D3.1 - ATCO Tasks prediction model

Deliverable ID:	**D3.1**
Project acronym:	**CODA**
Grant:	**101114765**
Call:	**HORIZON-SESAR-2022-DES-ER-01**
Topic:	**HORIZON-SESAR-2022-DES-ER-01-WA2-1**
Consortium coordinator:	**DBL**
Edition date:	**20 December 2024**
Edition:	**00.02.00**
Status:	**Final**
Classification:	**PU**

Abstract

This document contains the results of the CODA WP3.1 task, related to short term prediction of ACTO tasks based on data-driven models. The document describes two different approaches to perform this prediction, their architectures, training processes and validation. The purpose of this model is twofold, aiming to provide a rich input to predict the evolution of ATCO mental state, and helping elicit the collection of tasks to be implemented to assist the adaptive automation share tasks between the controller and a supporting digital assistant (automating some of the task types).

In this version of the document, both approaches are completely described. The results of the first, in its current state, although potentially promising, are demonstrated to be not good enough, due to data limitations. So, a second approach, based on a data-driven divide-and-conquer strategy is also described. This second strategy has been also implemented and its results reported in this second version of this document. Its objective is not so ambitious, but more realistic, and although it may have some impact on the final implementable adaptive automation strategy, the results show it may be good enough and specially much more explainable.

Authoring & approval

Author(s) of the document

Organisation name	Date
UPM	20/12/2024

Reviewed by

Organisation name	Date
DBL	19/12/2024
ENAC	18/12/2024
BS	17/12/2024
NLR	18/12/2024
ECTL	18/12/2024
IFATCA	17/12/2024
UGR	17/12/2024
CRIDA	18/12/2024

Approved for submission to the SESAR 3 JU by[1]

Organisation name	Date
DBL	19/12/2024
ENAC	18/12/2024
BS	17/12/2024
NLR	18/12/2024
ECTL	18/12/2024
IFATCA	17/12/2024
UGR	17/12/2024
CRIDA	18/12/2024

[1] Representatives of all the beneficiaries involved in the project

EUROPEAN PARTNERSHIP
Co-funded by the European Union

Rejected by[2]

Organisation name	Date

Document History

Edition	Date	Status	Company Author	Justification
00.00.01	13/02/2024	Draft	UPM	Initial description of architecture and tests. Table of contents.
00.00.02	06/05/2024	Draft	UPM	Inclusion of description of input, output and intermediate data
00.00.03	26/07/2024	Draft	UPM	Inclusion of Initial validation results
00.00.04	16/09/2024	Draft	UPM	Complete description of initial approach to the problem
00.00.05	23/09/2024	Draft	UPM	Initial description of second approach to the problem and document cleanup for review by partners
00.00.06	28/09/2024	Draft	All	Comments
00.01.00	30/09/2024	Final (v1)	UPM	Final cleanup for submission
00.01.90	02/12/2024	Draft (v2)	UPM	Implementation of changes based on SJU review
00.01.91	10/12/2024	Draft (v2)	UPM	Inclusion of complete v2 version description, including algorithms and validation
00.01.92	11/12/2024	Draft (v2)	UPM	Update of results and discussion on v1 approach
00.01.93	13/12/2024	Draft (v2)	UPM	Document cleanup for review by partners
00.01.94	18/12/2024	Draft	All	Comments
00.02.00	19/12/2024	Final (v2)	UPM	Final cleanup for submission

[2] Representatives of the beneficiaries involved in the project

CODA

CONTROLLER ADAPTIVE DIGITAL ASSISTANT

This document is part of a project that has received funding from the SESAR 3 Joint Undertaking under grant agreement No 101114765 under European Union's Horizon Europe research and innovation programme.

Table of contents

List of figures

EUROPEAN PARTNERSHIP

Co-funded by the European Union

List of tables

EUROPEAN PARTNERSHIP

Co-funded by the European Union

1 Introduction

This document contains the results of the CODA WP3.1 task related to the short-term prediction of ATCO tasks based on data-driven models. The document describes two approaches to performing this prediction, their architectures, training processes and validation. The purpose of this model is twofold: aiming to provide a rich input to predict the evolution of the ATCO mental state and helping elicit the collection of tasks to be implemented to assist the adaptive automation shared tasks between the controller and a supporting digital assistant (automating some of the task types). Its place within the complete CODA system is depicted in Figure 1.

