_{\theta} \leftarrow \nabla_{\theta} J((\theta, \omega)), g_{\omega} \leftarrow \nabla_{\omega} J((\theta, \omega)). \quad (21)$$

Considering the following the parameters,

$$\begin{aligned} \xi_t &\sim \mathcal{N}(\mu_t, C_t), \\ \mu_t &\leftarrow \rho \mu_{t-1} + (1-\rho) g(\Theta_t), \\ C_t &\leftarrow \rho C_{t-1} + (1-\rho) (g(\Theta_t) - \mu_t)(g(\Theta_t) - \mu_{t-1}). \end{aligned} \quad (22)$$

Here $g = \{g_{\theta}, g_{\omega}\}$, $\mu_t = \{\mu_{\theta}, \mu_{\omega}\}$, $C_t = \{C_{\theta}, C_{\omega}\}$. The update equation for the agent and the adversary parameter to obtain the mixed Nash equilibrium for the max-min objective for the sample k from the buffer at time t , is given as:

$$\theta_t^{k+1} \leftarrow \theta_t^k + \eta \left(\nabla_{\theta} J(\widehat{\theta}, \omega_t) + \psi \xi_t \right), \quad (24)$$

$$\omega_t^{k+1} \leftarrow \omega_t^k - \eta \left(\nabla_{\omega} J(\widehat{\theta}, \omega_t) (\Gamma_t) + \psi \xi_t \right). \quad (25)$$

Here ψ represents the noise parameter. The parameters updated with the time t is given by

$$\bar{\omega}_t = (1-\beta) \bar{\omega}_t + \beta \omega_t^{(k+1)}; \bar{\theta}_t = (1-\beta) \bar{\theta}_t + \beta \theta_t^{(k+1)}. \quad (26)$$

Obtain the parameters for the next time duration as,

$$\omega_{t+1} = (1-\beta) \omega_t + \beta \bar{\omega}_t; \theta_{t+1} = (1-\beta) \theta_t + \beta \bar{\theta}_t. \quad (27)$$

We can write the whole action robust DDPG with mixed NE with Langevin dynamics (LD) as given as Algorithm 2.

E. State of the Art Methods

Previous studies have employed various RL techniques as secondary systems to enhance operational security. In [28], DDPG is combined with artificial potential field (APF) methods to improve navigation while detecting adversarial attacks. In [38], intrusion detection is enhanced using DRL, where the strategy is designed considering a non-zero-sum game between the external adversary and the agent. The SOTA RL methods can be summarized as follows:

Algorithm 2 Zero Sum Action Robust RL

Initialize: Policy parameters ω , θ and Q function parameter χ .
Initialize: The target network parameters $\omega_{\text{targ}} \leftarrow \omega_1$, $\theta_{\text{targ}} \leftarrow \theta_1$, and $\chi_{\text{targ}} \leftarrow \chi$.

Initialize: Replay buffer $\mathcal{D}$

Initialize: $t \leftarrow 1$.

repeat

- 1: Observe state Φ , select the actions as per $\rho_{\text{agent}} = \mu_{\theta} + \zeta$, $\rho_{\text{adv}} = \mu_{\omega} + \zeta'$ where $\zeta, \zeta' \sim \mathcal{N}(0, \sigma I)$.
- 2: Execute the action $\rho_t = \alpha \cdot \rho_{\text{adv}} + (1-\alpha) \cdot \rho_{\text{agent}}$ in the environment to correct the CR manoeuvre.
- 3: Observe the reward r in (13), obtain the next state Φ' after velocity and heading correction from (1), and check whether the UAV has arrived its destination using the done signal d .
- 4: Store $(\Phi, \rho, r, \Phi', d)$ in the replay buffer $\mathcal{D}$.
- 5: Reset the environment if the state Φ' is terminal.
- 6: **if** its time to update agent and adversary model **then**
- 7: $\omega_t, \omega_t^{(1)} \leftarrow \omega_t, \theta_t^{(1)} \leftarrow \theta_t$
- 8: **for** $k = 1, 2, \dots, K_t$ **do**
- 9: Sample a random minibatch of N transitions $B = \{(\Phi, \bar{\rho}, r, \Phi', d)\}$ from $\mathcal{D}$.
- 10: Compute the targets as per (17).
- 11: Compute the loss function to update the critic $\nabla_{\phi} L(\phi)$ as per (18).
- 12: Compute the agent and adversary policy gradient estimate as per (19) and (20).
- 13: Compute the agent and adversary parameters as per (21).
- 14: Update the parameters of the agent and adversary network $\{\theta, \omega\}$ as per (24) and (25).
- 15: Obtain the value of $\{\bar{\theta}, \bar{\omega}\}$ as per (26).
- 16: Update the parameters of the target network used to compute the target as per (17).

$$\begin{aligned} \phi_{\text{targ}} &\leftarrow \tau \phi_{\text{targ}} + (1-\tau) \phi \\ \theta_{\text{targ}} &\leftarrow \tau \theta_{\text{targ}} + (1-\tau) \theta_t^{k+1} \\ \omega_{\text{targ}} &\leftarrow \tau \omega_{\text{targ}} + (1-\tau) \omega_t^{k+1} \end{aligned} \quad (23)$$

17: **end for**

18: Update $\{\omega_{t+1}, \theta_{t+1}\}$ as per (27).

19: $t \leftarrow t + 1$

20: **end if**

21: **until** convergence.

