UAB
Universitat Autònoma
de Barcelona

IMPLEMENTATION OF AN AI-ENHANCED UNMANNED TRAFFIC MANAGEMENT SYSTEM USING BLUESKY SIMULATION FOR URBAN AIR MOBILITY

Memory of the Final Thesis of Aeronautical Management Degree

By

Guillem Bertran Cañellas

Supervisor:

Ender Çetin

Escola d’Enginyeria

Sabadell, June of 2025

The undersigned, Ender Çetin, supervisor of the Final Degree Thesis, professor at the School of Engineering of UAB,

CERTIFIES:

That the work corresponding to this report has been carried out under their supervision by

Guillem Bertran Cañellas

And for the record, signs this document in Sabadell, June of 2025

CETIN ENDER - Y6196182L

Digitally signed by CETIN ENDER - Y6196182L
Date: 2025.06.25 20:59:11 +02'00'

Signed: Ender Çetin

ABSTRACT

This project explores the use of Artificial Intelligence in Air Traffic Management for decision-making and conflict-prediction tasks. Urban Air Mobility is a reality that requires cities to adapt and create new systems to control the dense urban airspace. The study includes a wide review of all Urban Air Mobility and Air Traffic Management concepts and limitations to give a based perspective of the topic. In addition, Machine Learning and Reinforcement Learning concepts will also be included to clearly understand how they work and how the experimentations will be conducted. The simulations will be conducted in the BlueSky Simulator environment through Python scripts. The investigation will focus on developing the proper environment for a drone agent which will be trained by interacting with other drones while moving towards a waypoint objective in a 2D scenario. Finally, the impacts produced by the project implementation will be slightly described, and the future works of the concept will be presented to conclude the thesis work.

Keywords: Air Traffic Management, Urban Air Mobility, Unmanned Traffic Management, eVTOLs, Reinforcement Learning, Proximal Policy Optimization, BlueSky Simulator, Multi-Agent Systems, Machine Learning.

ABSTRACTE

Aquest projecte explora l'ús d'Intel·ligència Artificial per a la gestió del trànsit aeri en tasques de presa de decisions i detecció de conflictes. El concepte de "Urban Air Mobility" (UAM) és una realitat que requereix que les ciutats s'adaptin de manera ràpida i creïn nous sistemes per tal de controlar el dens espai aeri urbà. L'estudi inclou una ampli repàs de tots els conceptes de UAM i gestió de trànsit aeri, així com les seves limitacions, per oferir una perspectiva més sòlida de l'àmbit. A més, per tal d'entendre amb més claredat com funcionaran i com es duran a terme les experimentacions, s'inclourà tambè una bona introducció dels conceptes de "Machine Learning" i de "Reinforcement Learning". Aquestes experimentacions seran realitzades a l'entorn de BlueSky Simulator mitjançant Python, i es centrarà en desenvolupar l'entorn idoni per un agent dron que serà entrenat a través de la interacció amb altres drons mentre avança cap a un determinat objectiu en un escenario de dues dimensions. Per últim, es revisaran els impactes derivats de la implementació del projecte i les futures línies d'investigació i millora d'aquest per concloure la tesi.

Paraules clau: gestió del tràfic aeri, mobilitat aèria urbana, gestió de tràfic no tripulat, vehicles d'enlairament i aterratge elèctric, aprenentatge per reforç. optimització propera de polítiques, simulador BlueSky, sistemes multi-agent, aprenentatge de màquines.

ABSTRACTO

Este proyecto explora el uso de Inteligencia Artificial para la gestión del tránsito aéreo en tareas de toma de decisiones y detección de conflictos. El concepto de “Urban Air Mobility” (UAM) es una realidad que requiere que las ciudades se adapten de manera rápida y creen nuevos sistemas para controlar el denso espacio aéreo urbano. El estudio incluye un amplio repaso de todos los conceptos de UAM y gestión de tránsito aéreo, así como sus limitaciones, para ofrecer una perspectiva más sólida del ámbito. Además, para entender con más claridad como funcionarán y como se realizarán las experimentaciones, se incluirá también una buena introducción de los conceptos de “Machine Learning” y de “Reinforcement Learning”. Estas experimentaciones serán realizadas en el entorno de BlueSky Simulator mediante Python, i se centrará en desarrollarl el entorno idóneo para un agente dron que será entrenado a través de la interacción con otros drons mientras avanza hacia un determinado objetivo en un escenario de dos dimensiones. Por último, se revisarán los impactos derivados de la implementación del proyecto y las futuras línias de investigación y mejora de este para concluir la tesis.

Palabras clave: gestión del tráfico aéreo, mobilidad aérea urbana, gestión de tráfico no tripulado, vehículos de despegue y aterrizaje eléctricos, aprendizaje por refuerzo, optimización próxima de políticas, simulador BlueSky, sistemas multi-agente, aprendizaje de máquinas.

INDEX

ACRONYM LIST

AI	Artificial Intelligence
ATM	Air Traffic Management
ATC	Air Traffic Control
ATFM	Air Traffic Flow Management
ATSP	Air Traffic Service Provider
ANN	Artificial Neural Network
ACC	Area Control Center
APP	Approach Control
A2G	Aircraft to Ground
A2A	Aircraft to Aircraft
CNN	Convolutional Neural Network
CISP	Common Information Service Provider
DAR	Dynamic Airspace Reconfiguration
DL	Deep Learning
DNN	Deep Neural Network
eVTOL	electric Vertical Take-Off Landing
FAA	Federal Aviation Administration
FIS	Flight Information Service
FSM	Finite State Machine
GAN	Generative Adversial Network
IoT	Internet of Things
MDP	Markov Decision Process
ML	Machine Learning
PPO	Proximal Policy Optimization
PG	Policy Gradient
RCM	Runway Configuration Management
RL	Reinforcement Learning
RNN	Recurrent Neural Network
SL	Supervised Learning
TRPO	Trust Region Policy Optimization
UAV	Unmanned Aerial Vehicle
UAM	Urban Air Mobility
UAS	Unmanned Aircraft System
UL	Unsupervised Learning
UTM	Unmanned Traffic Management
USSP	U-Space Service Provider

FIGURE LIST

1. INTRODUCTION

1.1 PROJECT DESCRIPTION

Urban Air Mobility (UAM) is an emerging transportation system that consists in the use of aerial vehicles to improve urban mobility. According to EASA (European Aviation Safety Agency), "UAM is a new safe, secure and more sustainable air transportation system for passengers and cargo in urban environments, enabled by new technologies and integrated into multimodal transportation systems. The transportation is performed by electric aircraft taking off and landing vertically, remotely piloted or with a pilot on board" [1]. By integrating UAM operations, EASA expects to produce a significant economic impact in EU with the creation of 90,000 jobs by 2030, and urban mobility will become safer due to the lower risk to have an accident in an air taxi and faster due to the saved time in comparison with current urban transportation. It also enhances urban environment by reducing significantly CO2 emissions for electric propulsion [1].

It is expected that commercial activities in EU are starting recently for goods deliveries and passenger transportations in piloted aircrafts, so besides these innovative aerial vehicles, the scenario requires an excellent design and application of new required infrastructures, such as Vertiports for air taxis and stations for delivery drones. In addition, it will also be essential to design an Air Traffic Management (ATM) system capable of managing this emerged urban air traffic that is practically going to conquer most of the main metropolitan areas in a short/medium term. Projects such as Sesar USpace or Next Generation Program aim to design and start implementing UAM imminently. However, managing this dense air traffic in city environments presents significant challenges in terms of safety, efficiency and scalability and at this moment is not being easy to develop an effective and clear system providing this safe and secure air traffic activity management.

By implementing the UAM project, the maximum safe workload of Air Traffic Control (ATC) is expected to be exceeded due to the huge projected scale and high density of UAM operations. Because of that, the clear necessity to develop and implement autonomous ATM solutions has emerged disruptively in the current scenario to take over functions such as aircraft management (for example for altitude adjustments or flightpath commands), ground holding times management or aircraft safe separations protection in order to avoid ATC collapse. In addition, the number of Urban Air Vehicles (UAVs) is expected to be high in a short to medium term, so automated ATM operations will be essential to ensure the proper and safe separation between UAVs, ensuring the required airspace density and operational efficiency. [2]

1.2 MOTIVATION

The motivation for doing this project comes from my interest in Air Traffic Management (ATM) and my intention of work as Air Traffic Controller in a future, so the opportunity to investigate and explore how ATM systems are expected to be in the incoming future is great for me to start introducing myself into the sector and gain clear insights of ATM future operations.

Also, the emergence and expected increase of Urban Air Mobility and how the whole world will adapt and develop new processes and infrastructure is an extremely relevant topic for me, as urban mobility will face one of the big technological and historic changes of all time with the introduction of air taxis and autonomous ATM systems.

Finally, the rapid emergence and constant improving of Artificial Intelligence (AI) with the purpose of assuming important responsibilities and automatize key processes in a short term forces us to understand and learn how to develop and utilize it as it will be absolutely required in a short future, and this project aims to explore AI capabilities while implementing them in ATM systems.

1.3 OBJECTIVES

This project explores the application of AI to develop an intelligent system capable of managing and controlling UAM traffic by using the BlueSky Air Traffic Management simulation tool.

The main goal of this project consists in demonstrate how AI-driven route optimization, conflict resolution and dynamic decision-making can enhance the safety, efficiency, and scalability of UAM operations.

To achieve this main goal, other specific objectives and goals of this project are:

1. **To give a solid oversight of current ATM systems and UAM environment:** by analyzing and defining all UAM environment concepts and ATM current structure and limitations.
2. **To investigate the potential of AI in Unmanned Traffic Management (UTM):** by analyzing ATM challenges and limitations and researching AI techniques to automatize ATM processes and face current ATM challenges.
3. **To design and implement AI-driven conflict resolution and dynamic decisionmaking model with the BlueSky Simulation tool:** by developing a Reinforcement Learning model which autonomously makes decisions to maximize rewards in an urban airspace 2D environment.
4. **To enhance the efficiency, scalability and safety of operations by AI-driven approaches:** by evaluating system performance and identifying key limitations and improvement areas. Key metric parameters to measure success will rely, for example, on conflict reduction, fuel efficiency and average reward.

1.4 CONTENT & METHODOLOGY

With the purpose of ensuring the achievement of the described project objectives, the following content and methodology will be followed.

1. **INTRODUCTION**

To introduce the project topic and purpose by briefly describing Urban Air Mobility context and situation and which challenges Air Traffic Management are facing to control and manage this increase in Air Traffic density.

In addition, this section will include my motivation to develop and carry out the project and the main objectives and goals of the research, as well as the project content and methodology specification, the chronogram to follow and the key risk analysis of the project.

LITERATURE REVIEW

In this part, the project aims to investigate and cover all key concepts of Urban Air Mobility and Air Traffic Management, as well as AI techniques in ATM and how can AI be used for optimizing and conflict-resolution processes in UTM. The sections will be:

2. **UAM concept and existing ATM systems:** to define and explain UAM concepts and ATM current structure and challenges.
3. **AI techniques in Air Traffic Management:** to compare traditional ATM with AI-driven ATM systems and to describe Machine Learning and Reinforcement Learning applications in ATM systems. In addition, U-Space and Next Generation programs are described and included in this section. Finally, Unmanned Traffic Management will be introduced.
4. **Reinforcement Learning for Air Traffic Management:** to introduce Reinforcement Learning and its main applications to ATM and to compare the different RL algorithms in order to explain the chosen algorithm for the project.

RESEARCH METHODOLOGY

In this part, the project aims to research and design the AI model to implement. Firstly, it will provide the research approach definition in order to clarify the tool-selection reasons. Then it will detail how the AI model is designed and implemented by using Reinforcement Learning with the PPO algorithm and will detail all the experimental setups and procedures followed to run them. Finally, it will define the performance evaluation methods by describing how to measure the success in all the experiments carried out. In addition,

5. **AI model design:** to define the research methodology implemented and to explain the reason of using AI simulation tools for this project research. This will include how the AI model is designed, trained and implemented in an ATM environment, how BlueSky is configurated and how the experiments are conducted. It will aso be described how success will be measured in the experiments in terms of safety, efficiency and scalability.

6. **Project impact:** to describe the project repercussions and social compromise in terms of social, economic and environmental impacts and for gender perspectives.

THESIS CONCLUSIONS

7. CONCLUSIONS AND FUTURE WORK

The last part of the project will review all the results about AI integration into ATM management and about the use of Reinforcement Learning for air traffic conflictresolution. Finally, it will also review its main challenges and limitations as well as the future applications in ATM environments.

1.5 CHRONOGRAM

Considering that the project will take approximately 19 weeks (from 10th of February to 26th of June), the planning schedule will be:

Task	Start date	End date
1. Introduction	**10/02/2025**	**16/02/2025**
1.1 Project description	10/02/2025	11/02/2025
1.2 Motivation	11/02/2025	12/02/2025
1.3 Goals and objectives	12/02/2025	13/02/2025
1.4 Content and methodology	13/02/2025	14/02/2025
1.5 Chronogram	14/02/2025	15/02/2025
1.6 Risks	15/02/2025	16/02/2025
2. UAM concepts and existing ATM systems	**16/02/2025**	**10/03/2025**
2.1 Urban Air Mobility concept	16/02/2025	20/02/2025
2.2 The need for UAM	20/02/2025	25/02/2025
2.3 Introduction to ATM	25/02/2025	02/03/2025
2.4 Challenges in current ATM systems	02/03/2025	10/03/2025
3 AI techniques in ATM	**10/03/2025**	**01/04/2025**
3.1 Traditional ATM challenges and limitations	10/03/2025	15/03/2025
3.2 Machine Learning in ATM	15/03/2025	20/03/2025
3.3 Deep Learning in ATM	20/03/2025	23/03/2025
3.3 Multi-Agent systems for ATC	23/03/2025	27/03/2025
3.5 Unmanned Traffic Management	27/03/2025	30/03/2025
3.6 UTM system architecture	30/03/2025	01/04/2025
4 Reinforcement Learning in ATM	**01/04/2025**	**12/04/2025**
4.1 Introduction and concepts	01/04/2025	05/04/2025
4.2 Key components	05/04/2025	07/04/2025
4.3 Markov Decission Processes	07/04/2025	09/04/2025
4.4 RL approaches	09/04/2025	12/04/2025
5. AI model design	**12/04/2025**	**20/05/2025**
5.1 Horizontal Model	12/04/2025	25/04/2025

5.2 Improved Horizontal Model	25/04/2025	15/05/2025
5.3 Comparison with other algorithms	15/05/2025	20/05/2025
6. Project implementation impacts	**20/05/2025**	**28/05/2025**
6.1 Social impacts	20/05/2025	22/05/2025
6.2 Economic impacts	22/05/2025	24/05/2025
6.3 Environmental impacts	24/05/2025	26/05/2025
6.4 Gender perspective	26/05/2025	28/05/2025
7. Conclusions and future work	**28/05/2025**	**05/06/2025**
Revision	**05/06/2025**	**26/06/2025**
Total TFG	**10/02/2025**	**26/06/2025**

1.6 RISKS

There are some potential risks to consider for this project with their adequate mitigation strategies, such as:

- **Data quality and fiability**: it can be used diverse data sources and references as well as data validation techniques.
- **AI Model performance**: it can be experimented with different parameters with a continuous monitoring and analyzing.
- **Simulator limitations**: the scalability can be improved by increasing air traffic density and use real data to reduce the simulation volume.
- **Computational complexity:** simulation complexity can be reduced and cloud computing resources to learn can be found.