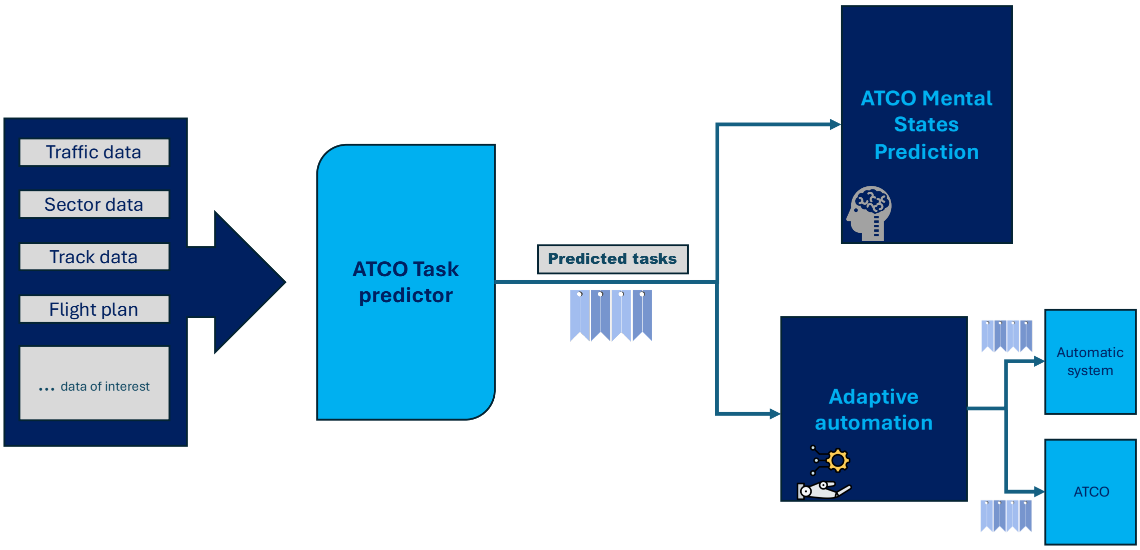


Figure 1: Potential ATCO task predictor roles in CODA architecture

Ideally, the task predictor would calculate a list of time-stamped tasks for the near future, which will later become an input for the rest of the CODA system. In any case, predicting ATCO tasks with a high degree of fidelity is a big challenge. As an indicator of the difficulty to do so, the CODA Advisory Board was explicitly asked about the feasibility of such a prediction approach in the Advisory Board Meeting held on 15th December 2023. After describing the task prediction model being designed and implemented in CODA, the AB members were asked three questions related to task prediction for their consideration.

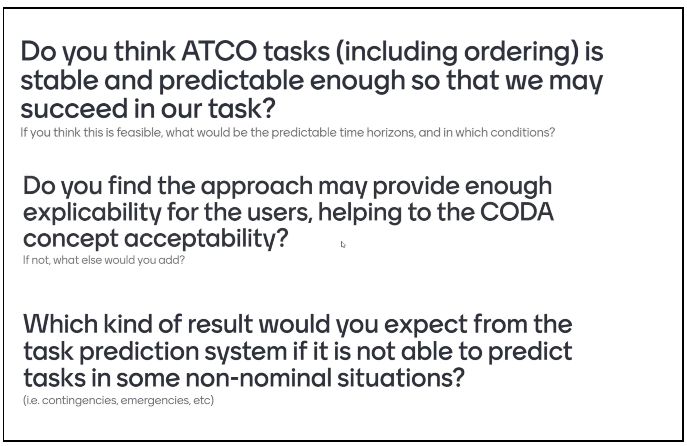


Figure 2: Questions to CODA AB on ATCO task prediction feasibility and role

The AB members directly provided no answer, and after an indication from the CODA team on the difficulty to answer them without additional research, no further reaction was received, which was understood by the CODA researchers as an agreement to the difficulty to fully address the problem.

After a first approach for the implementation of WP3.1 task was followed, results that were not considered good enough were obtained, with our current available data. Therefore, the CODA team decided to define a second approach. Its objective is not so ambitious but more realistic (i.e. not trying to provide exact task execution timestamps), and depending on the final adaptive automation strategy, it could be good enough.

This version of the document completely describes both approaches, their results and validation.

2 ATCO Task Prediction Problem Statement

2.1 Why an ATCO Task Prediction Model?

Developing an ATCO (Air Traffic Controller) task prediction model is a critical element of CODA as it forms the basis of the task allocation strategy based on adaptive automation principles. Predicting future tasks for ATCOs can significantly improve safety, operational efficiency, and overall controller performance by enabling management of the controller workload and the associated mental state.