Output: $\{\omega_T, \theta_T\}$

1) Deep Deterministic Policy Gradient (DDPG)

As per the schematic in Figure 4, the adversarial action robust RL is replaced by DDPG, where only agent has to be trained to deal with the conflict induced spoofing attacks. The critic targets have to be updated as per the following equation,

$$\vartheta_{\text{DDPG}}(r, \Phi', d) = r + \gamma(1-d) Q_{\phi_{\text{targ}}}(\Phi', \rho_{\text{agent}}). \quad (28)$$

The following change has to be incorporated as compared to the policy update equation described in Section III-D in (19),

$$\nabla_{\theta} J_{DDPG}(\theta) = \frac{1}{N} \sum_{\Phi \in \mathcal{D}} \nabla_{\theta} \mu_{\theta}(\Phi) \nabla_{\rho_{\text{agent}}} Q_{\phi}(\Phi, \rho_{\text{agent}}). \quad (29)$$

The update equations in (28) and (29) does not model the adversarial impact due to the random traffic conditions in the airspace as has been considered in Section III-D.

2) Non-Zero Sum Adversarial Robust RL (NZSAR RL)

A two-player non-zero-sum stochastic game has been modelled in [38], where the agent aims to minimize the impact of the spoofing attack due to the randomness of airspace traffic, while iteratively considering the adversarial impact. The value function to be maximized at time t is given as $Q(\rho_{\text{agent}}, \rho_{\text{adv}}, \Phi_t)$. The MDP in Section III-D has been designed while considering a zero-sum game where reward (13) is used. Hence for non-zero sum game the reward for the adversary which is the modified version as per (13) is given as:

$$R_{\text{adv}} = w_1 \cdot \sum_{N_{\text{steps}}} |\Delta \mathbf{V}_{\text{cor}} - \Delta \mathbf{V}_{\text{cor}}^{\text{sp}} - \Delta \mathbf{V}_{\text{agent}}| + w_2 \cdot N_{\text{con}} \cdot r_{\text{con}} + w_3 \cdot r_{\text{col}}. \quad (30)$$

The adversarial reward R_{adv} defined in (30) signifies the reward obtained by the adversary, if the necessary action from the agent under spoofed scenario does not help achieve the UAV necessary velocity which is obtained from the unspoofed scenario. The adversarial rewards from (30) are incorporated in the buffer $\mathcal{D} = (\Phi, \rho, r, r_{\text{adv}}, \Phi', d)$, where the samples are used to train the policies and critics of adversarial and agent networks. Unlike the zero-sum action robust RL as described in Section III-D, non-zero sum case, involves independent critics for agent Q_{agent} and adversary Q_{adv} . Hence the critic target for the agent and the adversary with separate rewards r_{ep} and r_{adv} respectively, is defined as follows:

$$\begin{aligned} v_{\text{agent}}(r_{\text{ep}}, \Phi'_{\text{agent}}, d) &= r_{\text{ep}} + \gamma(1-d) Q_{\phi_{\text{arg}}}^{\text{agent}}(\Phi'_{\text{agent}}, \rho) \\ v_{\text{adv}}(r_{\text{adv}}, \Phi'_{\text{adv}}, d) &= r_{\text{adv}} + \gamma(1-d) Q_{\phi_{\text{arg}}}^{\text{adv}}(\Phi'_{\text{adv}}, \rho_{\text{adv}}) \end{aligned} \quad (31)$$

where, $\Phi'_{\text{agent}} = \{\Phi', \rho_{\text{adv}}\}$ and $\Phi'_{\text{adv}} = \{\Phi', \rho\}$. Here ρ refers to the overall action is $\rho = \alpha \cdot \rho_{\text{adv}} + (1-\alpha) \cdot \mu_{\text{agent}}(\Phi_{\text{agent}})$, where μ_{agent} is the agent policy. Using these, returned values for agent and adversary, the critic loss function is independently framed for both agent and adversary, as

$$\begin{aligned} \nabla_{\phi} L^{\text{agent}}(\phi) &= \frac{1}{N} \sum_{(\Phi, \rho, r, r_{\text{adv}}, \Phi', d) \in B} (v_{\text{agent}} - Q^{\text{agent}})^2, \\ \nabla_{\phi} L^{\text{adv}}(\phi) &= \frac{1}{N} \sum_{(\Phi, \rho, r, r_{\text{adv}}, \Phi', d) \in B} (v_{\text{adv}} - Q^{\text{adv}})^2. \end{aligned} \quad (32)$$

Similarly policy gradient for both the agent and the adversary is written as:

$$\begin{aligned} \nabla_{\theta_{\text{agent}}} J(\theta_{\text{agent}}) &= \frac{1}{N} \sum_{\mathcal{D}} \nabla_{\theta_{\text{agent}}} \mu_{\theta_{\text{agent}}} \cdot \nabla_{\rho} Q_{\phi}^{\text{agent}}, \\ \nabla_{\theta_{\text{adv}}} J(\theta_{\text{adv}}) &= \frac{1}{N} \sum_{\mathcal{D}} \nabla_{\theta_{\text{adv}}} \mu_{\theta_{\text{adv}}} \cdot \nabla_{\rho} Q_{\phi}^{\text{adv}}. \end{aligned} \quad (33)$$

In the non-zero-sum game framework, the adversary has a distinct reward function as defined in equation (30), resulting in independent critic targets for both the agent and the adversary, unlike the shared critic target in the zero-sum scenario. As outlined in [38], adversarial actions influence the agent's state space and vice versa, forming a stochastic game. Regarding the implementation described in Algorithm 2, the process begins by calculating the agent's state using a random adversarial action. Subsequently, the agent's action is determined, becoming part of the adversarial state. The adversary's action, computed from this state, is incorporated into the agent's state in the following iteration. This iterative process allows for managing the adversarial impact under random traffic conditions in the airspace using a non-zero-sum agent-adversary game.

IV. RESULTS AND DISCUSSION

A. Numerical Implementation

The numerical configuration aligns with our prior study [39], wherein the simulation setup has been down-scaled by a factor of 30. The simulation setup of the 2D airspace traffic model has been designed as per the Github repository in [50]. The airspace, configured as a square with each side measuring 24 km, is central to the investigation. The conflict radius is set at 500 m, and the proximity for detecting nearby aircraft is established at 50 m, indicative of a potential near-midair collision. Air traffic is centralized, with missions assigned to each aircraft without incorporating conflict resolution manoeuvres. The maximum speed of the ownship UAV is stipulated at 80 m/s (approx. 178 mph), while the speed of other aircraft varies randomly within the range of 0 to 80 m/s.