2. UAM CONCEPT AND EXISTING ATM SYSTEMS

2.1 URBAN AIR MOBILITY

Urban Air Mobility (UAM) is the use of electrical Vertical Take-Off and Landing (eVtol) aircraft in urban areas for passengers and cargo transportation, piloted or remotely controlled, including the required and necessary infrastructure and ground handling services to ensure a complete safety and security in all the flight operations and activities. By implementing UAM, there is also an improvement in the urban environment because of the reduction of greenhouse emissions and noiseless policies, as the aerial vehicles are planned to be electric and work as efficiently as possible with the new electric propulsion system and take-off and landing capabilities.

The implementation of this new air transportation system requires high investments in vehicle maintenance and infrastructure development and proper planning and optimization of traffic operation and control, and it produces new concepts of terminal designs, runways and landing surfaces and innovative navigation aids to provide an absolute safety performance. The future high demand for Urban Air Transport requires the building of new ground services infrastructure to ensure a safe and efficient activity.

These new infrastructure concepts are expected to be:

- Vertipads: will be the designated areas for UAVs electrical Vertical Take-Off and Landings (eVTOLs).
- Vertiports: a building that will be composed of several Vertipads, passenger and cargo required handling services, charging stations, basic maintenance facilities and an ATM system.
- Vertihubs: a larger facility working as central hub for several Vertiports. It will have a large number of Vertipads, a terminal for passengers and it will provide UTM services (for strategic air traffic coordination) for UAV transiting between different vertihubs.

[1,2,3,4]

2.2 THE NEED FOR UAM

UAM is an emergent concept product of the real huge necessity to find an effective solution for the increasing urban traffic congestion. Nowadays, there is a high density of urban population and concentration in the big metropolitan areas, and if we also add the growth in the daily use of private transportation vehicles, it is common to observe multiple and long traffic queues in peak hours. UAM concept was initially introduced in United States around 1947 with the use of helicopters for passengers and cargo transportation mainly in Los Angeles and New York city, and the idea has been developed and improved continuously producing significant progresses in the design of Urban Air Vehicles, named as electrical Vertical Take-Off and Landing (eVTOL) aircrafts. Currently, some Aviation Companies are planning to use UAM for cargo, goods and passenger transportation. In addition, many manufacturers such as Airbus, Boeing, Sikorsky or Lilium were the first ones to start designing and developing electric UAVs with the purpose of start introducing them imminently, and they were lately followed by Uber, Rolls-Royce and Toyota. Collectively, they are developing different eVTOL aircraft design concepts varying some characteristics, for example the seat number, speed and distance, with the main objective of ensuring a safe, secure and efficient UAM. The large design variety and the increasing numbers of eVTOL manufacturers are forcing government institutions and aviation agencies to issue regulations, standards and safety procedures for UAM (in terms of weight, speed, automation, navigation, communication, surveillance...) with the purpose of regulate it and make this incoming technological revolution a safe and coordinated environment.

One limitation to consider about the actual eVTOL vehicles, is the fact that at this moment they have a limited range and they cannot cover long distances, as the battery can only support flights of a few hundred kilometers. To offer longer distances, it is necessary to develop and implement new power resources and battery technologies, but for the initial implementation of the environmentit is not a short-term important obstacle. [2,3,4]

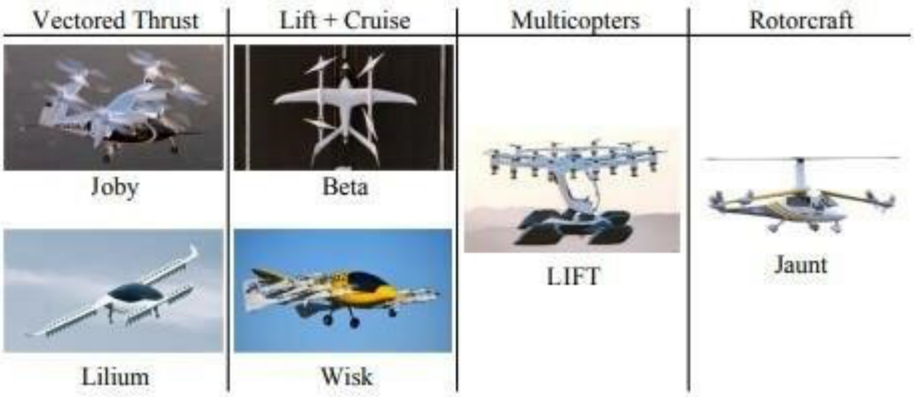


Figure 1. Different UAM aircraft designs [5].

2.3 INTRODUCTION TO ATM

Air Traffic Management is the dynamic and integrated management of air traffic and airspace to ensure a safe and efficient aircraft traffic flow in all flight operation phases. It includes ground-based functions such as Air Traffic services, airspace management and Air Traffic flow management. These three main services consist in:

- Air Traffic Service (ATS): ensures a safe and ordered air traffic flow and is composed by:
  - Advisory Service: provides flight recommendations to aircraft in noncontrolled zone.
  - Flight Information Service (FIS)

    - Alerting Service: notifies aircraft about potential dangers.
    - Air Traffic control Service (ATC), who provides the required information to flight crews. Its main function is to prevent aircraft collisions by ensuring proper separation standards between aircraft and keeping direct and continuous communication with operators during the entire flight phases. ATC is divided into:
        - Area Control service (ACC) to manage air traffic.
        - Approach service (APP) to handle aircraft arriving or departing from an airport.
        - Aerodrome Control service (ATWR) to manage aircraft on the ground.
- Air Traffic Flow Management (ATFM): optimizes and regulates aircraft flow in an efficient manner avoiding traffic congestion, reducing delays and improving airspace and airports efficiency. Its main function is to keep a balance with capacity and demand, ensuring that the capacity limits of ATC, airspace and runways are not exceeded at any time. To ensure these safety airspace requirements, ATFM can manage congestion allocating departure slots to aircraft and applying flow restrictions to control the aircraft density in a certain airspace.
- Air Space Management (ASM): manages airspace efficiently to satisfy its main users. It involves the way airspace is designed and structured through routes, areas… in order to provide ATS and ensure a safe an efficient air traffic.

Despite these clear defined and structured ATM frameworks, the rapid growth of UAM activity supposes new challenges that traditional ATM systems are not still prepared to face, and the implementation of AI systems is urgently necessary and essential to ensure an actual and future safe airspace traffic [6, 7, 8].

3. AI TECHNIQUES IN AIR TRAFFIC MANAGEMENT

3.1 TRADITIONAL ATM CHALLENGES AND LIMITATIONS

As ATM operatives were designed to face low volume of air traffic activity, they are now having big struggles to face and manage all this global air traffic that does not stop increasing its activity. With simple and resolutive radar tracking and voice communication systems, it was enough to manage and control the original airspace traffic flow, but currently with the emergence and irruption of eVTOLs, drones and the high density of flight operations, the traditional ATM systems are becoming obsolete and the need to find digital and automating solutions is gaining weight and urgency in order to ensure safe and efficient airspace traffic. At the moment, ATM operations are depending on human intervention and manual processes, and it causes huge inefficiency, safety insurance problems and large traffic congestion.

Firstly, the airspace limitation and the poor ATM resources cannot face and manage the increasing volume of flight operations that we are witnessing year to year, as especially after COVID-19 pandemic, air traffic density has grown with giant steps overloading the actual airspace capacity. In addition, different ATM solutions are being implemented in different global areas, such as Next Gen in USA, SESAR in Europe, traditional and manual ATM in China, and it creates interoperability issues and diversification of ATM operations.

On the other hand, the original ATM was designed and prepared to manage air traffic of manned vehicles, and the emergence of unmanned aircraft such as drones or eVTOLs is generating problems to integrate them into the current system. UAM new vehicles have different performance and activity requirements, so the implementation of AI-driven ATM systems in order to handle UAVs and eVTOLs traffic efficiently and safely is absolutely required. Besides this, the current ATM system works and relies on manual processes, so the digital and AI digitalization and implementation transition will be slow and complex as it requires large coordination and development work to ensure an effective ATM transformation.

In summary, current ATM limitations to manage airspace and UAM operations are:

1. Scalability and traffic volume limitation
2. Delay in decision-making
3. Conflict-resolution issues
4. Lack of UAV integration

To address these ATM limitations, some AI and digitalization solutions that are being considerated to implement could be:

- Enhancing and improving human-machine collaboration, as AI systems could assume decision-making, optimized flight trajectories and conflict-resolution responsibilities under human supervision, reducing human ATC workload.

- Integrating avionics and ATM systems, in order to ensure real-time coordination between both agents.
- ATFM optimization by AI-driven systems in functions such as air traffic density prediction and congestion detection to optimize air traffic flow efficiency.
- Implementing AI-driven collision avoidance and dynamic separation adjustments at real-time to improve airspace utilization and traffic efficiency.
- Integrating UAV operations into controlled space in order to manage and control the high-density volume of UAV traffic.

By implementing these AI-driven solutions, UAM would be managed mainly by the new air traffic control concept which would incorporate and implement all these technologic improvements driven by AI algorithms methods. This concept is known as Unmanned Traffic Management (UTM). With the ATM-X project, NASA aims to develop and implement AI models to optimize the National Airspace System (NAS) with an UTM system design.

[7, 8, 9]

3.2 MACHINE LEARNING (ML) IN ATM

If we look at society's historic evolution, we have always tried to simplify our tasks and processes by developing technologies that make our life easier. In this technological and digital era, the management and interpretation of large data sets required big amounts of time and resources, so by using the current AI capabilities is possible to automatize key processes and analyze large datasets to identify patterns and to make quick decisions in an optimized and efficient way.

As an AI subset, Machine Learning algorithms have significantly improved these processes optimization. In Air Traffic Management, Machine Learning is largely useful to analyze big amounts of air traffic data enhancing airspace efficiency, predicting traffic congestion and optimizing flight routes. But what is Machine Learning?

Machine Learning (ML) is a branch of AI that enables computers to learn from large data sets by identifying patterns and making decisions without human intervention. ML is basically used to teach machines how to handle data in an efficient way. According to the computer scientist Arthur Samuel, “Machine Learning is the computer’s ability to learn without being explicitly programmed”[11]. In ATM, ML can be used and applied to automate some key processes such as decision-making, airspace efficiency, traffic prediction and to detect air traffic anomalies. Depending on different factors, including the nature of the problem, the number of variables or the data availability, it is better to choose a certain type of ML algorithm. The main Machine Learning algorithm types are:

- **Supervised Learning (SL):** is a ML type where the model is trained with labeled datasets. In SL, the machine learns a set of input-output pairs where inputs are related with their desired outputs. Input dataset is divided into training set (to train the model) and testing set (to evaluate its performance). Main SL examples are Decision Tree, Support Vector Machine (SVM) are Naïve Bayes algorithm.

  In ATM, SL is applied to avoid congestion in real-time by using historic weather and air traffic data and for conflict prediction by identifying non normal flight patterns.

- **Unsupervised Learning (UL):** unlikely SL, UL is not trained with labeled datasets, as UL algorithms analyze unstructured data to learn few features and use them to identify patterns and similarities. Main UL examples are Principal Component Analysis and K-means Clustering.

  In ATM, UL is applied to identify flight patterns and unusual aircraft behaviors that can mean safety risks (air traffic anomalies); and for route optimizing by analyzing large weather and traffic data sets at real-time.
- **Reinforcement Learning (RL)**: is a ML paradigm based on how an agent interacts with an environment to find the optimal strategy. In RL, the agent takes actions and receives feedbacks in form of reward or penalty, and it modifies its policy with the objective of maximizing a cumulative reward over time, learning through trial and error.

  In ATM, RL is applied for route optimization and for conflict avoidance in a high density airspace.
- **Multitask Learning:** is an algorithm used to solve multiple tasks at the same time by identifying similarities between different tasks. It improves the efficiency in learning by sharing the knowledge extracted from all tasks.

  In ATM, Multitask Learning is applied for weather forecasting and for UAVs traffic management.

NASA's research applies ML models for the Runway Configuration Management (RCM) and for optimizing the airspace flow. For RCM, Machine Learning models are used for predicting runway demand and optimize its usage based on weather conditions and other operational constraints, helping ATC to select the optimal runway configuration under changing conditions and to adapt runway assignments dynamically with RL.

[10, 11, 12]

For this project, Reinforcement Learning is the most suitable approach due to its ability to optimize complex and dynamic systems without the need for predefining rules and datasets. A larger RL method definition will be described in the following section.

3.3 DEEP LEARNING IN ATM

Deep Learning (DL) algorithms are a class of artificial neural networks (ANNs) with multiple layers and parameters. They are designed to analyze and process large datasets efficiently to extract complex features, identify data patterns and make predictions with high precision. To extract and transform features, DL uses series of interconnected layers. Main DL architectures are:

- **Deep-Neural Network (DNN)**: formed by multiple processing units, known as neurons, ordered in different layers. These layers can be input layers (to receive the data), hidden layers (to process the data through weighted connections and activation functions) and output layers (for the final result or prediction). In DNN, a neuron produces an output and sends it to the following layer, and the network is trained to optimize the cumulative weights with different algorithms.

  In ATM, DNN can be used to predict flight departures and potential delays through air traffic and weather data. It can also detect congested areas and analyze airport operations with high quality precision, as most of delays depend on each airport organization and operations. Finally, DNN can be used to autonomously determine an aircraft approach trajectory thirty minutes before landing and to carry a real-time aircraft tracking.
- **Convolutional-Neural Network (CNN):** is used to process data in grid-like topology (for example time series and image data). CNN is formed by 3 layers: convolutional layers to extract key features data, pooling layers to reduce resolution of features by retaining important information and fully-connected layers to produce class scores and to make the final decision.

In ATM, CNN can be used to predict aircraft trajectories considering weather impacts and to realize aircraft maintenance image-based inspections to detect aircrafts defects. In addition, CNN can be used to process airspace traffic maps to analyze congested areas and predict ATC workload.

- **Recurrent-Neural Network (RNN):** used to detect patterns in data sequences. RNN features are feedback connection and memory to flow in a loop and temporal processing. It means that RNN can retain and use past information for predictions and data processing.

  In ATM, RNN can be used for speech recognition for ATC by processing their communications and transcripting them to text. It can also be used to detect congestion patterns in airspace traffic and to identify potential delays through large air traffic and weather forecasts datasets.

- **Generative Adversarial Network (GAN):** an algorithm based on the opposition between generation and a discriminator, as GAN consists in a generator that constantly produces synthetic data while a discriminator tries to distinguish if it is real or fake. When data cannot be classified as fake, the model produces realistic data.

  In ATM, GAN can be used to detect aircraft anomalies and suspicious behaviors. Besides this, GAN can generate synthetic air traffic scenarios to train AI models for ATM simulations by simulating realistic flight trajectories influenced by realistic weather conditions based on historical flight data.

- **Autoencoders**: are used to encode inputs in compressed representations and then decode them back aiming to produce an output identical to the input.

  In ATM, Autoencoders can be used to detect flight anomalies and to carry real-time monitoring of ATM systems by extracting meaningful features from large datasets. It can also predict airspace congestion areas by identifying common flight routes in large flight datasets.

  [12, 13, 14, 15]

3.4 MULTI-AGENT SYSTEMS FOR ATC

The increasing complexity of managing urban air traffic, especially due to the emergence of UAM at the scene, supposes the huge necessity of adaptive, flexible and decentralized traffic management solutions. Multi-Agent Systems provide a solid architecture for ATM capable of affording all of these solutions by implementing a distributed control, autonomous decision-making capabilities and coordination tasks for multiple aerial vehicles and infrastructure nodes at the same time, guaranteeing an efficient and safe urban air traffic for UAVs. But what is a Multi-Agent System?