If neurophysiological assessment can give information on the current state of operators to improve the interaction between humans and AI-based systems, the next step would be to be able to predict the future mental states to intervene in advance and avoid unwanted situations (e.g. workload peaks) that can immediately impact performance and be hard to recover from. A potential approach to achieve this would be to anticipate which tasks will need to be executed by the operator and the AI based systems in the future, so that time intervals with a high rate of concurrent and complex tasks would be related to workload peaks.

By forecasting which tasks controllers (with the support of AI based digital assistants) will need to perform, the model would allow for the identification of potential high-workload periods or complex situations before they arise. This foresight would support effective task allocation to the controller or to the digital assistants, potentially mitigating controller stress and fatigue and, therefore, enhancing decision-making. Furthermore, the model would help automate task scheduling and prioritisation, streamlining operational workflows and reducing human error. As a result, the task prediction model would improve the management of high-density traffic by addressing variability in traffic patterns and controller responses.

2.2 State of the art of task prediction models

This section describes some approaches in the literature for measuring, forecasting, and predicting controller tasks in this high-risk environment. Additionally, it mentions the challenges and limitations faced by task prediction, such as the dynamic nature of air traffic, and the requirements any method aimed at this must meet, including data security and privacy and real-time technological infrastructure.

2.2.1 Introduction

Air traffic controllers (ATCOs) play a crucial role in maintaining the safe, orderly, and efficient flow of air traffic while maintaining effective communication with pilots to ensure compliance with aviation clearances. These professionals serve in roles such as tower, en-route, area, approach, oceanic and terminal radar Controllers. They have various duties and responsibilities: communication, traffic management, safety assurance, emergency response, navigational assistance, documentation, training, and ongoing professional development. The working environment of an air traffic controller is often fast-paced, dynamic, and highly demanding. Controllers must remain focused and attentive to their workspace's multiple screens, radar displays, and communication systems. They work in a regulated environment, following procedures and protocols to ensure aircraft's safe and efficient movement, but with significant degrees on freedom in the strategies they apply. Their multi-faceted role includes critical responsibilities such as air traffic control clearances, contact procedures, instrument and missed approaches, radar vectoring, safety warnings, speed adjustments and traffic advisories, all with their associated tasks. All these tasks include the meticulous confirmation of pilots' readbacks in air ground communications, monitoring of related manoeuvres, ensuring that vital instructions on altitude, flight levels, track, radar vectors and other parameters are followed accurately.

ATCOs also perform other tasks, such as maintaining constant vigilance, ready to address any potential disruptions or delays affecting the aircraft's trajectory. The most relevant controller task, critical from the safety point of view, is providing radar heading vector guidance or other separation-oriented clearances and instructions to avoid proximate traffic in conflict. Also, sequencing related clearances and instructions that allow a more efficient traffic flow. Their commitment to tracking and relaying critical traffic information continues until the aircraft lands safely or moves to the next ATCO responsibility sector.

When operating within a high-stakes environment, safety and efficiency are paramount, as described in [1]. Prediction models are crucial in identifying potential incidents or issues in advance, enabling ATCOs to take proactive measures to prevent risks, minimise damage, and enhance security. Additionally, anticipating future requirements allows ANSPs to allocate resources more efficiently, reducing costs. The general moto of CODA project is that task prediction and the use of adaptive automation might also aid in process optimisation, potentially leading to ATCO error reduction, and assuring higher levels of quality and efficiency.

2.2.2 Methods for ATCO Task Prediction

Some different approaches for ATCO task predictions are described next.

2.2.2.1 Historical Data Analysis

Analysing historical air traffic patterns is essential for efficient and safe airspace management. This process involves collecting and studying data on past flights, including routes, schedules, aircraft types, weather conditions and congestion. This enables aviation authorities to forecast future traffic demand, optimise routes, plan airport expansions, proactively manage congestion, allocate resources efficiently and manage airspace safely.

A key concept here is traffic complexity, with many attempts to define, measure and predict it from historical flight analysis. For instance, the early paper from Chatterji & Sridhar [2] presented sixteen complexity measures describing air traffic patterns. Other classical and modern attempts to study this problem can be found in [3] or [4].

With a much higher granularity, the paper from D. Karikawa [5] presents the COMPAS (Cognitive system Model for Simulating Projection-based behaviours of Air traffic controllers in dynamic Situations) system, which allows automatic ATCO task identification and visualisation tools based on cognitive system simulation of an ATCOs.