A vertiport structure at the airspace midpoint (12, 12) km is considered as destination for the ownship UAV. Its starting point is defined at the vertices of the square vertiport system, located 1.5 km from the airspace's edge. The journey from the vertex of the vertiport structure to the airspace midpoint can be considered as the journey from a rural area with sparse air traffic to an urban setting with dense air traffic. The trajectory of other aircraft traverses the airspace centre, necessitating the ownship UAV to navigate through a densely populated airspace to fulfil its designated mission. Consequently, the potential compromise of the mission, as explained in the subsequent subsection, can occur through conflict-induced spoofing, as detailed in Algorithm 1.

B. Effect of Induced Spoofing Attack on MVP Conflict Resolution

Subsection II-A explains how the ownship UAV reacts when spotting an intruder within a certain radius with the help of UAM operator. Otherwise, it follows its planned straight path using the desired speed v_{des} . However, during a conflict-induced spoofing attack at a probability p_{att} . The impact of this spoofing on metrics like steps, conflict instances, and missions is shown in Figures 5 and 6. Different mission scenarios with various starting points and 20 aircraft on random paths were considered. Figure 5 shows that non-zero conflicts increase as the attack probability rises, peaking at 90%. There's more

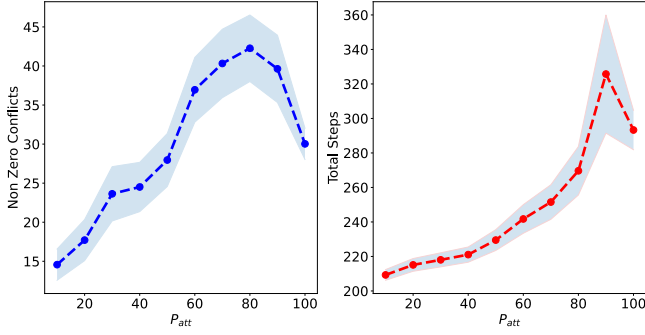


Fig. 5. The variation of the mean and standard deviation of non-zero conflicts and total steps to complete the mission for the ownship UAV for different percentage probability of the conflict-induced spoofing attack.

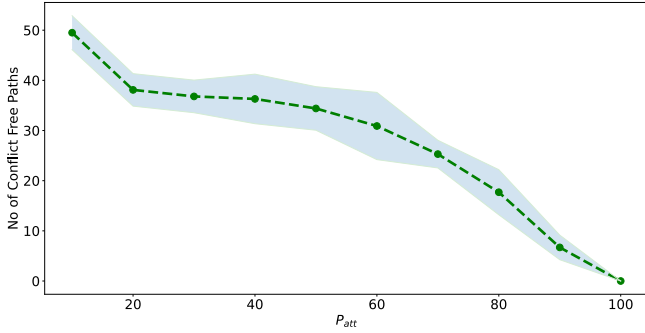


Fig. 6. The variation of the mean and standard deviation of a number of conflict-free paths for the ownship UAV for different percentage probability of the conflict-induced spoofing attack.

uncertainty in conflicts and steps beyond 70% attack probability. Steps increase linearly compared to non-zero conflicts. At 100% attack probability, indicating complete spoofing, there's a high collision risk, leading to more number of incomplete missions, especially after 90%. This trend is visible in Figure 6, where non-zero conflicts drop to zero at 100% attack probability. So, the geometric MVP shows robustness at lower spoofing probabilities, as seen in AAM metrics in Figures 5 and 6. Therefore, an RL strategy must be devised to counter higher-probability attacks. The flooding of UAV drone processors with spoofed data significantly increases the risk of the mission being compromised. The flood of false information to the UAV processor significantly increases the risk of mission compromise. Hence, it is essential to design robust RL at a higher probability of conflict-induced spoofing attack a sit caused increased conflict and fuel consumption.

C. Reinforcement Learning Training

The action-robust DDPG model was trained according to Algorithm 2, using parameters from Table I and running 10 Monte Carlo simulations. Action robust RL has been designed as per the Github implementation in [51]. It randomly executed 15000 steps to gather experiences for training the adversarial and agent neural networks. Training considered generating adversarial samples at attack probabilities $p_{\text{train}} = 0.8, 0.9$, and 1.0. As per Figure 7 the rewards converges after the exploration phase quickly for p_{train} of 0.8 and 0.9, while the fluctuation is quite high when $p_{\text{train}} = 1.0$. The rewards at

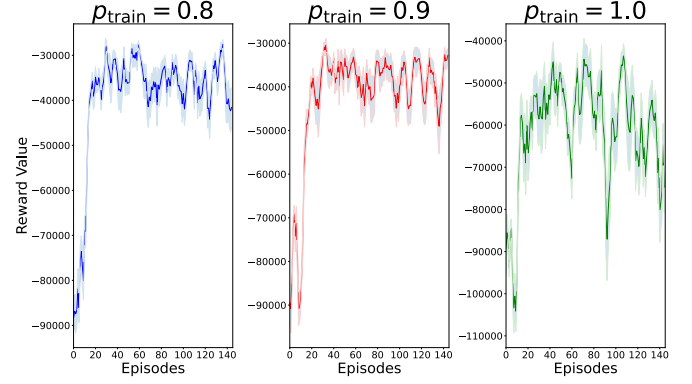


Fig. 7. The variation of the reward along with its uncertainty ZSAR-DDPG over 10 Monte Carlo runs.