Multi-Agent Systems (MAS) consist of multiple autonomous agents capable of perceiving their environment and making decisions while interacting with other agents at all times. These systems are often set with RL algorithms to learn optimal behaviors for determined scenarios and situations. In UAM, these agents would be the different UAVs transiting urban airspace, ground control units, vertiports... By implementing MAS, ATC decisionmaking tasks workload could be decentralized with an autonomous real-time coordination of UAVs in dense urban airspace. Main MAS application in UAM scenarios are:

- **Conflict-detection and resolution**: by predicting potential UAVs trajectories and adjusting heading, speed or altitude in order to maintain safe separation and to avoid collision conflicts with other agents.
- **Coordinated routing and scheduling**: by designing and scheduling optimal flight paths and time slots for UAVs departures and landings, taking in consideration routes fuel optimization and vertiports availability.
- **Dynamic Airspace Reconfiguration (DAR)**: real-time airspace segmentation and reconfiguration in a dynamic environment ensuring safe separations and managing flow control.
- **Emergency handling and rerouting**: if an agent fails, agents around can detect abnormal behaviors and adjust their trajectory to avoid the failed vehicle.

[16, 17, 18]

Recent studies have implemented Multi-Agent Deep Reinforcement Learning approaches in realistic air traffic scenarios, enabling collaborative behaviors between UAVs while avoiding collisions and optimizing their routes. As a result, the model improved traditional single-agent or rule-based systems in terms of scalability, energy efficiency and conflict avoidance [19].

3.5 UNMANNED TRAFFIC MANAGEMENT

Unmanned Traffic Management (UTM) is a designed AI-driven ATM system with the purpose of facilitating the integration of UAV traffic into airspace, mainly at loweraltitude levels, with the clear objective of ensuring safety and security in UAM operations. UTM is based in the collaboration and cooperation with traditional ATM, and key areas where UTM focuses on are in the airspace design and organization, communication and technical infrastructure and standarization, and in stablishing air traffic regulations for all unmanned vehicles.

In summary, by integrating AI algorithms functions, key process improvements of UTM in comparison with traditional ATM systems are:

- **Scalability enhancement**: by digitalizing and automatizing key management processes such as decision-making, trajectory adjustments and collision avoidance systems, UTM gets scalability in comparison with the limited air traffic capacity affordable for traditional ATM.

- **Communication systems:** currently, when an analogue voice radio communication system is used by ATC, all pilots must be tuned at the same frequency in order to establish communication with the controller and communication can only be Aircraft to Ground (A2G) or Aircraft to Aircraft (A2A). Looking at the expected Air Traffic growth in the short term, this can be challenging and limiting.

  What UTM incorporates and offers is a digital information flow exchange on Cloud between all entities with the purpose of guaranteeing real-time communication, more area coverage and a safer network by using 5G and IoT connectivity. New features UTM incorporates are:

  - Command and Control: to have UAVs under constant control even if they modify the destination or trajectory.
  - Air-to-Air: to have UAV-to-manned and UAV-to-UAV communication for positional information exchanging and to incorporate a two-way radio communication with ATC .

- **Decision-making processes:** during its operation, a UAV or eVTOL may experience or face external hazards such as technical failures, weather conditions or human errors, which can lead to incidents with potential deadly consequences. That is the reason why decision-making processes are essential and crucial to respond to these emergencies.

If an emergency occurs, UAVs need to have the capability to alter their original flight plan and to respond by spending the less time possible. To ensure this happens, a Finite State Machine (FSM) has been implemented to improve UAVs decision-making response time, and a rational to predict risk factors and to assess in the decisionmaking processes has also been incorporated to all UAVs. With these features, decisions are triggered by the combination of three functions that ensures UAVs independence and reduce their communication dependence with UTM. These functions are risk-factor prediction, decision generator and trajectory generator.

Additionally, UAVs have the capacity to predict conflicts before they occur and to adjust flight plans in a dynamic way.

In conclusion, after this UTM improvements description, we can briefly extract these key process enhancements respect traditional ATM:

FACTOR	**Traditional ATM**	**AI-driven UTM**
Scalability	Limited capacity	Higher capacity
Decision-making	Manual ATC intervention	Automated AI intervention
Communication	Two-way radio frequence	Digitalized through 5G and IoT
UAV integration	Limited integration	High integration
Conflict prediction	Radar and human	AI algorithms

[9, 20, 21]

3.5.1 U-SPACE SESAR PROJECT

The U-Space is a European Union project that is being developed from 2017 which aims to create an Unmanned Traffic Management (UTM) which implements Artificial Intelligence algorithms and to implement it gradually. It is regulated by EASA and implemented by EU member states under Regulation (EU) 2021/664. Its concept involves a set of services and procedures defined with the purpose of guaranteeing a safe and efficient integration in airspace for UAS. By using AI, U-Space features a high level of digitalization and automated functions, and it is strictly regulated in order to ensure an appropriate and safe integration and coworking of Unmanned Aircraft Systems (UAS) with the regular manned aircraft. With U-Space, UAS are permitted to operate if they follow the defined procedures and are supported by the approved U-Space services. The main goals of U-Space are to coordinate Aviation Authorities with UAS operators and automatize their information and petitions transmission operations in order to reduce Aviation Authorities workload, as U-Space avoids ATC collapse in front of the increment in the drones circulation.

The development steps planned for the U-Space project are:

- U1 Phase (2019): introduction of initial services covering e-registration, eidentification and geofencing.
- U2 Phase (2021): initial services for UTM, including flight planning, flight approval, tracking and integration with traditional ATC.
- U3 Phase (2025): introduction of advanced services such as assistance for conflict detection and automated detect and avoid functionalities.
- U4 Phase (2030): U-Space full services, high level of automatization, connectivity and digitalization.

To ensure and provide effective and safe UAM operations in Airspace, U-Space is designing and defining different safety measures. Firstly, in the design of Airspace, USpace is setting the required levels of safety and risk assessments by defining airspace geographical limits and the list of mandatory services for UAS activity. Then, in controlled space, ATC will incorporate and implement an AI-driven reconfiguration of the airspace (DAR) that consists of temporarily modifying a defined U-Space to permit short-term changes in manned traffic demand, as ATC will be responsible to segregate and differentiate manned and unmanned aircraft. For uncontrolled space, surveillance tech will be implemented to effectively signal all unmanned vehicles, and manned aircraft must be made electronically visible to USSP with conspicuity systems to avoid traffic proximity. Besides this, USSPs certifications will be required for all systematic approaches and communication facilities and information exchanges will be key and mandatory for all UAS operations in airspace.

Finally, U-Space implementation requires the coordination of new actors in the airspace environment, and we can find:

- **UAS Operators**: pilots and operators who need to connect and be supported by a U-Space service provider.
- **U-Space Service Provider (USSP)**: they are responsible for providing and facilitating U-Space services to UAS operators during all operational phases. They are supported by CISP's.
- **Common Information Service Provider (CISP)**: they are responsible for providing and facilitating the exchange of common information from USSP, ANSP (Air Navigation Service Provider) or a Competent Authority with the UAS operators.

The U-Space services provided by USSP are the set of digital and automated services designed to ensure UAS safe, efficient and secure access to U-Space. These services are based on AI algorithms and are differentiated as mandatory or optional. The first ones are:

- **Network identification services**: for identifying and providing identification of UASs and their position in U-Space.
- **Geo-awareness service**: for providing operational conditions and airspace limitations, to give an overview of the flying environment to the UAS operators.
- **UAS flight Authorization Service**: for authorizing UAS operations ensuring they comply with airspace restrictions and that are not causing any conflict with other UAS operating in the same area.
- **Traffic Information Service**: for providing traffic information from other UAS and manned aircraft to any UAS.

Optional:

- ***Weather Information Service***: for providing real-time weather information.
- ***Compliance Monitoring Service***: for ensuring UAS operations are aligned with the defined regulations and requirements.

On the other hand, the common information services provided by CISPs consist in analyzing and providing static and dynamic data to enable the U-Space services implementation. These services are composed by:

- **Air Navigation Service Provider (ANSP)**: provide geographical information, limitation and performance requirements as solve static and dynamic constraints.
- **Air Traffic Service Provider (ATSP)**: for dynamic airspace reconfiguration (DAR). It provides Air Traffic Control (ATC), flight information (FIS), flight alert services...
- U-Space Service Provider terms and conditions of access.

[22, 23, 24, 25, 26]

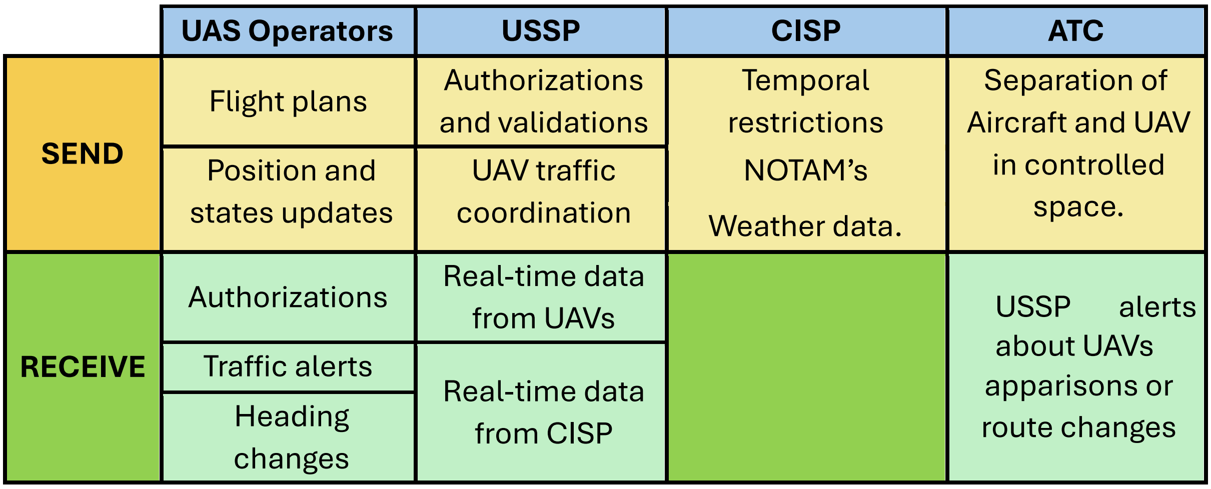
3.5.2 NEXT GENERATION PROGRAM

With the main purpose of modernizing and applying new technologies to improve the Airspace management, the Federal Aviation Administration (FAA) proposed the initiative Next Generation Air Transportation System (NextGen), and is currently working hard to develop new AI methods for ATC management, focusing on the air traffic capacity, the trajectory efficiency for eVTOLs, avoiding traffic congestion and defining a constraint management. Through NextGen, the FAA plans to reform the ATC infrastructure for communications, navigation, surveillance, automation and management to enhance operations safety, efficiency, flexibility and resiliency, as well as aims to improve infrastructure, introduce new AI technologies and procedures and enhance some safe and security aspects.

As a result of this program, FAA plans to develop an Urban Air Space and an Urban Traffic Management (UTM) system to provide an automated traffic management for remotely piloted UAV following and ensuring the proper urban separation and safety requirements. This automation system could provide an automatic self-separation of vehicles and a collision avoidance by programing processing algorithms for risk and threat detection and resolution, and could allow the optimate route with its adequate speed and altitude ensuring safety and reliability in all its operations. Besides this, the NextGen program aims to reduce aviation activity impact on environment, facilitating sustainable fuel uptake and aircraft and engines with lower fuel consumption and emissions. [26, 27]

3.6 UTM SYSTEM ARCHITECTURE

To better understand how the UTM system would work, it is important to know how its actors interact, and which tasks and responsibilities each one has to ensure a safe operation and effective information exchange. The functional architecture of an UTM system would result as the following table:



	UAS Operators	USSP	CISP	ATC
SEND	Flight plans	Authorizations and validations	Temporal restrictions NOTAM's Weather data.	Separation of Aircraft and UAV in controlled space.
	Position and states updates	UAV traffic coordination		
RECEIVE	Authorizations	Real-time data from UAVs		USSP alerts about UAVs apparisons or route changes
	Traffic alerts	Real-time data from CISP		
	Heading changes			

Figure 2. UTM system architecture

4. REINFORCEMENT LEARNING IN ATM

4.1 INTRODUCTION AND CONCEPTS

As mentioned earlier, Reinforcement Learning is an AI model where an agent learns optimal decision-making policies through trial and error from its direct interaction with an environment, as an agent executes an action in a determined state and this state is changed while the agent receives a numerical reward based on the quality of the action. This reward indicates if the agent's behavior has been “good” or “bad”. The goal of RL is to learn a policy that maps a set of states to actions that maximize the cumulative reward, and it tries to maximize the total cumulative reward on each timestep to achieve a goal. The key concept to understand RL is that an agent’s actions influence its future states, and that the agent has a reward notion to perceive if the action has been correct or not to learn the optimal decision-making processes by trial and error. Differing from SL and UL, RL does not rely on any pre-existing labels or exemplary supervisions, as the agent learns with direct interaction with the environment.

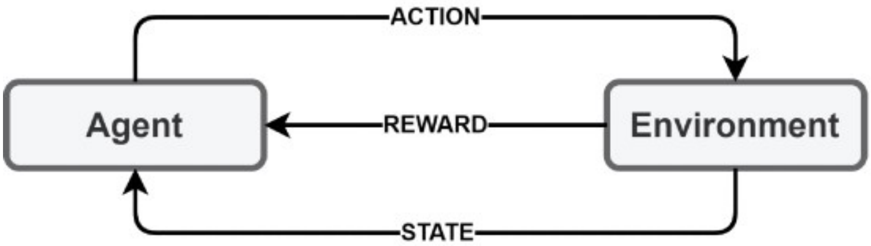


Figure 3. Reinforcement learning framework

For Air Traffic Management applications, RL is excellent because of its ability to optimize a complex and dynamic system such as the airspace traffic. Unlike traditional ATM, with RL a system learns from the continuous interaction with the environment, and it improves its decision-making over time. In addition, it can help to solve the UAVs airspace integration problems and the ATC limited scalability by optimizing their routes autonomously, reducing ATC workload. Finally, RL can be used to manage real-time conflict-resolution and collision-avoidance by predicting UAVs trajectories and ensuring safety separations between them at all times.

[28, 29, 30, 31, 32]

4.2 KEY COMPONENTS

The key components to understand of RL include:

- **Agent**: the decision-making entity that interacts with the environment by executing actions and receiving feedback depending on the policy satisfaction and learning the optimal decision-making processes by trial and error.
- **Environment**: the external system where the agent interacts and operates.
- **States**: current environment situations or scenarios before and after an agent action that is used by the agent to determine and decide its action.
- **Actions**: choices made by the agent that influence the environment's state and determine the feedback received.
- **Reward**: feedback received by the agent that provides the measure of success or failure of an agent's action, so after several attempts the agent learns the optimal action to take in each situation.
- **Policy**: the strategy followed by an agent for decision-making processes in order to maximize long-term cumulative rewards. It can be understood as a mapping of a state perception with the optimal action to take.
- **Value function**: estimates the expected cumulative reward of a state to determine the agent action feedback.

[30]

4.3 MARKOV DECISION PROCESSES

The Markov Decision Processes (MDPs) are the classical formalization of the sequential decision-making processes where actions influence on states and their future rewards. For RL, MDPs are fundamental to define the interaction between an agent and its environment and work as a framework for all RL algorithms.