Of particular interest for CODA is the existence of the CRIDA data warehouse, which has stored historical data since 2018 on flight plans, radar tracks, sector configurations and ATC activity of ENAIRE. A system called ATON, quite similar to COMPAS, is in place in the CRIDA data warehouse to define ATCO activity. This system can semiautomatically detect ATCO actions (in the form of events). This ATC activity, in the form of tasks being executed, will be used both as an input to the CODA ATC task prediction system (in the form of previous and current tasks being performed) and as the label for the supervised training (for future tasks, structured as a plan).

In summary, authorities can anticipate peak periods and allocate resources by knowing past traffic trends, ensuring that airports and air traffic control centres can cope with traffic peaks. Historical data informs infrastructure decisions, ensuring that airports can accommodate future growth without compromising safety. In other words, analysing historical air traffic patterns provides critical insights for authorities to anticipate and prepare for impending tasks, ultimately enhancing airspace management efficiency and safety and benefiting the aviation industry and passengers. In general, this analysis is performed at the strategic level, not in real-time; this idea of complexity may also be relevant for real-time task prediction, as we will see.

2.2.2.2 Real-Time ATCO Workload Prediction

In this section the following approach is discussed: Predicting controller workload levels can be considered a pattern recognition problem and is, therefore, suitable for data-driven learning algorithms. To this aim, several potential strategies exist.

The first is to directly link workload with the traffic situation. This is the approach followed, for instance, by the paper mentioned in the previous section [2], which linked the complexity measures with the ATCO workload using a neural network. Pang also followed the same approach [6]. In this work, the problem of predicting ATCO workload based on the spatiotemporal layout of the airspace is considered a dynamically evolving time series graph classification task. This study proposes introducing multiple historical graphs into the model to predict the workload level at the next time stamp. Moreover, the spatiotemporal layout of the graph structure varies at each timestamp (i.e., the number of nodes and the connections of the graph edges), giving rise to a dynamic graph classification problem. The paper also presents a dynamic density model, building a regression model to find linear relationships between traffic complexity factors and ATCO workload. However, dynamic density metrics do not consider human cognitive capabilities, the primary source of real-world ATCO workload sources.

Related to the link between complexity and ATCO Load, the paper [7] presents a methodology to analyse and react to the expected traffic flows that will take advantage of its more predictive ATM network. The paper describes eTLM (Enhanced Traffic Load Monitoring), which aims to react to traffic complexity by dynamically adjusting the sector configurations to real traffic situations.

2.2.2.3 Real-Time ATCO Task Prediction

An alternative to direct workload derivation from traffic advocated in the CODA project is first to identify controllers' tasks. This has the additional advantage of enabling task distribution in the AI-human team. Next, we will focus on previous attempts to monitor/predict the controllers' tasks/actions.

The paper [8] describes a prediction model derived through supervised learning, where the target variables are planning controller actions (including altitude, speed and course changes). At the same time, the inputs are ADS-B data (aircraft 4D trajectory) and the spatial information about the sector, collected and processed from Aeronautical Information Publication (AIP). The system automatically derived the sector entry and existing points and identified the needed ATCO interventions with reasonable accuracy (99% for vertical manoeuvres, 80-90% for horizontal actions).

EUROPEAN PARTNERSHIP
Co-funded by the European Union

In [9], a Deep Learning approach is used to model actions related to conflict detection and resolution. It covers both the prediction of the time of the intervention and the type of resolution. For us, this approach may be especially interesting, as in the CODA problem (as in the one addressed in the paper), it is not only important to know what action will need to be performed but also when.

In any case, it should be emphasised that the previous literature focuses on actions related to conflict resolution. At the same time, our needs are more extensive, as we need to incorporate other tasks related to vigilance, routine communications, etc.

2.2.3 Challenges and limitations

ATCO task prediction has many different potential difficulties and limitations. Some of them will be summarised below:

1. Traffic uncertainty: Multiple unpredictable elements can produce unexpected changes in flight plans, even in the short term, during the different stages of flight, such as modifications derived from atmospheric conditions (especially with turbulent weather), conflict-induced modifications, and onboard emergencies.
2. ATCO task execution variability: although ATM is heavily regulated, and the types of tasks and typical interventions are identified, different ATCOs may adopt different strategies and solutions to similar problems, and the order of their execution may change.
3. Constraints on data security and privacy: To predict ATCO tasks, lots of operational data (at least sector definitions, flow definitions, recorded flight plans, recorded tracks, and ATCO actions) will need to be accessed. Some of them may have security/privacy problems, which must be solved.
4. Scalability, the system's capability to handle large data volumes, must be guaranteed to maintain effectiveness and accommodate growth.