TABLE I
ACTION ROBUST RL PARAMETERS

Parameter	Value
Number of aircrafts	20
p_{att}	{0.8, 0.9, 1.0}
Replay buffer size	1e6
Minibatch size	128
Discount factor	0.99
Target network soft update rate	0.01
Agent and adversary network learning rate	0.01
Agent and adversary network momentum	0.9
Adversarial action ratio α	0.2
Adversary noise parameter	0.1
Weight decay	5e-4
Critic learning rate	1e-3
Neural network optimizer	Adam
Exploration noise	$\mathcal{N}(0.05, 0.02)$
Maximum steps per episode	1500
Total number of episodes	150
Total neural network layers	3
Robustness update rate (β)	0.9
Number of neurons in each layer	128
Minimum velocity magnitude	0.75
Heading correction range	$[-\pi, \pi]$

$p_{\text{train}} = 1.0$ is comparatively worse than the other cases due to higher number of conflicts induced at higher probabilities as shown in Figure 5. The convergence of ZSAR based MVP navigation strategy as per Figure 3 showed better convergence than non-geometry-guided RL algorithms in [52].

D. Ablation Study on Deconfliction

The trained models at various spoofing probabilities p_{train} from the previous section were tested against random number of air traffic intruders between 15 and 24 and conflict-induced spoofing attacks at probabilities p_{att} of 0.8, 0.9, and 1.0 across 100 episodes as per the schematic in Figure 3. The metrics evaluated included the number of conflict-free navigation paths and the number of conflicts over these episodes. The results were compared with the MVP strategy for navigation without incorporating RL as a secondary system to defend against spoofing attacks. As shown in Figure 8, integrating the ZSAR RL strategy significantly improves the number of conflict-free paths across all probabilistic attack scenarios. However, there is a noticeable increase in variability in conflict-free

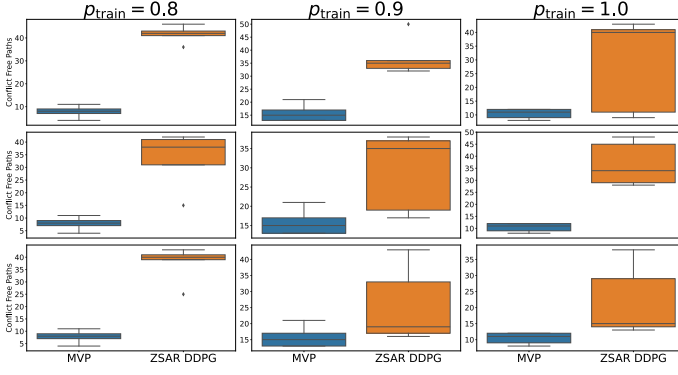


Fig. 8. The box-plot signifying the number of conflict-free paths for MVP with ZSAR strategy along with pure MVP based UAV navigation against conflict induced spoofing. The rows signifies the performance against attack probabilities p_{att} (a) 80% (b) 90% (c) 100%.

paths with models trained at $p_{train} = 0.9$ and 1.0 compared to $p_{train} = 0.8$ under different attack scenarios. A similar trend is observed in the number of conflicts: the ZSAR RL strategy consistently reduces the number of conflicts for $p_{train} = 0.8$, 0.9 and 1.0 compared to the pure MVP strategy. The reduction in average conflicts is more pronounced for $p_{train} = 0.9$ and 1.0 than for $p_{train} = 0.8$ across all attack scenarios. Thus, we

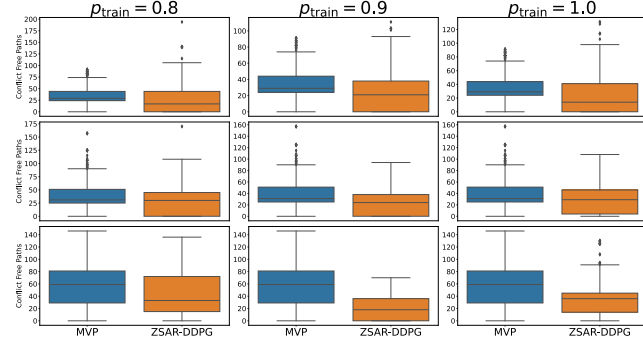


Fig. 9. The box-plot signifying the number of conflicts over 100 testing episodes for MVP with ZSAR strategy along with pure MVP based UAV navigation against conflict induced spoofing. The rows signifies the performance against attack probabilities p_{att} (a) 80% (b) 90% (c) 100%.

conclude that the RL strategy combined with MVP enhances the number of conflict-free paths and reduces the number of conflicts under conflict-induced spoofing attacks compared to using the MVP strategy alone. However, the variability in conflict-free paths is lower when the model is trained at lower spoofing probabilities, while the average number of conflicts is minimized more effectively when trained at higher spoofing probabilities.

E. Comparison with the SOTA RL Deconfliction

First, we assess the convergence of the state-of-the-art (SOTA) methods described in Section III-E. As observed in Figure 10, the reward plot for SOTA shows slower convergence with higher variability compared to the ZSAR convergence shown in Figure 7. Specifically, for models trained with a deterministic spoofing attack $p_{train} = 1.0$, where the number of conflict-free paths is zero, the SOTA convergence is significantly

poorer with lower rewards compared to ZSAR. Additionally, the SOTA algorithms exhibit higher variability in rewards than the ZSAR algorithm, as illustrated in Figure 7. Now we