On each step, known as state, an agent chooses an action that produces a reward and moves to the next state, having each action a probability to lead to a certain state. This is known as state transition probability. Every single MDP is defined as a tuple (S, A, P_a, R) composed by:

- S (State space): set of states in which the agent can be.
- A (Action space): set of available actions the agent can take.
- P_a (Transition probability): state transition matrix. Set of probabilities of transitioning to a new state with the current state *s* and action *a*.
- R (Reward function): a function that defines the reward received for taking action *a* in the state *s*.
- γ Discount factor: a parameter between 0 and 1 that balances the importance of future rewards against immediate ones. A higher γ leads to long-term rewards while a lower γ favors immediate rewards.

In addition, MDPs have a key characteristic named Markov Property that states that the future state or next step is only determined by the present state, not by the past states or actions. It is important to simplify the decision-making processes as only the current state is needed to determine the next action. Other key elements of MDPs are:

- Policy (π:):is the behavior of the agent that consists of a mapping between a state and each action probability. To determine the policy that maximizes cumulative reward we use a recursive relation for functions.
- Value function (V(s)): used to estimate if the state is good or bad by comparing the expected return from agent actions. Is a kind of measure of the expected cumulative reward the agent will receive if it starts from the state *s* following the policy π .
- Action-Value function: expected return of taking the action *a* in the state *s* following the policy π. [30]

4.4 RL APPROACHES

RL approaches can be classified into 3 main categories depending on the strategy followed by the algorithm to maximize the cumulative reward of the agent. The main RL approaches are: value-based methods and policy-based methods…

4.4.1 VALUE-BASED METHODS

Value-based methods are the ones focused on determining the optimal value function from every state and state-action mappings. These methods use the called Bellman equation to continuously improve the value estimation in an iterative way. Some valuebased algorithms are:

- **Q-Learning**: uses Bellman equation to continuously improve value estimation.
- **Deep Q-Networks (DQN)**: based on Q-Learning algorithms but including Deep Neural Networks to find the optimal value function.

Value-based methods are solid and have strong theoretical bases, but they are only effective in discrete spaces as they have problems working in large and continuous spaces such as autonomous driving or control tasks.

[33, 34]

4.4.2 POLICY-BASED METHODS

Are the methods focused on learning state-action mappings without the necessity of estimating a value-function. These methods use policy functions to choose the best action for every state instead of value-functions. Some value-based algorithms are:

- **Policy Gradient (PG)**: uses gradient ascents on expected rewards to optimize the policy function.
- **Proximal Policy Optimization (PPO)**: improves efficiency and stability of the policy function by restricting updates to avoid large policy changes.
- **Trust Region Policy Optimization (TRPO)**: ensures stable updates by constraining the divergence between old and new policies.

Differing from value-based methods, these algorithms work well in continuous and dynamic spaces, so they allow to carry a better exploration. Besides this, policy-based methods have a slower convergence in comparison to the value-based. [33]

For this project of implementing an AI-driven algorithm for ATM purposes, policy-based methods are more suitable than value-based ones as the airspace is a dynamic and continuous space. Instead of TRPO or PG, this project uses the PPO algorithm because of its robustness, learning stability and sample efficiency in simulation. These characteristics have a huge importance for ATM to prevent erratic behaviors, to have control over speed and heading, to prevent dangerous actions and to support multiple aircraft in airspace. [35]

4.4.3 HYBRID METHODS

In addition to the main RL method families, hybrid methods combine the advantages of policy-based and value based paradigms, as the agent makes actions according to its current policy while tries to maximize the value function reward. These hybrid methods are effective in complex and dynamic environments such as UAM, where decisionmaking agents must continuously adapt to changing conditions while optimizing short- and long-term objectives. Main hybrid methods are:

- Actor-Critic (AC): policy and value functions are simultaneously learned.
- Deep Deterministic Policy Gradient (DDPG): extends the AC method to continuous spaces, ideal to manage some UAV control tasks.
- Twin Delayed DDPG (TD3): improves DDPG by using two critic networks and delayed policy updates, to make it more robust and stable.
- Soft Actor-Critic (SAC): a recent algorithm that maximizes return and entropy, encouraging agent's exploration with robust policies under dynamic conditions.

4.4.4 PPO ALGORITHM

The Proximal Policy Optimization algorithm (PPO) is a policy-based method that restricts the magnitude of policy updates with the feature known as clipped objective function. This function protects the model of large policy changes that could destabilize learning.

Another key element of PPO is the known as advantage function, that provides a measure of how much better taking an action in a given state is in comparison to the expected value of following the current policy from that state. By this way, the agent learning is guided by prioritizing actions that lead to above-average outcomes. In addition, it also includes stochastic policies that promote the agent's exploration to allow him to test large variety of actions in training. It is positive to improve the agent's ability to adapt and operate in a dynamic and changing environment.

The PPO algorithm flow diagram would be like:

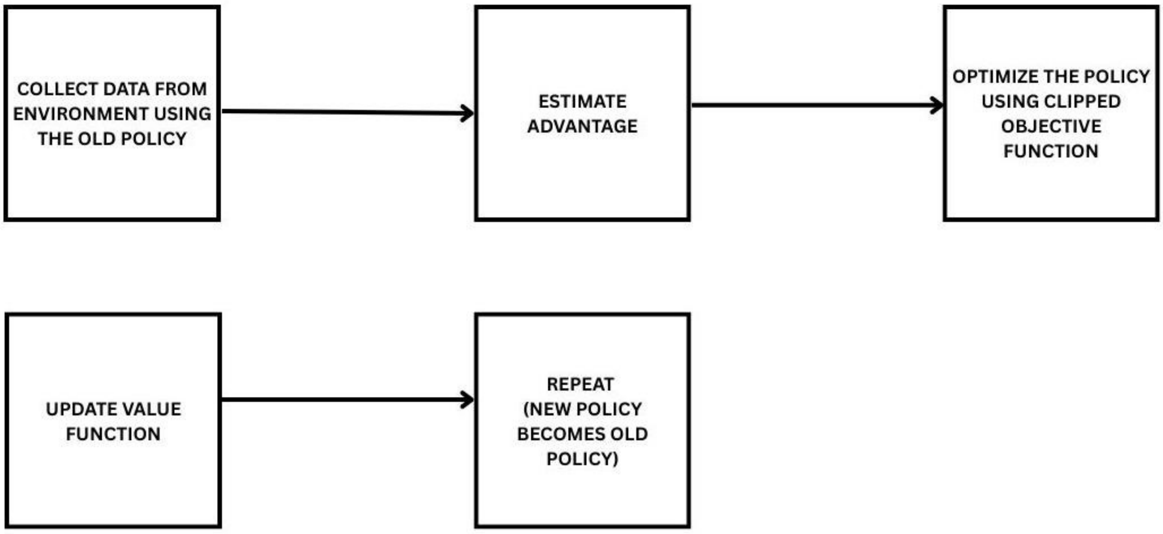

COLLECT DATA FROM ENVIRONMENT USING THE OLD POLICY
ESTIMATE ADVANTAGE
OPTIMIZE THE POLICY USING CLIPPED OBJECTIVE FUNCTION
UPDATE VALUE FUNCTION
REPEAT (NEW POLICY BECOMES OLD POLICY)

Figure 4. PPO algorithm flow diagram

As seen, PPO is a policy-based method designed to improve the stability and performance of RL algorithms by constraining the policy updates, and its flexibility allows the agent to operate in dynamic environments while exploring large variety of action sets, therefore it is perfectly suitable for Air Traffic Management purposes where airspace changes every second with constant aircraft conflicts.

[34, 35]

5. AI MODEL DESIGNS

As a background recap, there is an incoming growth in UAM with the implementation and launch of eVTOLs and other UAVs, and the installation of the required infrastructures for their operation. Urban airspace is expected to experience a high density of air traffic, leading to a congested flow of air vehicles. Air traffic controllers and all ATM processes were designed a long time ago to manage and control a limited number of aircraft at the same time, so with the emergence of drones and other eVTOLs in urban airspace, UAM will exceed ATM capacity and tower controllers will result large overloaded. Some initiatives, such as Next Generation Program or Sesar JU U-Space, plan to design and integrate AI-algorithms to assume some ATC decision-making tasks and functions, reducing ATC workload. For example, AI could be applied to predict and avoid some air traffic conflicts like aircraft collisions by adjusting aircraft trajectories or to reduce airspace congestion by planning more efficient routes. This is known as Unmanned Air Traffic Management.

As a demonstration, this project aims to design, implement and evaluate a simulation environment in which an AI-driven agent autonomously manages aircraft trajectories in an urban area using Reinforcement Learning. Looking at the scenario of urban airspace, with several conflicts and changes, the chosen RL algorithm will be Proximal Policy Optimization because of its robustness and flexibility to work in dynamic environments such as UAM. By this way, the PPO agent will learn to control multiple agents resolving conflicts and making decisions in a 2D BlueSky Air Traffic simulator environment, as it provides a high realistic model of the airspace dynamic potentially scalable for UAM real scenarios. To directly interact with simulation aircraft, it will integrate an OpenAI Gym environment, which enables the agent to learn and make decisions in a simplified and realistic airspace.

In this section, the AI model design used in this project will be described and introduced. This model consists of a PPO agent interacting with a BlueSky environment, analyzing airspace states and then making actions to avoid conflicts, such as aircraft heading and altitude adjustments. As an RL algorithm, a reward function will be designed in order to guide the agent's decisions, ensuring airspace safety and efficiency. The key components to understand in the design are the model environment, the state space, the action space and the reward function description.

5.1 HORIZONTAL MODEL

To start with the design and implementation of an AI system capable of managing UAM using RL, training a PPO agent in a BlueSky Simulation Environment, the investigation initially trains the PPO agent in a 2D simplified airspace environment. It means that in this first environment, the PPO agent will only focus on avoiding drone collisions and to ensure safe separations between them while it moves to a determined waypoint in a dynamic scenario. Multiple drones “Dji M600” will be generated, and the PPO agent will need to make decisions about the drone heading to avoid collisions, maintain safe separations and resolve conflicts efficiently.

The objectives of this model are to demonstrate the feasibility of AI-driven algorithms for autonomous conflict-resolution, to show how an agent can navigate toward a goal maintaining a safe separation in all the process and to prove the idea of using AI for UAVs traffic management.

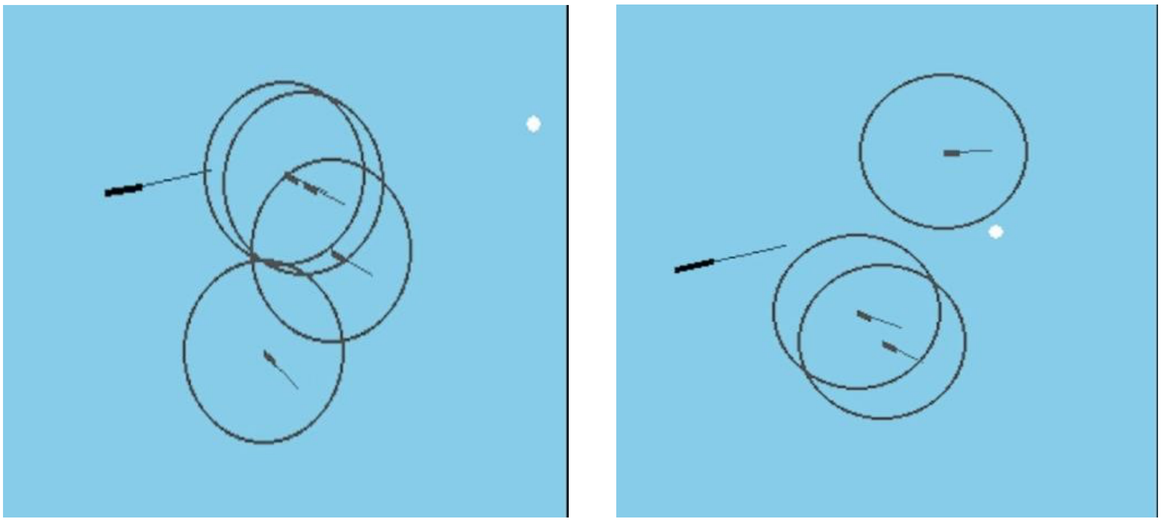
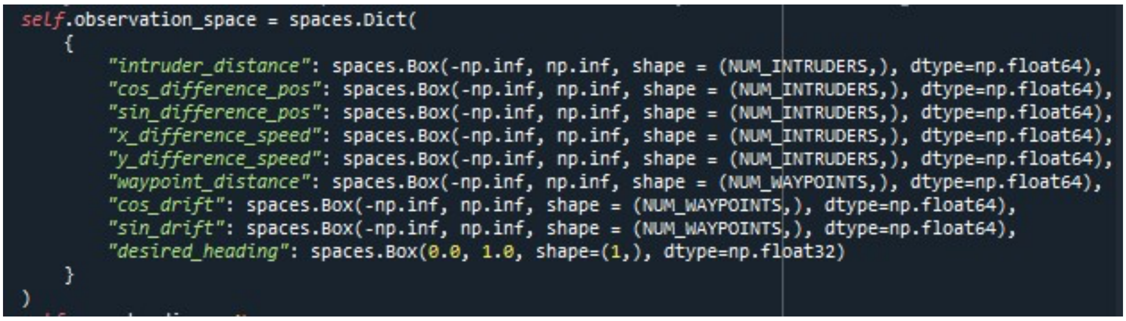


Figure 5. Visual representation of the environment

5.1.1 ENVIRONMENT INTRODUCTION

The environment for Horizontal Conflict Resolution has been taken from GitHub repository detailed in the reference section [37]. As an introduction, it is important to understand that the environment provides the PPO agent with:

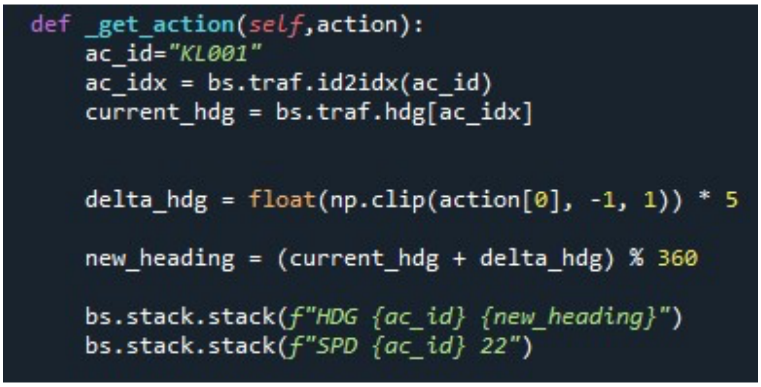
- **State Space**: the PPO agent receives observation states about the relative positions of the intruders and the separation distance with them, their relative velocities and headings and the heading alignment and distance to waypoint.



```
self.observation_space = spaces.Dict(
    {
        "intruder_distance": spaces.Box(-np.inf, np.inf, shape = (NUM_INTRUDERS,), dtype=np.float64),
        "cos_difference_pos": spaces.Box(-np.inf, np.inf, shape = (NUM_INTRUDERS,), dtype=np.float64),
        "sin_difference_pos": spaces.Box(-np.inf, np.inf, shape = (NUM_INTRUDERS,), dtype=np.float64),
        "x_difference_speed": spaces.Box(-np.inf, np.inf, shape = (NUM_INTRUDERS,), dtype=np.float64),
        "y_difference_speed": spaces.Box(-np.inf, np.inf, shape = (NUM_INTRUDERS,), dtype=np.float64),
        "waypoint_distance": spaces.Box(-np.inf, np.inf, shape = (NUM_WAYPOINTS,), dtype=np.float64),
        "cos_drift": spaces.Box(-np.inf, np.inf, shape = (NUM_WAYPOINTS,), dtype=np.float64),
        "sin_drift": spaces.Box(-np.inf, np.inf, shape = (NUM_WAYPOINTS,), dtype=np.float64),
        "desired_heading": spaces.Box(0.0, 1.0, shape=(1,), dtype=np.float32)
    }
)
```

Figure 6. Observation space

- **Action space**: in this environment, the PPO agent has a discrete action space based on heading adjustments up to 15 degrees.



```
def _get_action(self,action):
    ac_id="KL001"
    ac_idx = bs.traf.id2idx(ac_id)
    current_hdg = bs.traf.hdg[ac_idx]


    delta_hdg = float(np.clip(action[0], -1, 1)) * 5

    new_heading = (current_hdg + delta_hdg) % 360

    bs.stack.stack(f"HDG {ac_id} {new_heading}")
    bs.stack.stack(f"SPD {ac_id} 22")
```

Figure 7. Action space

- **Reward**: to guide a safe and efficient flight, the PPO agent receives a positive reward of +1 if it reaches the waypoint safe, a negative reward of –2 if a conflict is detected (separation violation) and a –0.01 penalty if it deviates from the desired path. It encourages and stimulates the agent to avoid intrudes and to reach the goal efficiently and ensures PPO agent's heading alignment.