2.3 ATCO Tasks Taxonomy

As previously described, in the rest of the document the ATCO tasks will follow the CRIDA taxonomy used by the ATON tool. It should be clarified that the defined tasks do not fully cover all ATCO activities, but only those linked to actions with the controller position interface and tools, finally leading to an interaction with a given flight crew. So, for instance, general vigilance actions are not covered. ATON records events related to the execution of the action through the interface (or using voice commands, which are also analysed). 28 different types of tasks are being recorded. Table 1 describes each of the task types.

Abbreviation	Meaning
CTE	Radar Contact
CTE31	Entry into the Shared Flow Flight Sector
CS	Flight Released/Cleared
CTE32	Departure from the Shared Flow Flight Sector

EUROPEAN PARTNERSHIP
Co-funded by the European Union

Abbreviation	Meaning
Ac5	Instruction: Level Change by Procedure
Ac6	Speed Adjustment by Procedure
Ac7	Approach Authorization
Ac8	Direct route to a point to shorten flight plan
Ac9	Provide essential information
Ac12	Changing SSR Transponder Code
Ac13	STAR Assignment
S2	Vector for Separation or Sequence
S3	Detours caused by storm areas
S4	Final vector on approach
X1	Standard or Procedural Vector
A1	Level Change for Separation or Sequence
A2	Speed Adjustment for Separation or Sequence
A3	Direct to a point for separation or sequence
A4	Standby Instruction
A6	Separation via non-approval of requirement
H1	Aircraft entry in circuit waiting over a certain point
R1	Transition to arrival
C1	Transition to flyover
Coordinations	
CO2	Receipt of authorizations, permits or instructions generated by other sectors and their retransmission to aircraft
CO5	Exchange of general information (equipment failure, general transfer conditions, etc.)
Y1	Create Flight Plan
Y2	Modify Flight Plan

Abbreviation	Meaning
Y3	Coordinated Flight Level when it involves different FL than expected by the collateral sector

Table 1: ATCO task types

2.4 ATCO Tasks Prediction Requirements and Data Sources

CRIDA provided a collection of historical datasets to research methods for ATCO task prediction by UPM. The datasets belong to two different years, 2018 and 2023, although the training was carried out using only the data from the year 2023. There have been relevant post-COVID changes in the sector's traffic and procedures, and it was not expected the data would be homogeneous enough to allow consistent results mixing different years.

There are three datasets covering:

- GIPV, related to flight plans.
- FLOWS, related to track updates.
- ATON, related to ATCO actions.

The sector to be used for training and validating the models in this project is the so-called LECMDGU sector, whose description has been given using the coordinates that describe a polygon. In the following image, you can see the shape of the polygon in red and the trajectory in blue and purple made by an aircraft. It is an in-route sector, covering FLs from 345 to 660. Most flights go through it, as in its vertical there are only a few small aerodromes. In its vicinity, there are some much larger airports, such as Madrid or Bilbao.

EUROPEAN PARTNERSHIP
Co-funded by the European Union

Figure 3: Interest Sector: LECMDGU

CRIDA selected this sector, considering the large amount of ATCO tasks data (ATON recordings) available for training.

Additional airspace information is available and may be used in our models. Specifically, based on historical data, the sector has a collection of associated “flows” or most prevalent traffic patterns. Both 2022 and 2023 flows descriptions are available. For our model training, we used the “past” flow (that of 2022) to train the models with “real-time” 2023 data. The reason is that it would not be possible to operate in real-time with flows that are calculated for the current year; the ATCO prediction system will always need to make use of relative past flow data.

Details of the CRIDA datasets will be provided in the following subsections.

2.4.1 GIPV (flight plans)

This dataset represents the flight plans corresponding to each flight, including all updates made along the execution of this flight. Each flight plan is described by a set of waypoints planned to be flown over in order. Waypoints can be added, removed, or completely updated along time. The flight plans include future waypoints (with their expected times of overfly) and the collection of past waypoints (with actual times). Table 2 depicts the different information available for each waypoint in a flight plan.

Value	Description
eventId	Numerical value that represents, for each flight plan associated with an aircraft, which flight plan update it belongs to. The first time the flight plan is sent, the eventId is 1 and as updates are sent, the eventId becomes 2, 3, etc.
dateFrom	Date representing the initial time of the time window to which the flight plan update that is being observed belongs.
dateTo	Date representing the end time of the time window to which the flight plan update being observed pertains.
processDateReference	All values assigned as NaN
Region	String representing the region where the data is taken: in this case it is from Madrid FIR
callsign	String representing the aircraft identifier according to the type of aircraft it is in relation to the manufacturing company or aircraft type (IBK20C, TVF63HL etc.). A single aircraft can make several flights, so it is not a unique flight identifier, but rather an aircraft identifier.
adep	String representing the airport of origin of the flight.
ades	String representing the airport of destination of the flight.