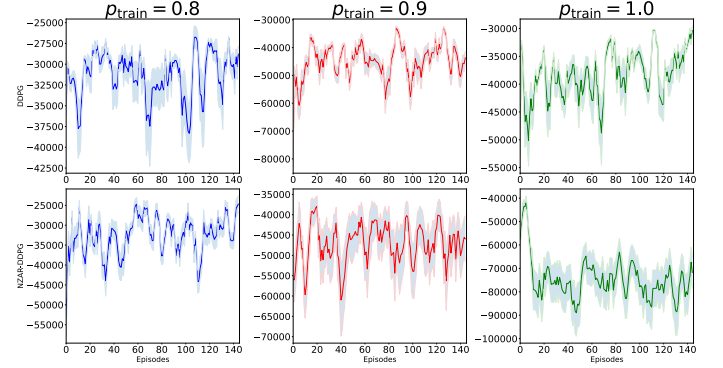


Fig. 10. The variation of the reward along with its uncertainty zero-sum action robust DDPG over 10 Monte Carlo runs

need to test the trained SOTA RL model aided MVP based navigation strategy with random number of airspace traffic, while comparing the number of conflict free paths and conflicts over 100 episodes. As we see in Figure 11, the number of conflict free path is higher for ZSAR as compared to SOTA for different test spoofing attack probabilities while comparing the models trained at the spoofing probability of $p_{train} = 0.8$, even with lower variability. Similarly, we also observe that number of conflict free path is higher for $p_{train} = 1.0$ for ZSAR RL algorithms as compared to SOTA algorithms. However, in this case, the variability in number of conflict-free path is higher as compared to the variability in SOTA algorithms. But we do see a different trend when comparing different attack probabilities for $p_{train} = 0.9$, where the number of conflict-free path is higher for NZSAR RL as compared to ZSAR RL. However, the average number of conflict free path is more for ZSAR RL strategy as compared to NZSAR RL strategy. The anomaly

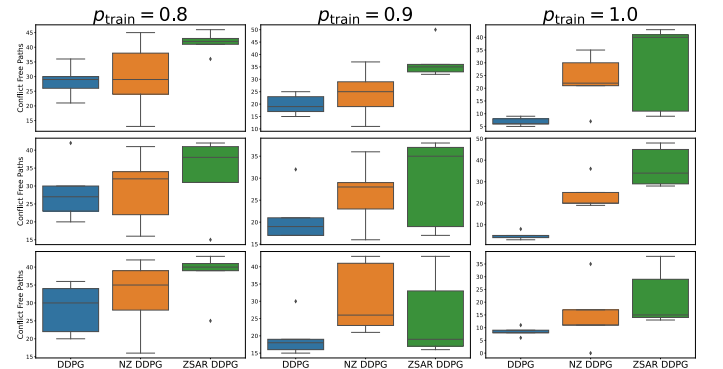


Fig. 11. The box-plot signifying the number of conflict-free paths for MVP with ZSAR strategy along with SOTA DDPG and NZSAR-DDPG based UAV navigation against conflict induced spoofing. The rows signifies the performance against attack probabilities p_{att} (a) 80% (b) 90% (c) 100%.

in the trend can be explained from the conflict plot shown in Figure 12. We observe that even though the number of conflict free path for ZSAR RL strategies have been reduced as

compared to NZSAR RL strategies for most of the iterations, number of conflicts over the episodes has reduced for NZSAR RL as compared to ZSAR RL strategy. Moreover, the average number of conflict free paths for ZSAR RL strategy is more as compared to NZSAR RL strategy. Hence, we can conclude that the ZSAR strategy aided MVP based UAV navigation can considerably improve the number of conflict free paths as compared to SOTA NZSAR and DDPG aided MVP based navigation strategy. However, for higher attack trained models, the increase in conflict free strategies comes at the price of higher variability across Monte Carlo simulation runs.

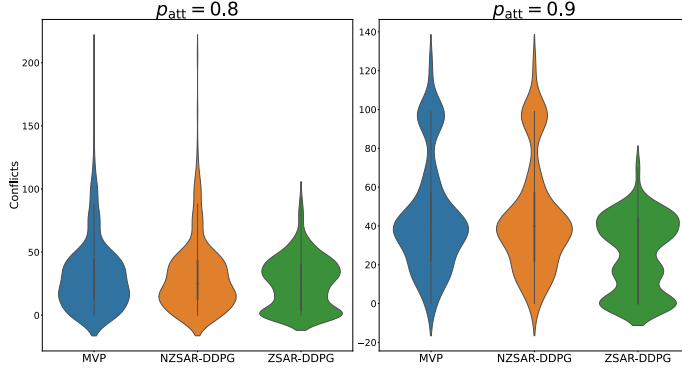


Fig. 12. The variation of the conflicts obtained over 100 testing episodes for MVP with ZSAR strategy along with SOTA DDPG and NZSAR-DDPG based UAV navigation against conflict induced spoofing while considering the $p_{train} = 0.9$.

F. Total Number of Steps Comparison

As observed in Figure 13, the MVP navigation with the RL setup shown in Figure 3 reduces the number of steps only when deterministic spoofing attack is considered. However, there is no significant improvement in the number of steps required to complete the mission when comparing the ZSAR RL strategy to other SOTA RL algorithms described in Section III-E. This indicates that the setup in Figure 3 makes the UAV more conservative in mission planning at lower attack probabilities, resulting in more steps to complete the mission compared to the setup without the RL strategy.

G. Advantages and Limitations

Based on the metrics discussed above in this subsection the overall advantages are summarized and the limitations are also discussed.

1) Advantages

The advantages of the proposed algorithm can be summarized as follows:

- **Enhanced Navigation against Spoofing Attacks:** The setup depicted in Figure 3 enhances UAV navigation by reducing the average number of conflicts and increasing the average number of conflict-free paths in the presence of conflict-induced spoofing attacks, as shown in Table II. However, it only improves fuel efficiency under deterministic attack conditions.
- **Improved Conflict-Free Paths with ZSAR aided MVP:** The ZSAR with MVP strategy significantly increases the