The simulation runs for 6.666 episodes to train the agent. Every episode generates 5 intruders with random positions and velocities, a random waypoint and the PPO agent in a determined position. By this way, this approach ensures that the agent learns effective strategies rather than general or unique solutions. An episode finishes when the agent reaches the waypoint, a conflict occurs, or a maximum step count is reached (300 steps).

5.1.2 SCENARIO FOR DRONE-SPECIFIC DYNAMICS

To start designing the scenario, this project notes that this Horizontal Conflict Resolution environment is designed for Airbus A320 aircraft. As a result, various parameters have been adjusted to more accurately reflect the dynamics of M600 drones, allowing for more realistic conflict resolution in UAM scenarios.

Figure 8. Drone Dji M600

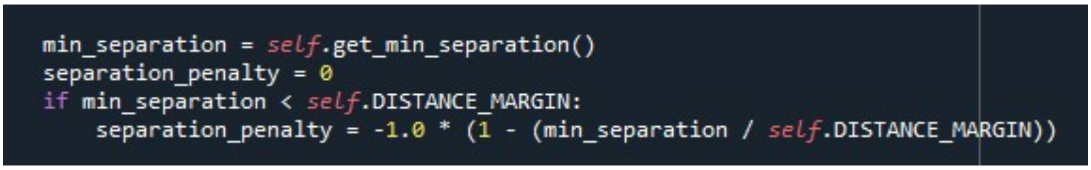
Drones M600 are professional hexacopters designed for stability, long-range operations and are mainly used for delivery services, infrastructure inspection, aerial photography and environmental monitoring. They can carry up to 6kg of payload and a maximum takeoff weight of 15,5kg, and their maximum durations are between 15 to 35 minutes (depending on payload). They can reach a maximum velocity of 18 m/s, which means around 65 km/h. Finally, they count with different sensor features, equipment, cameras, LiDAR and custom AI modules [38].

These parameters have been modified taking in consideration M600 drones configuration settings [38, 39]:

- **Distance margin (DISTANCE_MARGIN)**: this is the defined safe separation between agents. For this project, it has been adjusted from 5 to 0.02 for UAM contexts, as drones need less minimum separation distance than airplanes. If the minimum separation distance is violated, the PPO agent will receive a penalty, encouraging the agent to learn routes without commit collision conflicts.
- **Delta heading (D_HEADING)**: this is the maximum change of heading that the agent can apply in a single step. For this project it has been adjusted from 45 to 15 degrees, as drones can make larger heading changes than aircraft, to provide a more realistic UAM scenario transited by UAVs.
- **Aircraft speed (AC_SPD)**: this is the aircraft's maximum speed. For this project it has been adjusted from 150 to 18 knots as M600 drones have less cruise speed than aircraft and we need to provide a more realistic UAM scenario transited by UAVs.
- **Waypoint minimum distance (WAYPOINT_DISTANCE_MIN)**: this is the minimum distance from the agent's generation point at which the waypoint can be generated. It has been adjusted from 100 to 0.2.
- **Waypoint minimum distance (WAYPOINT_DISTANCE_MAX)**: this is the maximum distance from the agent's generation point at which the waypoint can be generated. It has been adjusted from 150 to 0.5.

Additionally, the reward function has also been modified in order to guide the PPO agent to the desired behavior ensuring safe and efficient flights. This modified reward function consists of:

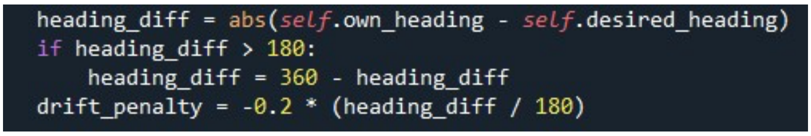
- Rewards:
  - ✓ Reach reward: the agent now receives a +1 reward if it reaches the objective waypoint and the episode finishes, encouraging the agent to fly towards the waypoint.
  - ✓ Alignment bonus: the agent receives an additional reward if the current heading is aligned with the waypoint heading, to guide the agent to an efficient flight trajectory.
- Penalties:
  - ✗ Conflict penalty: if the agent violates the minimum safe separation with an intruder, now it receives a –1 penalty instead of –2 to encourage the agent to learn in the initial steps. With the initial penalty, the agent may have been discouraged from exploring different routes as it was focused on avoiding intruders all the time.
  - ✗ Separation penalty: the agent receives a proportional penalty down to –1 depending on its safe separation violation. If it is at the border of the minimum separation with another drone it receives low penalties. If it is closer, it receives higher penalties. This penalty is useful to train the agent in maintaining safe separation progressively without the necessity of learning with only collision penalties.



```
min_separation = self.get_min_separation()
separation_penalty = 0
if min_separation < self.DISTANCE_MARGIN:
    separation_penalty = -1.0 * (1 - (min_separation / self.DISTANCE_MARGIN))
```

Figure 9. Separation penalty function

  - ✗ Drift penalty: the agent receives a penalty down to –1,5 based on the difference between its current heading and the optimal heading to the waypoint. It is useful to encourage the agent to have direct and efficient trajectories. The original drift penalty function was too basic, so I extended it improving the angular calculation precision and applying low and constant corrections at all times. The improved function is this one:



```
heading_diff = abs(self.own_heading - self.desired_heading)
if heading_diff > 180:
    heading_diff = 360 - heading_diff
drift_penalty = -0.2 * (heading_diff / 180)
```

Figure 10. Drift penalty function

On each step, the PPO agent calls the step function by itself, which calls the "get_action" function to perform a turn and to advance, and it also updates the observation space by adjusting agent's and intruders position. The episodes duration has been limited and controlled by setting a maximum of 300 timesteps per episode, avoiding the episode extending in excess without reaching the waypoint.

Every modification in the environment has been monitored and tested several times to maximize agent's performance in a high realistic scenario simulation. The design has been constantly adjusted in the reward function, conflict rates and waypoint reach capacity to ensure an effective training and evaluation.

5.1.3 PPO AGENT CONFIGURATION

The following configuration was used to train the agent with the PPO algorithm in the Horizontal Conflict Resolution environment, using the Stable Baseline 3 library [40] to get PPO hyperparameters information. The configuration is:

- **Policy**: MultiInputPolicy, to accept dictionary inputs for observation states.
- **Environment**: the "HorizontalCREnv", as seen before.
- **Learning rate**: 3e – 4, to control how fast the model is updated.
- **Gamma:** at 0.99, to favor a long-term planning.
- **Verbose:** at 1, to enable logging output during training.
- **Tensorboard_log**: to store training logs for visualization.
- Other hyperparameters, such as clip range (0.2) or number of steps, are set at a fix default value, as they are imported directly from Stable Baseline 3.

The training is set to take 2,000,000 environment steps and records all training statistics in a csv file, allowing to track learning curves, performance trends and policy stability over time.

5.1.4 EXPERIMENTAL PROCEDURE AND PERFORMANCE METRICS

The experimental procedure to train the PPO agent and evaluate the resulting model has been:

1. Training phase: the PPO agent has been trained by executing the Python script with the training setup in the Horizontal CR Environment, and the trained model has been saved.

2. Evaluation Phase: The trained model has been loaded and ran multiple times without training. All metrics such as number of conflicts, time to resolve and fuel efficiency have been collected.

Then, to evaluate the PPO agent performance looking at the collected information, these performance metrics have been used:

- Average reward: average reward per simulation.
- Total conflicts: number of aircraft conflicts, like safe separation violations.
- Conflict rate: proportion of episodes with aircraft conflicts.
- Average drift: average time with good heading alignment per episode.
- Waypoint reach rate: proportion of episodes where the agent reaches the waypoint.

5.1.4 TRAINING RESULTS

Once the training of the PPO agent has been completed, I proceed to evaluate the collected data, and I extracted these results

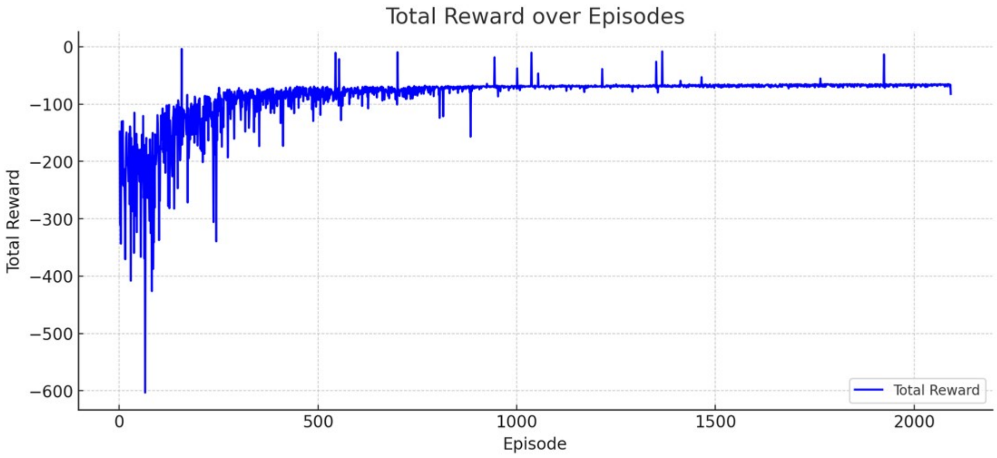


Figure 11. Average reward over episodes graph

As the above figure, we can see how the reward starts strongly negative and it increases over the episodes, stabilizing itself at approximately –70. This is not a good result as it shows that the agent has not found any effective policy during the training and it is actually far from it. The average reward per episode is –85.19, but since the middle of the training it stabilizes at –69 without visible improvements.

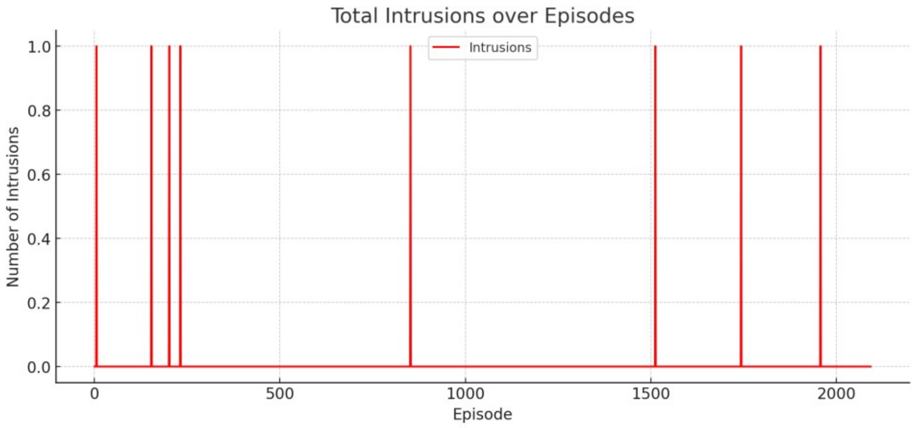


Figure 12. Total intrusions over episodes graph

As the above figure, we can see that no intrusions have occurred during training, only 8 conflicts during 2.000 timesteps, which means a 0.0038% of conflict rate. This is not a good result, as it means that, besides it has is a high-security policy, the agent never gets close to an intruder as probably it only focuses on escaping from them without any desire of reaching the waypoint.

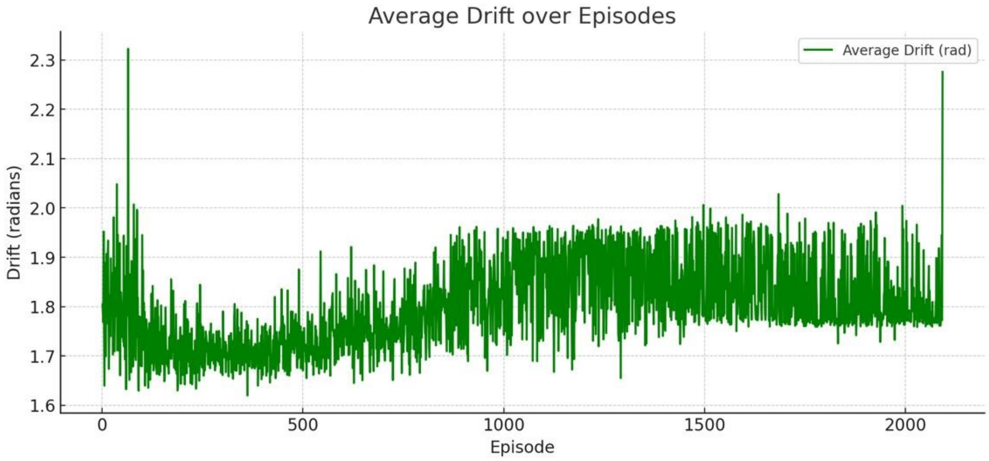
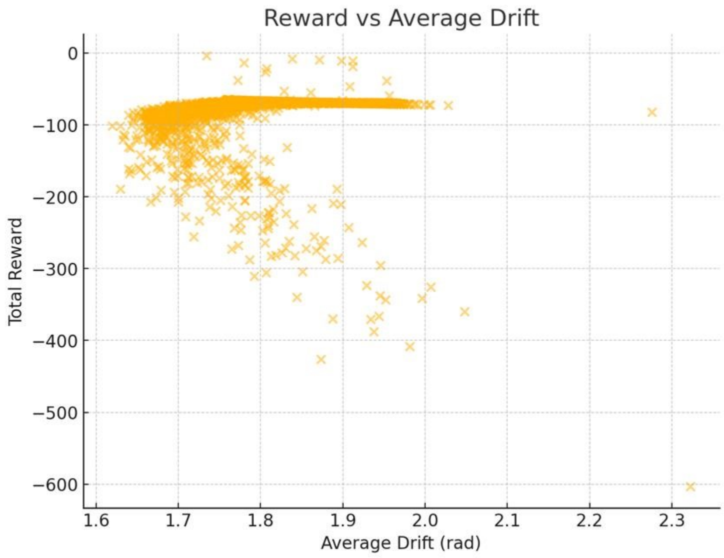


Figure 13. Average drift over episodes graph

During all training, we can see that the average drift remains high at all times at an average of 1.8, which means that the agent is not aligned with the waypoint heading, and no improvements have been noticed during all training.



As the following figure, which shows the relation of the reward in relation with the average drift, we can appreciate how the reward goes down as the drift increases, which means that the drift penalty is working, and the agent receives negative rewards when it deviates from the waypoint heading.