Value	Description
flightPlanEventType	String representing the type if event related to the flight plans.
state	String representing the type of state in relation to the flight.
stateDescription	String representing the description for the type of state in relation to the flight.
waypointName	String representing the identifier of the waypoint.
waypointLatitude	Float representing the latitude of the waypoint.
waypointLongitude	Float representing the longitude of the waypoint.
level	Float representing the level assigned to the flight when passing over a waypoint.
calculatedCrossingLevel	Numeric value with NaN for most waypoints.
speed	Float representing the speed assigned to the flight when passing over a waypoint.
waypointOrder	Number of the order in which the waypoint within the flight plan.
waypointSubOrder	Number of the sub-order in which the waypoint is to be passed within the flight plan order
waypointFlightRulesCode	-
waypointRouteType	String representing the type of waypoint according to its use within the sector and routes
eto	Estimated Time Over significant point.
etoTypeCode	-
airCraftRegistration	-
aircraftType	-
wake	Wake vortex: Classification of the type of generated wake vortex.
etot	Date representing the Estimated Take-Off Time
eldt	Date representing the Estimated Landing Time
iobt	Date representing the Initial Estimated Off-Block Time
HFIR	String representing the human factors impact rating.
flightRulesCode	String representing the establishment of flight regulations rules.

EUROPEAN PARTNERSHIP
Co-funded by the European Union

Value	Description
flightRulesDescription	String representing the description of the codes for the flight Rules Code
cruiseSpeed	Float representing the cruising speed of the aircraft when flying over the described waypoint
rfl	-
regulationCode	String representing a regulation code.
flightKey	Number representing the identifier for the flight, this identifier is unique for each flight and relates the flight with the rest of the datasets.
flightCode	Number representing other kind of identifier.
flowsFlightKey	Number representing other kind of identifier.

Table 2: GIPV dataset waypoint associated data

The whole GIPV dataset is composed of these waypoints' rows, which can be filtered out considering (flightKey, eventid) to obtain each individual flight plan update. The collection of waypoints will then be ordered along flight making use of the associated order and suborder fields.

The database covers the whole year of interest.

2.4.2 Flows (tracks)

This dataset contains recordings of radar tracks, quite accurate estimates of real-time positions and kinematic information of flights. There are time-stamped track updates for each flight every 5 seconds. The fields of this dataset are summarized in Table 3.

Value	Description
flightKey	Number representing the identifier for the flights only for the flows dataset.
callsign	Callsign of the flight
adep	String representing the airport of origin of the flight.
ades	String representing the airport of destination of the flight.
aircraft	-
instant	Timestamp representing the instant at which the information of the flight has been updated.

lat	Float representing the latitude of the aircraft position
Lng	Float representing the longitude of the aircraft position
modo_c	Float representing the current Flight Level of the aircraft
ac	-
vel_z	Float representing the speed of the aircraft in the vertical axis
gipvCanariasFlightKey	Number representing the identifier for the flight inside the insular (Canary Islands) territory.
gipvMadridFlightKey	Number representing the identifier for the flight inside the peninsular territory, this identifier is unique for each flight and relates the flight with the rest of the datasets.

Table 3: Flows track update associated data

The complete database contains all track updates of all flights in the whole Spanish airspace, although the relevant data (that of flights going through the interest sector) has been filtered for their use in the models training.

The database covers the whole year of interest.

2.4.3 ATON (ATCO task)

The ATON database contains recordings of these actions basically in the form of a table with a timestamp, the task type identifier, and a flight identifier. As previously described, they are related with interactions between aircraft and air traffic control officers (ATCOs), following CRIDA taxonomy summarized in Table 1. The complete format for each of the records is summarized in Table 4.

Value	Description
Time	Date representing the time instant at which the entry of the task is taken.
Name	String representing the name of the task performed
State	-
callsign	String representing the identifier of the aircraft, it corresponds to the same value as the callsign of the track (Flows) dataset
sector	String representing the name of the sector in which this intervention is carried out. In this case, the sector is always the same: LECMDGU, since it is the sector under study

Table 4: ATON ATCO task record

In some cases, there is additional information recorded (i.e. related to conflicts), but this information has not been exploited by our models.