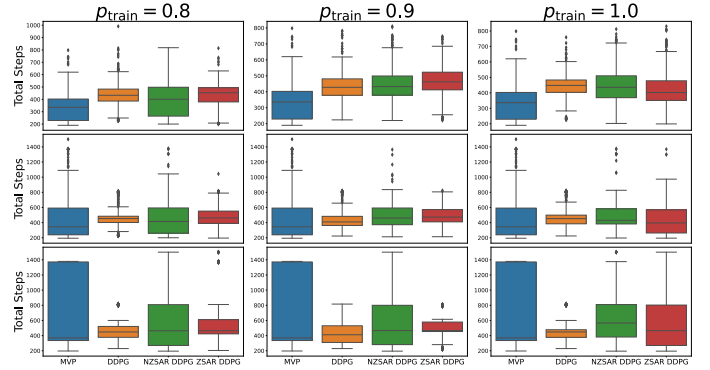


Fig. 13. The box-plot signifying the number of steps to complete the mission for MVP with ZSAR strategy along with SOTA DDPG and NZSAR-DDPG based UAV navigation against conflict induced spoofing. The rows signifies the performance against attack probabilities p_{att} (a) 80% (b) 90% (c) 100%.

average number of conflict-free paths compared to the SOTA NZSAR and DDPG-based MVP navigation strategies in random airspace traffic scenarios, as indicated in Table II.

- **Reduced Conflicts in Stochastic Spoofing Attack Scenarios:** The ZSAR with MVP-based UAV navigation strategy reduces the number of conflicts under stochastic conflict-induced spoofing attack scenarios, as demonstrated in Figure 12.

Considering the advantages practical impact can be stated as follows:

- **Reduction in Sensor Cost:** Utilizing RL algorithms as a secondary system to counter spoofing attacks offers a cost-effective alternative to using IMU sensors alongside ADS-B, thereby reducing overall expenses.
- **Negating the Impact of ADS-B Data Asynchronization:** An increased number of conflict-free navigation instances decreases the need for detect-and-avoid (DAA) alerts. This reduction lowers the computation time required for navigation control and minimizes the impact of ADS-B data synchronization on navigation planning.
- **Improved Safety:** Fewer conflicts provide UAVs with sufficient time to avoid collisions, enhancing safety even in the presence of spoofing attacks, thereby allowing the UAVs to complete their missions successfully.
- **Improved Autonomy:** The autonomous defence strategy as shown in Figure 3 will reduce the ATC workload, hence improving the autonomy levels of the unmanned aircrafts [53] for safer integration in the airspace.

2) Limitations

The limitations of this study can be summarized as follows:

- The strategy depicted in Figure 3 is ineffective at lower attack probabilities or under normal conditions, as it does not show considerable improvement.
- As shown in Figure 13, the fuel required to complete the mission increases when deploying RL algorithms with the MVP strategy, indicating a need for a larger energy supply to defend against attack scenarios, which can raise the design and operational cost.

TABLE II

COMPARISON OF THE AVERAGE NUMBER OF CONFLICT FREE PATHS FOR MVP, MVP AIDED DDPG, NZSAR RL, ZSAR RL OVER 100 TESTING EPISODES AGAINST 10 MONTE CARLO SIMULATION RUNS

Attack Probabilities	$p_{\text{train}} = 0.8$				$p_{\text{train}} = 0.9$				$p_{\text{train}} = 1.0$			
	MVP	DDPG	NZSAR	ZSAR	MVP	DDPG	NZSAR	ZSAR	MVP	DDPG	NZSAR	ZSAR
$p_{\text{att}} = 0.8$	7.80	28.4	29.8	41.6	7.80	28.4	29.0	33.4	7.80	28.4	32.0	37.6
$p_{\text{att}} = 0.9$	15.8	19.8	24.2	37.2	15.8	21.2	26.4	29.2	15.8	19.6	30.8	25.6
$p_{\text{att}} = 1.0$	10.4	6.80	23.0	28.8	10.4	5.00	24.0	36.8	10.4	8.40	14.8	21.8

- The setup described in Figure 3 involves data exchange among several actors, which can lead to data synchronization issues and potentially result in an overall ineffective defense against spoofing attacks. This area will be explored in future papers.

V. CONCLUSION

In this paper, a zero-sum action robust (ZSAR) DDPG-aided MVP-based deconflicted navigation system has been designed, which will facilitate the ownship UAV to fulfil its mission against uncertain novel dynamic conflict-induced spoofing attacks and dynamic air traffic. The algorithm is compared against the SOTA NZSAR and DDPG aided MVP-based deconflicted navigation strategy against novel conflict-induced spoofing strategy. Final results can be summarized as follows:

- MVP is resilient against novel conflict spoofing attack at lower probabilities, however the performance degrades rapidly with increase in attack probability.
- RL aided MVP strategy considerably improves the deconfliction in random airspace traffic scenarios against various stochastic and deterministic conflict-induced spoofing attack scenarios.
- ZSAR with MVP strategy considerably improves the average number of conflict-free paths compared to the SOTA NZSAR and DDPG-based MVP navigation strategy against random airspace traffic and conflict-induced attack scenarios.
- ZSAR with MVP based UAV navigation strategy improves the number of conflicts under stochastic conflict induced spoofing attack scenarios.

Future work revolves around designing an adaptive AI algorithm which will fulfill the safety as well with optimal fuel efficiency while maintaining operational efficiency at normal scenarios. The setup described in this paper serves as a preliminary investigation into future unmanned air traffic management infrastructure. Due to the exploratory nature of this study, real-world data and scenarios were not available. Future research will focus on integrating the real-world trajectory data of manned aviation flights with unmanned air traffic management systems.