Figure 14. Reward vs average drift

In some training results, different drift values give the same total reward. This happens because the reward is not only based on drift, it is also composed by conflict penalties, smooth turns, heading alignment... So, even if the agent receives drift penalties, it can get the same reward because it did something else better, for example a smooth turn or staying properly aligned with the waypoint objective. That's why the reward can sometimes be the same even if the drift is different.

Finally, during the PPO training only 3 times the agent reached the waypoint, which means that the agent manages to avoid conflicts, but it never reaches the objective waypoint.

5.1.5 TRAINING CONCLUSIONS AND SUGGESTED IMPROVEMENTS

After evaluating the training results and key performance metrics extractions, I reach the conclusion that the agent learns how to avoid intrusions highly to avoid severe penalties, so it is clear to not proceed with the evaluation part as the training has been unsuccessful. Besides this, the average reward is too negative, which means that the agent never reaches the waypoint and that it is not well aligned with the waypoint heading, receiving a lot of penalties because of that. As a result, all episodes finish on time as the agent cannot manage to complete its objective as it is only focused on avoiding intrusion conflicts and the project demonstration is not satisfied. To revert this situation, there are some improvements and ideas that can lead to a better performance of the PPO agent:

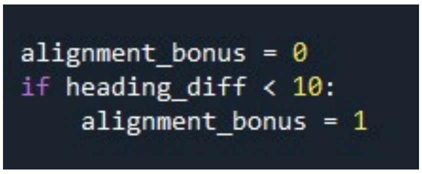
- The agent's reward can be increased for good heading alignment and for being close to the waypoint, to encourage the agent to move toward the objective.
- Drift deviations can be penalized stronger to readdress agent's trajectories.
- All episodes that finish because of time can also be penalized, to encourage the agent to desire the waypoint.

5.2 IMPROVED HORIZONTAL MODEL

5.2.1 IMPROVED REWARD FUNCTION

As mentioned in the evaluation conclusions of the first experiment, it is necessary to apply some changes in the reward function of the PPO agent, as it only focuses on avoiding conflicts without going to the waypoint. These changes are:

1. **A higher alignment bonus**: to encourage the agent to be aligned with the waypoint, the alignment bonus has been incremented from +0.4 to +1.



```
alignment_bonus = 0
if heading_diff < 10:
    alignment_bonus = 1
```

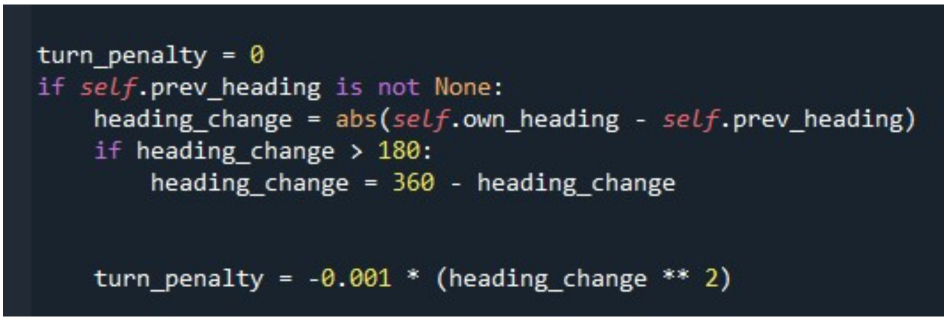
Figure 15. Alignment bonus improvement

2. **A stronger drift penalty**: as in the first experiment the agent was not properly heading aligned, the drift penalty has been increasd from –0.1 to –0.3 and in an augmentative form.
3. **A higher reward for reaching the waypoint**: I provide a bigger incentive to the agent for reaching the waypoint to encourage him to move toward it. To do it, the reach reward is increased from +1 to +15.
4. **Larger episode penalty:** the agent now receives a higher penalty per timestep, encouraging the agent to reach the waypoint in an efficient way.

5. **A stronger intrusion penalty:** as the agent was still having distance conflicts and to balance the penalties with the rewards increments, the intrusion penalty has been increased from -1 to –5.

5.2.2 HARD TURNS CONTROL

In addition, hard drifts control has been implemented with the purpose of encouraging the agent to avoid close turns, to make the environment even more realistic. To achieve this goal, a section has been added to the reward function:



```
turn_penalty = 0
if self.prev_heading is not None:
    heading_change = abs(self.own_heading - self.prev_heading)
    if heading_change > 180:
        heading_change = 360 - heading_change


    turn_penalty = -0.001 * (heading_change ** 2)
```

Figure 16. Hard turns control function

As seen in the above figure, a quadratic turn penalty is provided to the PPO agent in all turns it does. By this way, the penalty is higher for harder drifts, and the agent needs to learn how to avoid this penalty by realizing softer turns in its trajectories.

5.2.3 EVALUATION AND RESULTS

After implementing all the changes in the reward function and parameters definition, I proceed to train again the PPO agent, and these are the results of the agent's training in the improved environment:

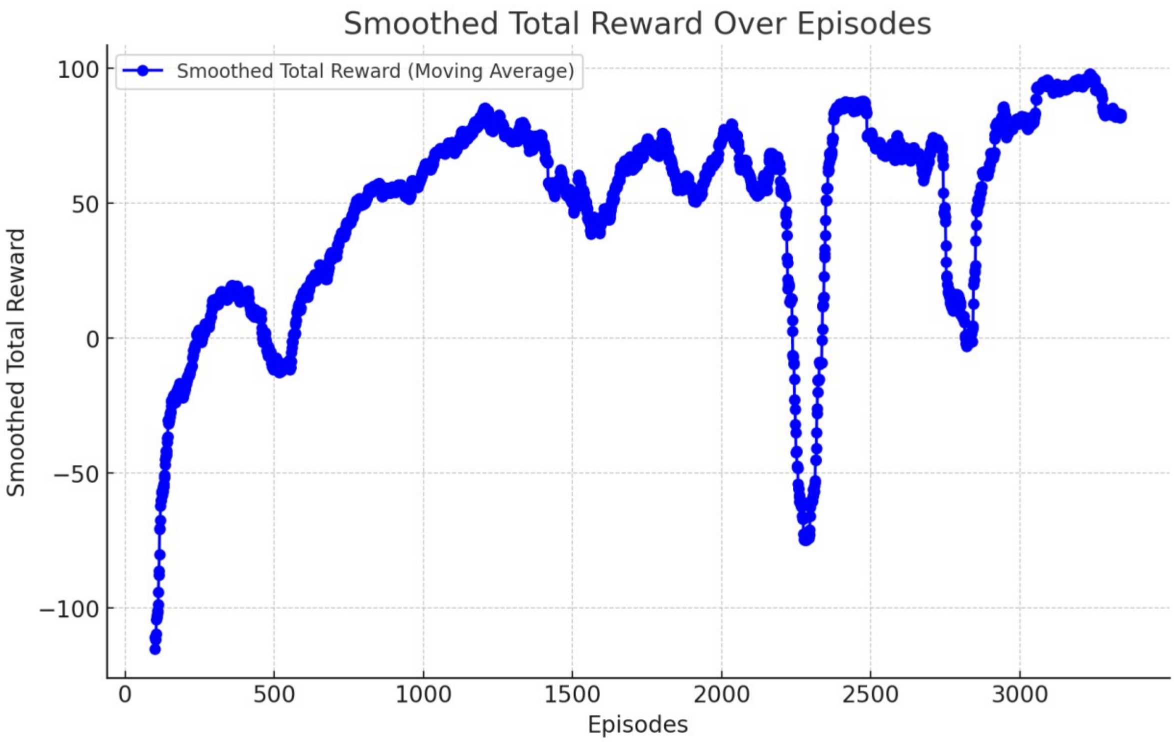


Figure 17. Improved average reward over episodes

As the above figure, we can see how the reward starts strongly negative with results below –100. As the training runs, the agent improves its performance reaching positives rewards before 500 episodes. As seen in the graphic, the rewards constantly increases over episodes, and the agent runs some explorations th

at produce visible downs in the curve. This is normal as the agent explores other options to get better rewards, and after carry this tests it implements it or returns to the previous knowledge recovering the reward value until it reaches approximately +100 rewards.

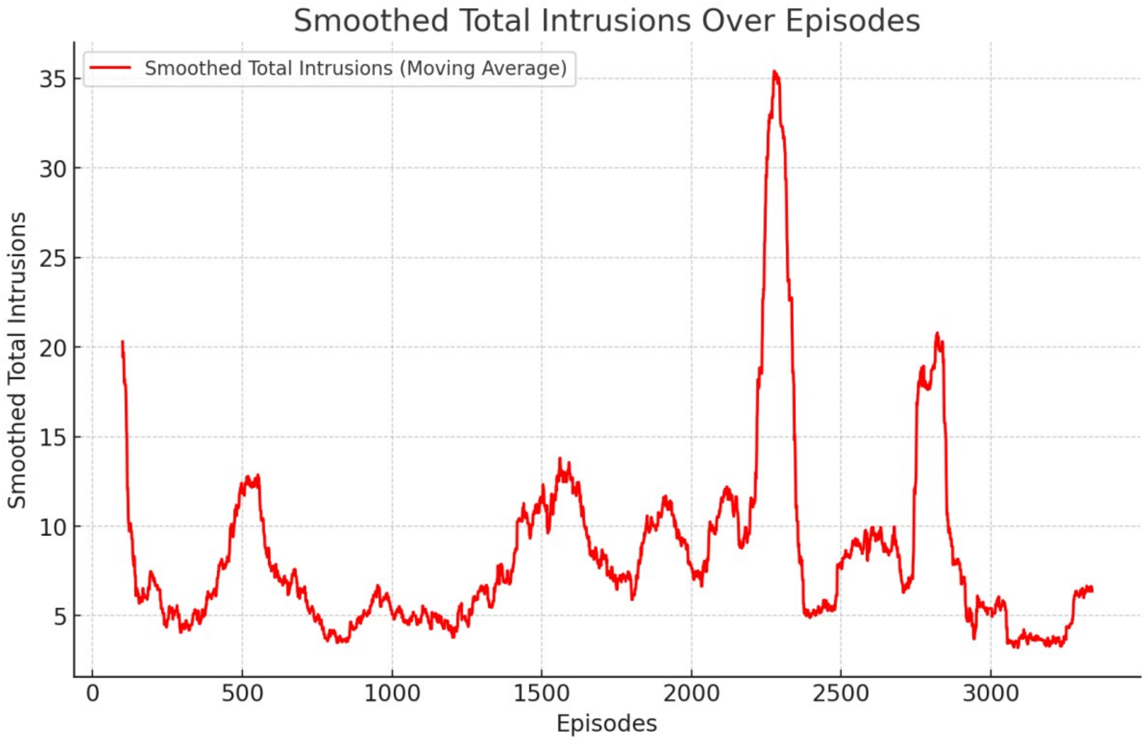


Figure 18. Improved total intrusions over episodes

As the above figure, we can see how the intrusion conflicts fluctuate among episodes. Starting with high intrusions number at the beginning of the training, the agent learns how to avoid them to increase its reward. As explained before in the reward curve, the agent carries some exploitations to find different ways that could lead to higher results, so the received reward goes down in these periods as well as the total number of intrusions, because of the implicit penalty produced when a conflict occurs. At the end of the training, we can see how the intrusion number is almost inexistent.

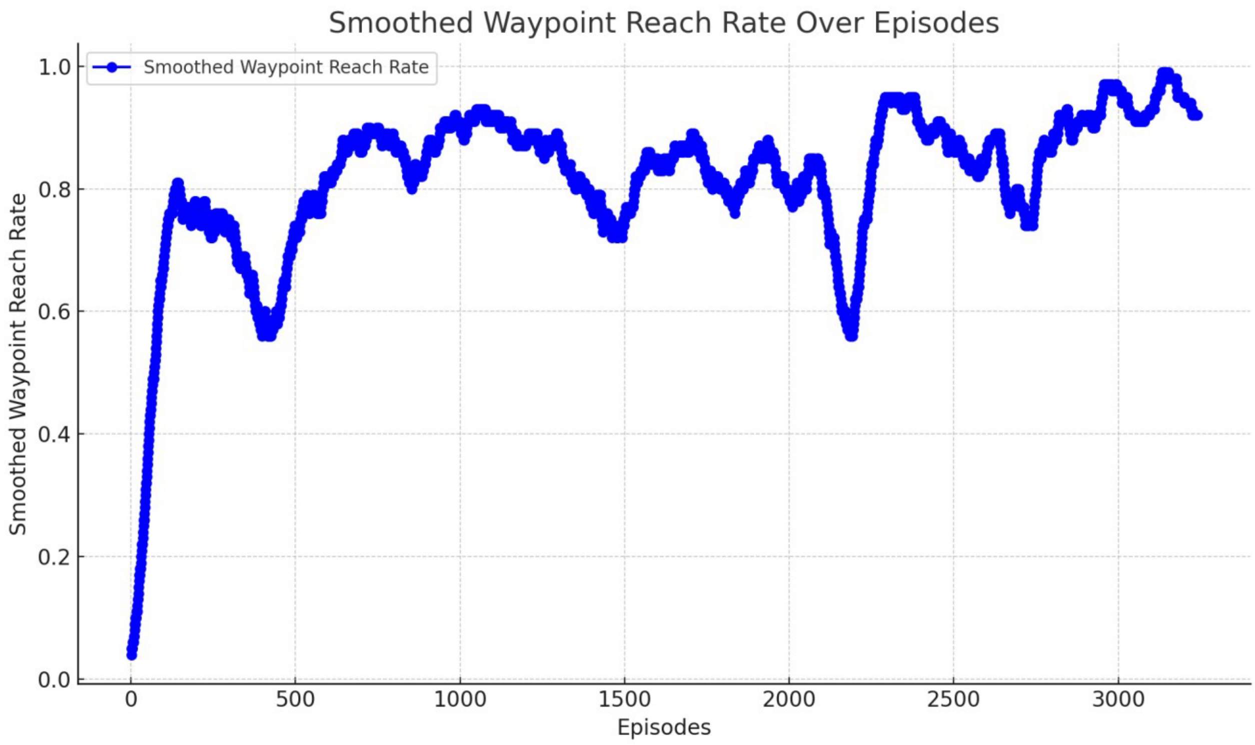


Figure 19. Improved waypoint reach rate

As the above figure, we can see how the waypoint reach rate of the agent increases over episodes, which means that the agent learns how to reach it while avoiding conflicts. In total it reaches the waypoint a total of 2.697 times out of 3.338 episodes, which means a 80,80% of waypoint reach rate. As said before, we can appreciate a few downs in the curve as the agent carried explorations to find different options to maximize its cumulative rewards, leading to episodes without reaching the waypoint and having intrusion conflicts.

5.2.4 TRAINING CONCLUSIONS AND EVALUATION

After running the PPO agent's training, I consider it as successful as the agent autonomously learns how to avoid conflicts with other drones while it follows a designated waypoint. In this improved version, there is more control over learning dynamic, and the obtained reward increases over episodes while the intrusions decrease, so it means that the designed reward function is working and the agent tries to maximize it avoiding intrusions and being properly aligned with the waypoint. I perform the training evaluation among 10 episodes, and the obtained results are:

EPISODE	REWARD	STEPS	CONFLICT
1	+86,001	300	False
2	- 86,267	300	True
3	+88,277	300	False
4	+89,384	300	False
5	+88,877	300	False
6	+86,163	300	False
7	+87,784	300	False
8	+91,188	300	False
9	+92,650	300	False
10	+91,458	300	False

During the evaluation, the PPO agent performed 10 episodes with a 90% success rate, as there is an episode with a light intrusion which led to a negative reward. For this project, these results justify and demonstrate the viable possibility of using AI-driven algorithms to assume some ATC tasks, as it demonstrates how an agent learns how to solve and avoid conflicts over training episodes by adjusting its heading. It also shows the AI's capacity for decision-making and conflict-resolution tasks in an autonomous urban airspace environment transited by drones.