The database does not cover the whole year of interest. Instead, there are recordings covering all the tasks in one-hour intervals. Typically, there is only one recording (1 hour) per day, and not for all days. A total of 319 one-hour recordings is available for 2023.

2.4.3.1 ATON Data Analysis

In this subsection an initial ATCO task database analysis will be performed. For this analysis flights fully traversing the sector in a recorded hour were used. Also, only flights under ATCO control at those time intervals are included here.

There are a total of 12011 flights in this situation in the ATON recordings for 2023. In Figure 4 you may see a histogram with the number of those flights that had an associated action of any given type.

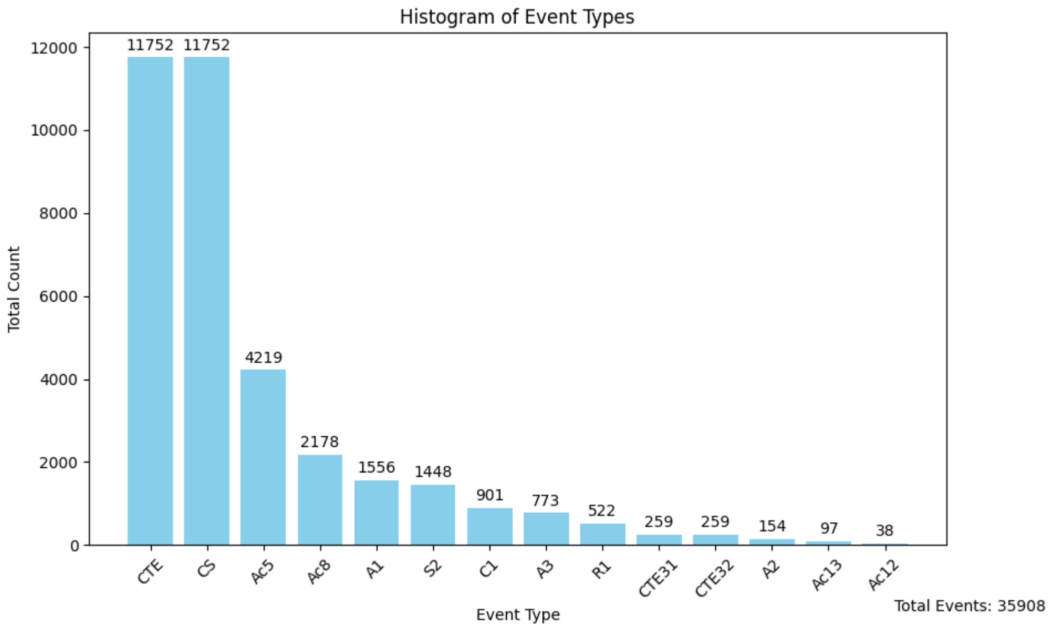


Figure 4: Histogram of ATCO actions

A set of observations can be extracted by inspecting this histogram:

- Half of the types of tasks (14 out of 28) have no associated action/event recorded at all.
- The total number of actions of the types CTE+CTE31, and the number of actions of type CS+CTE32 are equal to the number of flights (12011). In fact, most of the flights have actions CTE (initial contact) and CS (final contact), and only a few (259 in our recording) follow the alternative CTE31 and CTE32 pattern for start and end of the associated events recording.

- All other actions are much less prevalent. The following histogram shows one third of the ATCO actions are initial contacts, another third are final contacts/releases, and the remaining third encompasses all the rest of the interventions.