REFERENCES

- [1] A. P. Cohen, S. A. Shaheen, and E. M. Farrar, "Urban air mobility: History, ecosystem, market potential, and challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 6074–6087, 2021.
- [2] S. Li, X. Cheng, X. Huang, S. A. Otaibi, and H. Wang, "Cooperative conflict detection and resolution and safety assessment for 6g enabled unmanned aerial vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 2183–2198, 2023.
- [3] E. U. S. Agency. "Artificial intelligence roadmap 2.0." (2023), [Online]. Available: <https://www.easa.europa.eu/en/document-library/general-publications/easa-artificial-intelligence-roadmap-20> (visited on 01/10/2024).
- [4] K.-F. Chu and W. Guo, "Privacy-preserving federated deep reinforcement learning for mobility-as-a-service," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 2, pp. 1882–1896, 2024.
- [5] M. R. Manesh and N. Kaabouch, "Cyber-attacks on unmanned aerial system networks: Detection, countermeasure, and future research directions," *Computers & Security*, vol. 85, pp. 386–401, 2019.
- [6] L. Xiao, X. Lu, D. Xu, Y. Tang, L. Wang, and W. Zhuang, "Uav relay in vanets against smart jamming with reinforcement learning," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4087–4097, 2018. DOI: [10.1109/TVT.2018.2789466](https://doi.org/10.1109/TVT.2018.2789466).
- [7] T. Liu, C. Yang, X. Liu, R. Han, and J. Ma, "Rpau: Fooling the eyes of uavs via physical adversarial patches," *IEEE Transactions on Intelligent Transportation Systems*, 2023.
- [8] K.-F. Chu and W. Guo, "Passenger spoofing attack for artificial intelligence-based mobility-as-a-service," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2023.
- [9] Y. Guo, M. Wu, K. Tang, J. Tie, and X. Li, "Covert spoofing algorithm of uav based on gps/ins-integrated navigation," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 6557–6564, 2019.
- [10] P. Geragersian, I. Petrunin, W. Guo, and R. Grech, "A hybrid deep learning approach for robust multi-sensor GNSS/INS/VO fusion in urban canyons," in *Proceedings of the 36th International Technical Meeting of the Satellite Division of The Institute of Navigation, ION*, 2023, pp. 2624–2643.
- [11] Y. I. Jenie, E.-J. Van Kampen, J. Ellerbroek, and J. M. Hoekstra, "Taxonomy of conflict detection and resolution approaches for unmanned aerial vehicle in an integrated airspace," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 558–567, 2016.
- [12] M. Ribeiro, J. Ellerbroek, and J. Hoekstra, "Review of conflict resolution methods for manned and unmanned aviation," *Aerospace*, vol. 7, no. 6, p. 79, 2020.
- [13] J. Yang, Y. Hu, Z. Yu, F. Chen, and X. Xu, "In-depth coordination and extension: Decentralized onboard conflict resolution of uavs in the low altitude airspace," *IEEE Transactions on Intelligent Vehicles*, pp. 1–14, 2023.
- [14] I. Erev, G. Barron, and R. Remington, "Right of way in the sky: Two problems in aircraft self-separation and the auction-based solution," *Human Factors*, vol. 46, no. 2, pp. 267–276, 2004.
- [15] M. A. Piera, J. J. Ramos, R. M. Ortiz, and M. Radanovic, "A negotiation mechanism for a consensus in air traffic conflict resolution," in *2018 International Conference on Advances*