5.3 COMPARISON WITH OTHER ALGORITHMS

After evaluating PPO's effectiveness to manage air traffic scenarios, this project will proceed to train the agent with different RL algorithms within the same simulating environment. This aims to determine which method is more suited for decision-making and conflict-avoidance tasks in ATM. These algorithms performances will be evaluated according to their reward per episode, the number of conflicts or intrusions per episode and their TimeLimit truncation rate.

After carefully reviewing the different RL algorithms, I consider that the hybrid methods SAC, DDPG or TD3 are the most suitable because of their capacity to adapt to changing conditions and continuous spaces, such as airspace traffic.

5.3.1 SAC

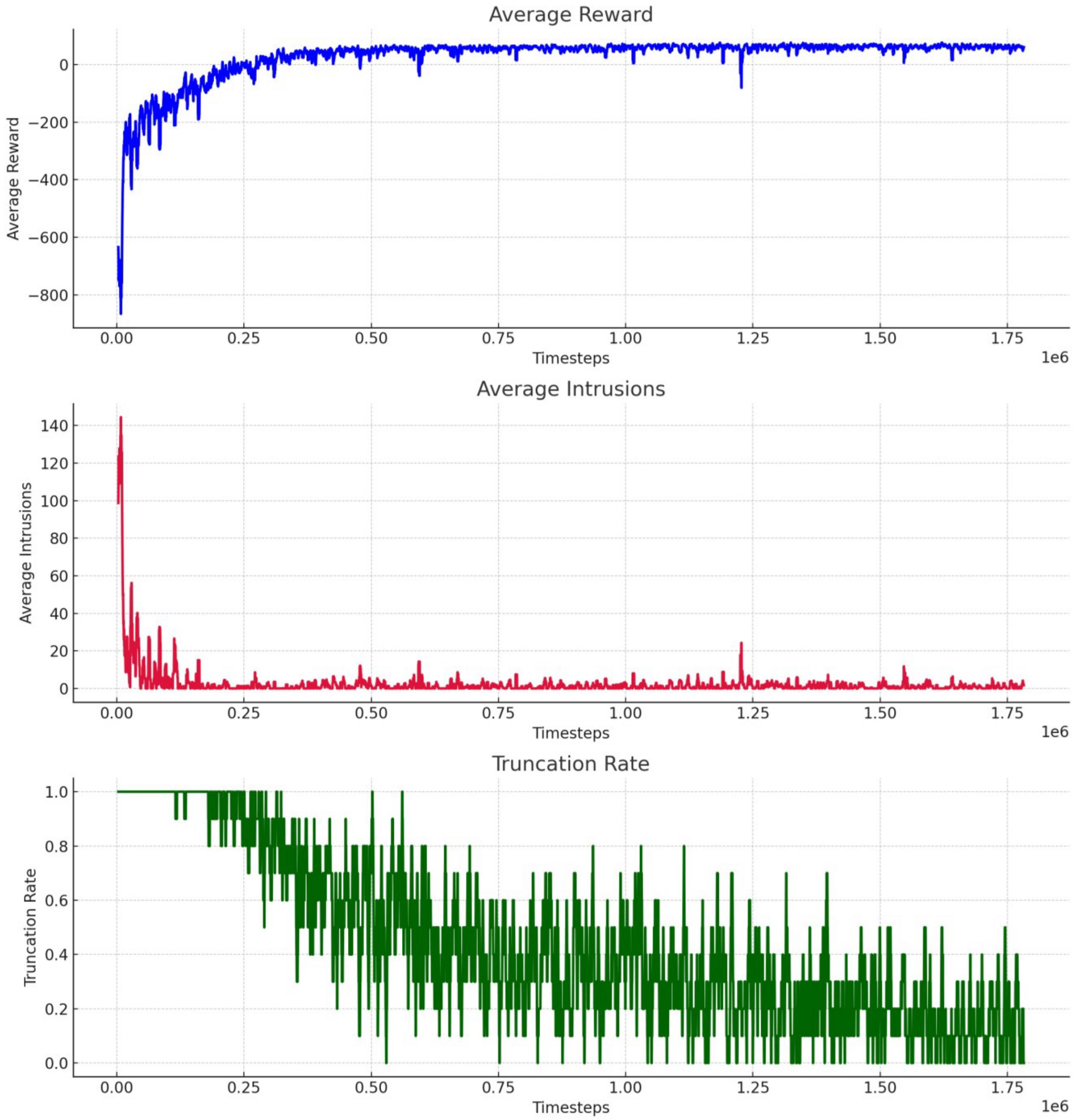


Figure 20. SAC training graphs

As the above graphs, we can see how the average reward per episode increases over time until reaching reward values around +80, but in a less sharply way in comparison with the PPO agent. For average intrusions, the total amount per episode considerably decreases to low levels, but the high levels of TimeLimit truncations indicate that beside its ability to avoid intrusions the agent does not reach the waypoint objective at most part of the times. In summary, the agent is good at avoiding intrusions but poor at reaching the objective.

5.3.2 DDPG

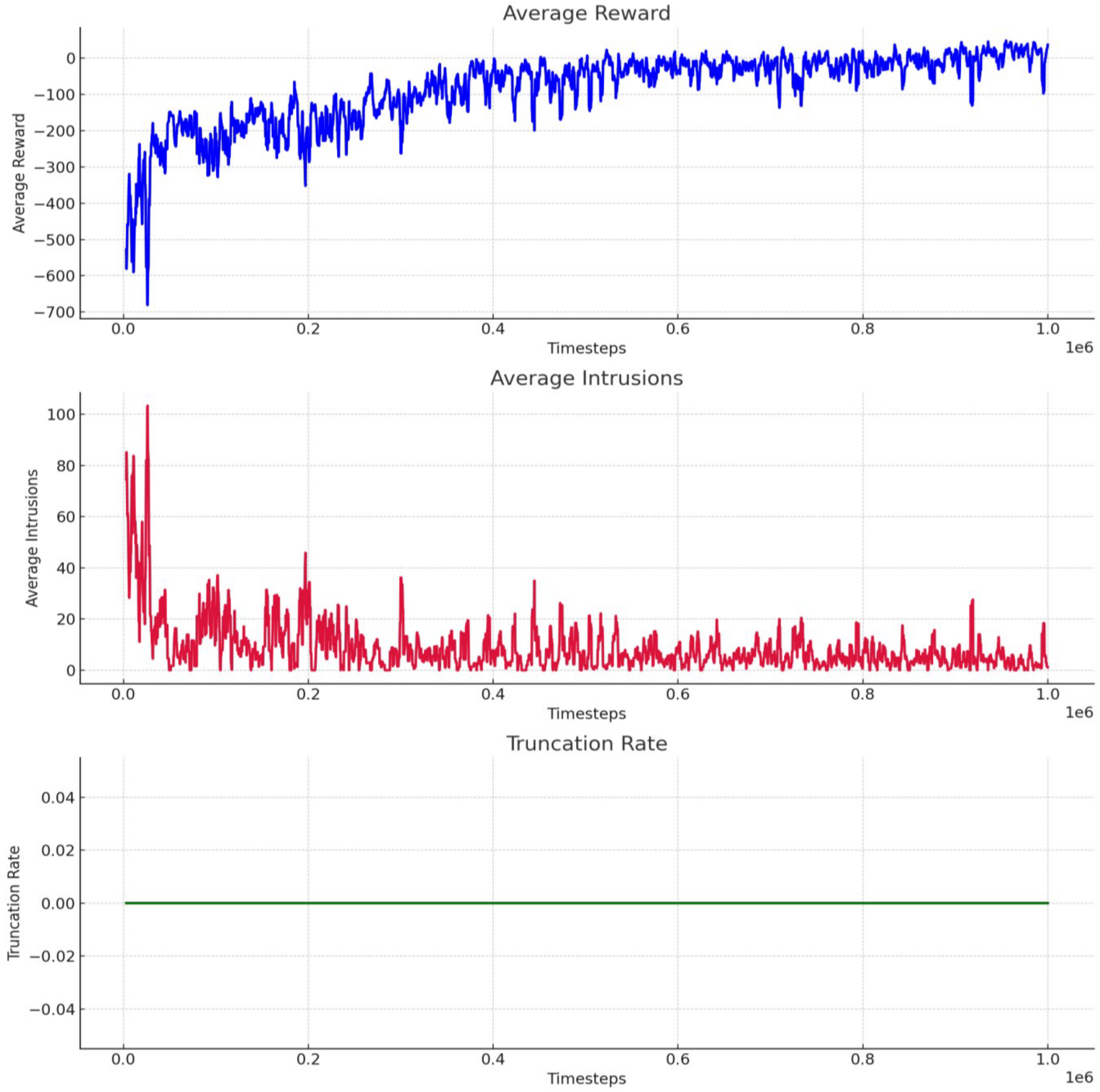


Figure 21. DDPG training graphs

As the above graphs, we can see how the average rewards stay negative during all training and there is no clear convergence in the results. In relation with the intrusion avoiding, it also remains high during all training despite little improvements, so it means the agent does not have a strong conflict avoiding policy learning. Finally, the agent always reaches the final objective but committing intrusion conflicts which means it fails to learn an effective policy.

5.3.3 TD3

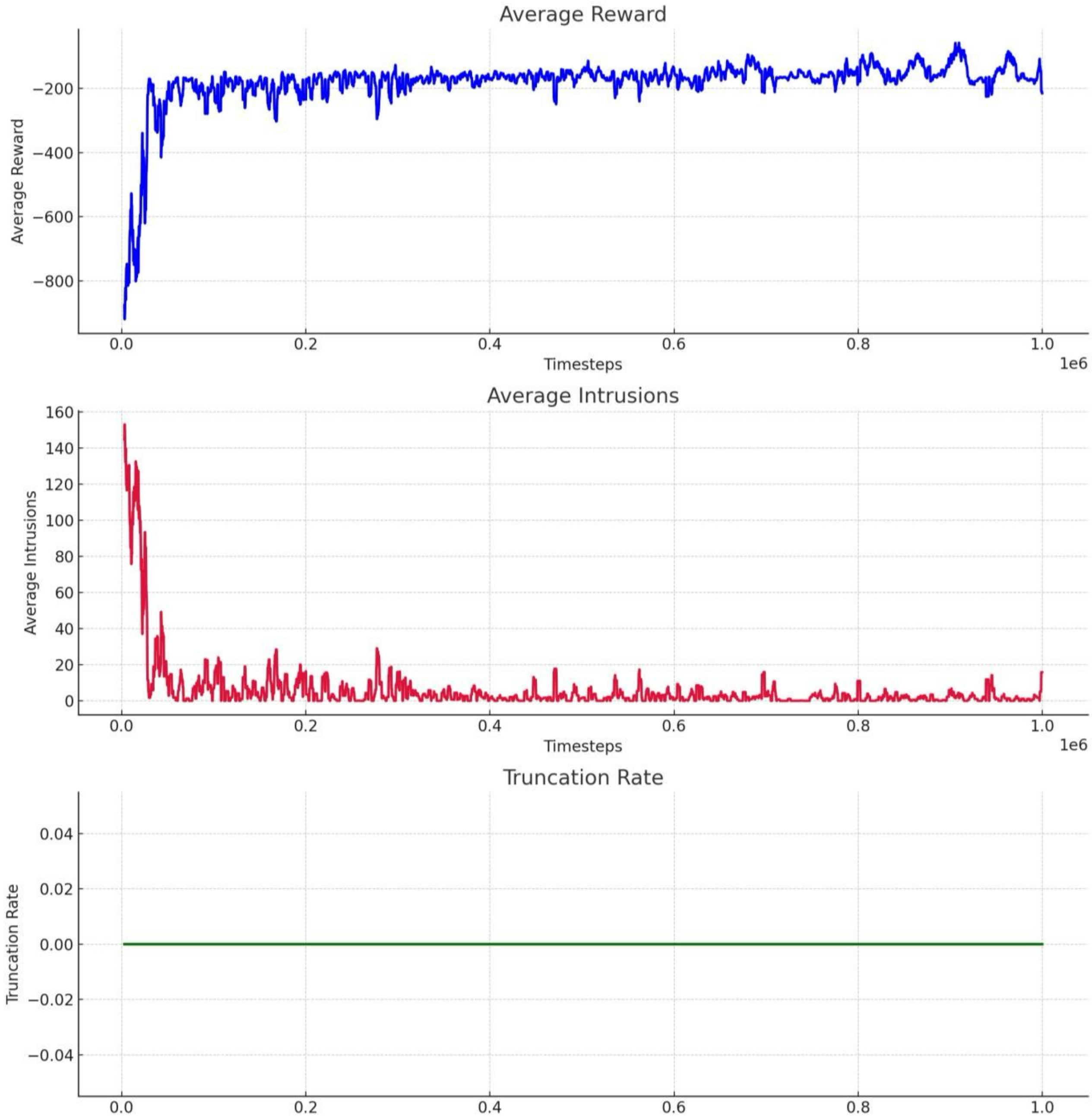


Figure 21. TD3 training graphs

As the above graphs, we can see how the average rewards stay negative during training even lower than the DDPG method. The average intrusions also start high and decrease slightly despite the agent reaches the objective all times, so it means the agent has the same issue as DDPG and the policy learning is completely ineffective.

5.3.4 COMPARISON AND CONCLUSIONS

METRIC	PPO	DDPG	TD3	SAC
Average reward	40.34	-83.07	-182.33	26.03
Total intrusions	10,104	30,825	21,563	16,117
Average intrusions	2.9	8.81	6.47	2.6
TimeLimit truncation rate	19.7%	0%	0%	43.9%

To evaluate the performance of PPO against other algorithms such as DDPG, TD3 and SAC, multiple training runs were conducted in the same simulation environment. The performance of each algorithm was measured in total reward, number of conflict intrusions and TimeLimit truncation rate.

After conducting the training simulations, we see that the PPO agent outperformed all other models with the highest average reward and the lowest average number of intrusions per episode, which means it has the most effective policy learning for safety (conflict-avoidance and intrusion separations) and for reaching the desired waypoint objective. Despite SAC had competitive results in safety, it did not learn to reach the waypoint as it was mainly focusing on avoiding collisions with other drones. Finally, DDPG and TD3 basically underperformed, failing to learn an effective policy for ensuring safety and for reaching the objective waypoint.

As a result, these conclusions reinforce the correct decision of selecting PPO algorithms for high-density and dynamic environments like Urban Air Mobility, where robustness, adaptability and safety are crucial and necessary to ensure safe and effective operations.

6. PROJECT IMPLEMENTATION IMPACTS

After carefully studying and investigating the potential impacts of the implementation of the U-Space project and an AI-based ATM and UTM, I can briefly extract some consequences classified by social, economic and environmental impacts.

6.1 SOCIAL IMPACTS

The social impacts of implementing UAM with autonomous traffic management are:

- **Accessibility and inclusivity:** EASA promises that the UAM integration will improve access for people with reduced mobility and will provide better accessibility for regions with poor traditional transit services.
- **Community impact:** the World Economic Forum emphasizes the importance of engaging communities near Vertiports during the planning and testing phases of the project to address their concerns about noise, safety and privacy. By this way, the operating hours, noise limits and buffer zones should be clearly regulated in order to ensure their comfort and acceptance.
- **Training and education:** UAM implementation will suppose the necessity of creating new job positions and their related trainings and formations. Some training programs will be delivered for UAT management, AI operators…

[44, 45]

6.2 ECONOMIC IMPACTS

The economic impacts of implementing UAM with autonomous traffic management are:

- **High initial investments:** according to Sesar JU, the implementation of the project and UAM integration will require high economical investments in digital infrastructure, AI networks and airspace services. However, it is expected that the automated systems will increase operative efficiency generating returns in the long term by operational savings and passenger revenues.
- **Job creation:** the project is expected to generate around 90.000 job positions in the European Union by 2.030 in UAV manufacturing, software engineering, traffic monitoring and vertiport operations. Besides these, there would also be job opportunities in regulatory and maintenance services.
- **Industry competitiveness:** the appearance of UTM will present a competitive and innovative opportunity for regional and national sectors. Early movers in integrating UAM will have potential for dominating future commercial aerial services, because of the increasing trend in urban congestion where traditional ATM becomes inefficient.

[46, 47]

6.3 ENVIRONMENTAL IMPACTS

The environmental impacts of implementing UAM with autonomous traffic management are:

- **Co2 emissions:** eVTOLs could produce between 30% and 40% less emissions per passenger-kilometer in comparison with the current combustion-powered vehicles. In addition, UTM systems could also optimize route emissions by minimizing route deviation and time spent on holdings.
- **Energy efficiency:** AI-driven flight path planning and Multi Agent Systems (MAS) will optimize traffic flows, reducing over 20% of the energy consumption by avoiding traffic congestion and inefficient routes.
- **Biodiversity impact:** UAM routes will be designed in order to avoid ecologic sensitive areas. As a matter fact, FAA advises that there would be defined no-fly zones over protected habitats, and AI-driven rerouting systems will dynamically avoid wildlife activity and environmental alerts.

[48, 49]

6.4 GENDER PERSPECTIVE

Looking at a future society, it is crucial and fundamental that UAM implementation with UTM systems offers equal opportunities and job distributions for both genders. As a result, FAA and Sesar JU designers promise:

- **Participation and workload balance:** UAM planning and implementation should ensure gender equality in hiring, training, and decision-making roles.
- **Stereotypes and culture:** initiatives such as Women in Aerospace Europe (WIAE) promote cultural changes in traditional aviation gender roles through scholarship, mentorship and leadership supports.

Gender diversity is not only for ethical accomplishments. It supposes a strategic advantage for companies to design safer and more innovative air traffic systems. By including women in all stages of the development process, the ATM sector gains creativity, resilience, and inclusivity.

[50]

7. CONCLUSIONS AND FUTURE WORK

In summary, this project aimed to implement an AI algorithm to an agent, PPO, in the Bluesky simulator environment to manage drone traffic in a dynamic urban environment. The PPO agent had the objective of reaching a generated waypoint while adjusting its heading and velocity in order to avoid collisions and intrusions with other transiting drones. The scope of investigation was simplified to a single-agent PPO architecture in a 2D space to set a solid base potentially upgraded in the short term. After many attempts and corrections, the results of the training and evaluating episodes were successful as the PPO agent finally learnt how to avoid all intrusions while being well aligned with the waypoint objective and reaching it.

However, to get a more realistic scenario, is necessary to amplify the project scope to develop a better system capable of being implemented in real scenarios for UAM. The main future line work where this project can advance to is by incorporating also altitude parameters changes in a 3D space to perform simulations in scenarios similar to urban airspace. In addition, adding weather and transit density factors would be interesting to study how the agent performs and reacts in front of unexpected and conditioning events.

On the other hand, one interesting feature to implement would be the integration of MAS capabilities into different PPO agents transiting the airspace. By simultaneously coordinating themselves to maintain safe separations and avoiding conflicts while moving towards their defined waypoint, negotiating flight paths, managing handling conflicts and coordinating landings at busy Vertiports. This implementation would mean more than a basic structure for future UTM and a huge step for the idea of moving towards an autonomous UAM system in urban airspaces.

REFERENCES

[1] European Aviation Safety Agency. “What is UAM.”. Access 13th of February 2025. https://www.easa.europa.eu/en/what-is-uam

[2] Mehmet Necati, Cizreliogulari. Cyprus Science University. “Future Air Transportation Ramification: Urban Air Mobility (Uam) Concept: Urban Air Mobility”. 3rd of May 2022. https://www.academia.edu/88074391/Future_Air_Transportation_Ramification_Urban

[3] Gollnick, V. (2021). *Methodology and first Results for an Urban Air Mobility System in Hamburg*. https://doi.org/10.13140/RG.2.2.13587.91688

[4] Waltz, M., Okhrin, O., & Schultz, M. (2024). Self-organized free-flight arrival for urban air mobility. *Transportation Research Part C: Emerging Technologies*, *167*. https://doi.org/10.1016/j.trc.2024.104806

[5] Mehmet Necati. Research Gate. “Different UAM aircraft design type”. (Garrow et al, 2021). August 2021: https://www.researchgate.net/figure/Different-UAM-aircraftdesign-type-Garrow-et-al-2021_fig1_363151069

[6] Skybrary. Articles. “Air Traffic Management (ATM)”. Access: 20th of February. https://skybrary.aero/search/google?keys=air+traffic+management

[7] Chen, Z., Zhu, Y., Pu, F., & Tian, W. (2024). A study on basic research priorities and development suggestions for the digital transformation of air traffic management. *Aerospace Traffic and Safety*, *1*(1), 1–9. https://doi.org/10.1016/j.aets.2024.06.004

[8] Ren, L., & Castillo-Effen, M. (2017). *GE Global Research Technical Information Series Air Traffic Management (ATM) Operations: A Review*. https://www.researchgate.net/profile/Liling-Ren/publication/323244123_Air_Traffic_Management_ATM_Operations_A_Review/links/5a8c94f60f7e9b2285908afa/Air-Traffic-Management-ATM-Operations-A-Review.pdf

[9] Spalas, K. (2024). *Towards the Unmanned Aerial Vehicle Traffic Management Systems (UTMs): Security Risks and Challenges*. http://arxiv.org/abs/2408.11125

[10] Pham, D. T., Alam, S., & Duong, V. (2020). An Air Traffic Controller Action Extraction-Prediction Model Using Machine Learning Approach. *Complexity*, *2020*. https://doi.org/10.1155/2020/1659103

[11] Xie, Y., Pongsakornsathien, N., Gardi, A., & Sabatini, R. (2021). Explanation of machine-learning solutions in air-traffic management. *Aerospace*, *8*(8). https://doi.org/10.3390/aerospace8080224

[12] Mahesh, B. (2020). Machine Learning Algorithms - A Review. *International Journal of Science and Research (IJSR)*, *9*(1), 381–386. https://doi.org/10.21275/art20203995

[13] Yousefzadeh Aghdam, M., Kamel Tabbakh, S. R., Mahdavi Chabok, S. J., & Kheyrabadi, M. (2021). Optimization of air traffic management efficiency based on deep learning enriched by the long short-term memory (LSTM) and extreme learning machine (ELM). *Journal of Big Data*, *8*(1). https://doi.org/10.1186/s40537-021-00438-6

[14] Pinto Neto, E. C., Baum, D. M., Almeida, J. R. de, Camargo, J. B., & Cugnasca, P. S. (2023). Deep Learning in Air Traffic Management (ATM): A Survey on Applications, Opportunities, and Open Challenges. In *Aerospace* (Vol. 10, Issue 4).

https://doi.org/10.3390/aerospace10040358

[15] Wu, C., Ding, H., Fu, Z., & Sun, N. (2024). Air Traffic Flow Prediction in Aviation Networks Using a Multi-Dimensional Spatiotemporal Framework. *Electronics (Switzerland)*, *13*(19). https://doi.org/10.3390/electronics13193803

[16] Canese, L., Cardarilli, G. C., di Nunzio, L., Fazzolari, R., Giardino, D., Re, M., & Spanò, S. (2021). Multi-agent reinforcement learning: A review of challenges and applications. In *Applied Sciences (Switzerland)* (Vol. 11, Issue 11).

https://doi.org/10.3390/app11114948

[17] Van der Hoff, D. (2020). *A Multi-Agent Reinforcement Learning Approach to Air Traffic Control*. Delft University of Technology.

https://resolver.tudelft.nl/uuid:4d02e51a-0187-404d-b465-7ae01feba8e8

[18] Wooldridge, M. (2009). An Introduction to MultiAgent Systems - 2nd Edition. In *ACM SIGACT News* (Vol. 41, Issue 1).
[19] Park, C., Kim, G. S., Park, S., Jung, S., & Kim, J. (2023). Multi-Agent Reinforcement Learning for Cooperative Air Transportation Services in City-Wide Autonomous Urban Air Mobility. *IEEE Transactions on Intelligent Vehicles*, *8* (8).
https://doi.org/10.1109/TIV.2023.3283235

[20] Kopardekar, P., Rios, J., Prevot, T., Johnson, M., Jung, J., & Robinson, J. E. (2016). Unmanned aircraft system traffic management (UTM) concept of operations. *16th AIAA Aviation Technology, Integration, and Operations Conference*.

https://ntrs.nasa.gov/citations/20190000370

[20] A. Alharbi, A. Poujade, K. Malandrakis, I. Petrunin, D. Panagiotakopoulos and A. Tsourdos, "Rule-Based Conflict Management for Unmanned Traffic Management Scenarios," *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*, San Antonio, TX, USA, 2020, pp. 1-10, doi: 10.1109/DASC50938.2020.9256690.
https://ieeexplore.ieee.org/abstract/document/9256690

[21] EASA Drones Team. “What is U-Space". Access: 19th February 2025.
https://www.easa.europa.eu/en/what-u-space

[22] U-Space and Enaire’s role. “Everything you need to know to fly your drone”. EASA. Access: 19th February 2025.

https://www.enaire.es/services/drones/everything_you_need_to_know_to_fligh_your_drone/uspace_and_enaires_role

[23] Sesar Joint Undertaking. “Smart ATM: U-Space and urban air mobility”. Access: 19th February 2025. https://www.sesarju.eu/U-space

[24] Dron Europa. “U-Space”. Access: 18th February 2025.
https://www.droneuropa.com/U-Space/

[25] A. Sipe and J. Moore, "Air traffic functions in the NextGen and SESAR airspace," *2009 IEEE/AIAA 28th Digital Avionics Systems Conference*, Orlando, FL, USA, 2009, pp. 2.A.61-2.A.6-7, doi: 10.1109/DASC.2009.5347554.

[26] Federal Aviation Administration. "Next Generation Air Transportation System". Access: 16th February 2025.

https://www.faa.gov/nextgen

[27] Vonk, B. (2019). *Exploring reinforcement learning methods for autonomous sequencing and spacing of aircraft*. Delft University of Technology. https://resolver.tudelft.nl/uuid:2e776b60-cd4e-4268-93e3-3fcc81cd794f

[28] Ribeiro, M., Ellerbroek, J., & Hoekstra, J. (2020). Determining Optimal Conflict Avoidance Manoeuvres At High Densities With Reinforcement Learning. *SESAR Innovation Days*. https://www.researchgate.net/publication/346647401

[29] Sutton, R. S., & Barto, A. G. (2017). Reinforcement Learning, Second Edition An Introduction. In *Encyclopedia of Machine Learning and Data Mining*.

[30] Brittain, M., & Wei, P. (2021). One to any: Distributed conflict resolution with deep multi-agent reinforcement learning and long short-term memory. *AIAA Scitech 2021 Forum*. https://doi.org/10.2514/6.2021-1952

[31] Brittain, M. W., Yang, X., & Wei, P. (2021). Autonomous separation assurance with deep multi-agent reinforcement learning. *Journal of Aerospace Information Systems*, *18*(12). https://doi.org/10.2514/1.I010973

[32] Liu, Z.". (2025). Value-Based Reinforcement Learning. In: Artificial Intelligence for Engineers. Springer, Cham. https://doi.org/10.1007/978-3-031-75953-6_14

[33] Byeon, H. (2023). Advances in Value-based, Policy-based, and Deep Learningbased Reinforcement Learning. *International Journal of Advanced Computer Science and Applications*, *14*(8). https://doi.org/10.14569/IJACSA.2023.0140838

[34] A. d. Rio, D. Jimenez and J. Serrano, "Comparative Analysis of A3C and PPO Algorithms in Reinforcement Learning: A Survey on General Environments," in *IEEE Access*, vol. 12, pp. 146795-146806, 2024, doi: 10.1109/ACCESS.2024.3472473.

[35] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). *Proximal Policy Optimization Algorithms*. http://arxiv.org/abs/1707.06347

[36] Hoekstra, J., Ellerbroek, J., & Hoekstra, J. M. (2016). *BlueSky ATC Simulator Project: an Open Data and Open Source Approach*.
https://www.researchgate.net/publication/304490055

[37] Iberica Dron. Tienda. Dji matrice 600. Access in 2025.
https://www.ibericadron.com/tienda/dji/dji-matrice/matrice-600/

[38] J. M. Hoekstra and J. Ellerbroek, "BlueSky ATC Simulator Project: an Open Data and Open Source Approach", Proceedings of the seventh International Conference for Research on Air Transport (ICRAT), 2016.https://github.com/TUDelft-CNS-ATM/bluesky/blob/master/bluesky/resources/performance/OpenAP/rotor/aircraf t.json

[39] Stable Baselines 3. Modules. PPO algorithm. Access in 2025.
https://stable-baselines3.readthedocs.io/en/master/modules/ppo.html

[40] Github. MJRibeiro TU Delft. Bluesky. Access in 2025.

https://github.com/MJRibeiroTUDelft/bluesky

[41] Github. Marc Brittain . Autonomous ATC. Access in 2025.
https://github.com/marcbrittain/Autonomous-ATC

[42] Github. Devanderhoff. Bluesky. Access in 2025.

https://github.com/devanderhoff/bluesky

[43] EASA. (2021). *Study on the societal acceptance of Urban Air Mobility in Europe*. https://www.easa.europa.eu/sites/default/files/dfu/uam-full-report.pdf

[44] World Economic Forum. (2021). *Principles of the Urban Sky.* https://www.weforum.org/publications/principles-of-the-urban-sky/

[45] SESAR Joint Undertaking. (2023). *U-Space CONOPS 4th Edition* https://www.sesarju.eu/node/4544

[46] NASA Technical Reports Server. (2018). https://ntrs.nasa.gov/citations/20190001472

[47] R. A. Saeed, E. S. Ali, M. Abdelhaq, R. Alsaqour, F. R. A. Ahmed and A. M. E. Saad, "Energy Efficient Path Planning Scheme for Unmanned Aerial Vehicle Using Hybrid Generic Algorithm-Based Q-Learning Optimization," in *IEEE Access*, vol. 12, pp. 1340013417, 2024, doi: 10.1109/ACCESS.2023.3344455.

[48] Hoffmann, R., Silva, F., & Nishimura, H. (2024). Evaluating the Eco-Efficiency of Urban Air Mobility: Understanding Environmental and Social Impacts for Informed Passenger Choices. *INCOSE International Symposium*, *34*(1), 967–984. https://doi.org/10.1002/iis2.13189

[49] SESAR Joint Undertaking. (2024). *Mind the gap: why gender equality in air traffic management matter.*

https://www.sesarju.eu/news/mind-gap-why-gender-equality-air-traffic-managementmatters#:~:text=To%20mark%20International%20Women%E2%80%99s%20Day%2C %20the%20SESAR%20Joint,can%20be%20done%20to%20bridge%20the%20gender%20gap.