- in *Big Data, Computing and Data Communication Systems (icABCD)*, IEEE, 2018, pp. 1–8.
- [16] J. Yang, D. Yin, Q. Cheng, and L. Shen, “Two-layered mechanism of online unmanned aerial vehicles conflict detection and resolution,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 10, pp. 3230–3244, 2017.
 - [17] J. M. Hoekstra, R. N. van Gent, and R. C. Ruigrok, “Designing for safety: The ‘free flight’ air traffic management concept,” *Reliability Engineering & System Safety*, vol. 75, no. 2, pp. 215–232, 2002.
 - [18] Y. I. Jenie, E.-J. v. Kampen, C. C. de Visser, J. Ellerbroek, and J. M. Hoekstra, “Selective velocity obstacle method for deconflicting maneuvers applied to unmanned aerial vehicles,” *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 6, pp. 1140–1146, 2015.
 - [19] S. Li, M. Egorov, and M. Kochenderfer, “Optimizing collision avoidance in dense airspace using deep reinforcement learning,” *Thirteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM)*, 2019.
 - [20] M. Zhang, C. Yan, W. Dai, X. Xiang, and K. H. Low, “Tactical conflict resolution in urban airspace for unmanned aerial vehicles operations using attention-based deep reinforcement learning,” *Green Energy and Intelligent Transportation*, vol. 2, no. 4, p. 100 107, 2023.
 - [21] M. W. Brittain and P. Wei, “One to any: Distributed conflict resolution with deep multi-agent reinforcement learning and long short-term memory,” in *AIAA Scitech 2021 Forum*, 2021, p. 1952.
 - [22] D.-T. Pham, N. P. Tran, S. K. Goh, S. Alam, and V. Duong, “Reinforcement learning for two-aircraft conflict resolution in the presence of uncertainty,” in *2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF)*, IEEE, 2019, pp. 1–6.
 - [23] R. Isufaj, D. Aranega Sebastia, and M. Angel Piera, “Toward conflict resolution with deep multi-agent reinforcement learning,” *Journal of Air Transportation*, vol. 30, no. 3, pp. 71–80, 2022.
 - [24] M. W. Brittain, X. Yang, and P. Wei, “Autonomous separation assurance with deep multi-agent reinforcement learning,” *Journal of Aerospace Information Systems*, vol. 18, no. 12, pp. 890–905, 2021.
 - [25] P. Razzaghi *et al.*, “A survey on reinforcement learning in aviation applications,” *arXiv preprint arXiv:2211.02147*, 2022.
 - [26] M. Ribeiro, J. Ellerbroek, and J. Hoekstra, “Determining optimal conflict avoidance manoeuvres at high densities with reinforcement learning,” in *Proceedings of the Tenth SESAR Innovation Days, Virtual Conference*, 2020, pp. 7–10.
 - [27] J. Zhang and C. Li, “Adversarial examples: Opportunities and challenges,” *IEEE transactions on neural networks and learning systems*, vol. 31, no. 7, pp. 2578–2593, 2019.
 - [28] T. Hickling, N. Aouf, and P. Spencer, “Robust adversarial attacks detection based on explainable deep reinforcement learning for uav guidance and planning,” *IEEE Transactions on Intelligent Vehicles*, 2023.
 - [29] J. Tian, B. Wang, R. Guo, Z. Wang, K. Cao, and X. Wang, “Adversarial attacks and defenses for deep-learning-based unmanned aerial vehicles,” *IEEE Internet of Things Journal*, vol. 9, no. 22, pp. 22 399–22 409, 2021.
 - [30] S. Chen, G. Liu, Z. Zhou, K. Zhang, and J. Wang, “Robust multi-agent reinforcement learning method based on adversarial domain randomization for real-world dual-uav cooperation,” *IEEE Transactions on Intelligent Vehicles*, 2023.
 - [31] D. Silver *et al.*, “Mastering the game of go without human knowledge,” *nature*, vol. 550, no. 7676, pp. 354–359, 2017.
 - [32] S. Levine, C. Finn, T. Darrell, and P. Abbeel, “End-to-end training of deep visuomotor policies,” *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1334–1373, 2016.
 - [33] X. He and C. Lv, “Robotic control in adversarial and sparse reward environments: A robust goal-conditioned reinforcement learning approach,” *IEEE Transactions on Artificial Intelligence*, 2023.
 - [34] C. Wang and N. Aouf, “Explainable deep adversarial reinforcement learning approach for robust autonomous driving,” *IEEE Transactions on Intelligent Vehicles*, 2024.
 - [35] X. Y. Lee, Y. Esfandiari, K. L. Tan, and S. Sarkar, “Query-based targeted action-space adversarial policies on deep reinforcement learning agents,” in *Proceedings of the ACM/IEEE 12th international conference on cyber-physical systems*, 2021, pp. 87–97.
 - [36] J. Parras and S. Zazo, “Robust deep reinforcement learning for underwater navigation with unknown disturbances,” in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2021, pp. 3440–3444.
 - [37] H. Benaddi, M. Jouhari, K. Ibrahim, J. Ben Othman, and E. M. Amhoud, “Anomaly detection in industrial iot using distributional reinforcement learning and generative adversarial networks,” *Sensors*, vol. 22, no. 21, p. 8085, 2022.
 - [38] H. Benaddi, K. Ibrahim, A. Benslimane, M. Jouhari, and J. Qadir, “Robust enhancement of intrusion detection systems using deep reinforcement learning and stochastic game,” *IEEE Transactions on Vehicular Technology*, vol. 71, no. 10, pp. 11 089–11 102, 2022.
 - [39] D. K. Panda and W. Guo, “Fragility impact of rl based advanced air mobility under gradient attacks and packet drop constraints,” in *2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*, IEEE, 2023, pp. 1–7.
 - [40] J. Maas, E. Sunil, J. Ellerbroek, and J. Hoekstra, “The effect of swarming on a voltage potential-based conflict resolution algorithm,” in *7th International Conference on Research in Air Transportation*, 2016.
 - [41] A. C. Tang, “A review on cybersecurity vulnerabilities for urban air mobility,” in *AIAA Scitech 2021 Forum*, 2021, p. 0773.
 - [42] A. Ramanandan, A. Chen, and J. A. Farrell, “Inertial navigation aiding by stationary updates,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 235–248, 2011.
 - [43] D. Howell and J. King, “Measured impact of ads-b in applications on general aviation and air taxi accident rates,” in *2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC)*, IEEE, 2019, pp. 1–9.
 - [44] L. Pinto, J. Davidson, R. Sukthankar, and A. Gupta, “Robust adversarial reinforcement learning,” in *International Conference on Machine Learning*, PMLR, 2017, pp. 2817–2826.
 - [45] C. Tessler, Y. Efroni, and S. Mannor, “Action robust reinforcement learning and applications in continuous control,” in *International Conference on Machine Learning*, PMLR, 2019, pp. 6215–6224.
 - [46] Y.-P. Hsieh, C. Liu, and V. Cevher, “Finding mixed nash equilibria of generative adversarial networks,” in *International Conference on Machine Learning*, PMLR, 2019, pp. 2810–2819.
 - [47] P. Kamalaruban, Y.-T. Huang, Y.-P. Hsieh, P. Rolland, C. Shi, and V. Cevher, “Robust reinforcement learning via adversarial training with langevin dynamics,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 8127–8138, 2020.
 - [48] M. L. Littman, “Markov games as a framework for multi-agent reinforcement learning,” in *Machine learning proceedings 1994*, Elsevier, 1994, pp. 157–163.
 - [49] C. A. Bhardwaj, “Adaptively preconditioned stochastic gradient langevin dynamics,” *arXiv preprint arXiv:1906.04324*, 2019.
 - [50] <https://github.com/WLeiii/Gym-ATC-Attack-Project>, [Accessed 16-06-2024].
 - [51] <https://github.com/ythuangyt/Robust-Reinforcement-Learning-via-Adversarial-training-with-Langevin-Dynamics>, [Accessed 16-06-2024].

- [52] L. Wang, H. Yang, Y. Lin, S. Yin, and Y. Wu, “Explainable and safe reinforcement learning for autonomous air mobility,” *arXiv preprint arXiv:2211.13474*, 2022.
- [53] “Unmanned Aircraft System Operations in UK Airspace – Policy and Guidance,” UK Civil Aviation Authority, Tech. Rep., Apr. 2024.



DSPACE

<https://dspace.org/>

Action robust reinforcement learning for air mobility deconfliction against conflict induced spoofing

Panda, Deepak Kumar; Guo, Weisi

2024-12

Institute of Electrical and Electronics Engineers (IEEE)

<https://doi.org/10.1109/tits.2024.3454354>

Panda DK, Guo W. (2024) Action robust reinforcement learning for air mobility deconfliction against conflict induced spoofing. IEEE Transactions on Intelligent Transportation Systems, Volume 25, Issue 12, December 2024, pp. 21343-21355