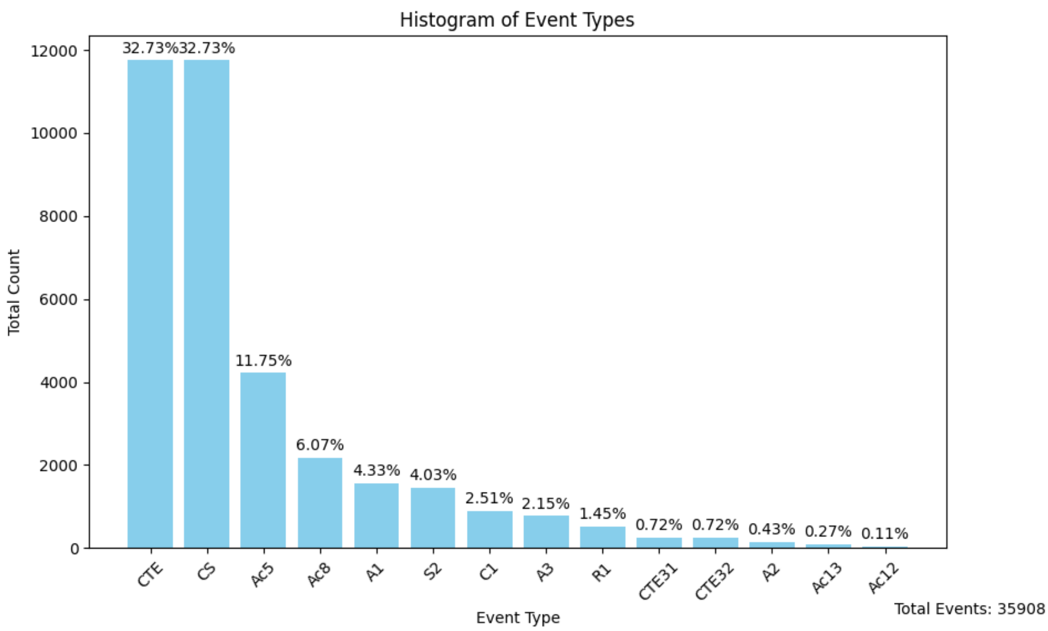


Figure 5: Percentage of ATCO actions per type

The same data, with a different normalization, gives us the percentage of flights that have an intervention of any given kind.

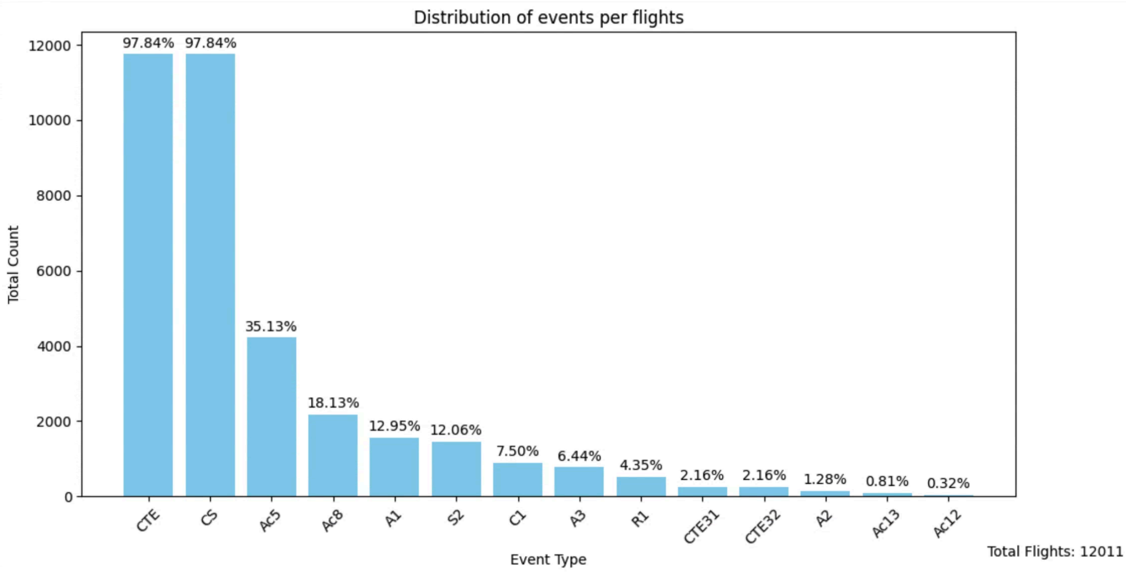


Figure 6: Percentage of flights with associated ATCO actions

EUROPEAN PARTNERSHIP
Co-funded by the European Union

It can be seen in the results that, apart from the in-out events, the other most relevant tasks are those related to:

- Procedural level changes (Ac5): more than one third of flights.
- Direct routes (Ac8), almost 20% of flights.
- Level changes (A1), Vectors (S2) for separation or sequence, just above 12% of flights. Directs (A3) for this same reason are nearly 6%.
- Transitions to Flyover (C1), or arrival (R1) are around 7% and 4%, respectively.

The rest of the actions are much less prevalent (in the order of 1%), and therefore much less predictable.

EUROPEAN PARTNERSHIP
Co-funded by the European Union

3 First Iteration of CODA ATCO Task Prediction Model Design

In the next section a first approach for the prediction of ATCO tasks is described. It tries to predict all the previously mentioned ATCO tasks for each individual flight.

3.1 CODA ATCO Task Prediction Architecture (v1)

The architecture designed for predicting ATCO tasks comprises four stages, represented in figure 7: the first stage focuses on data ingestion and filtering, the second on label generation, the third on task prediction, and the last stage on task scheduling forecasting.

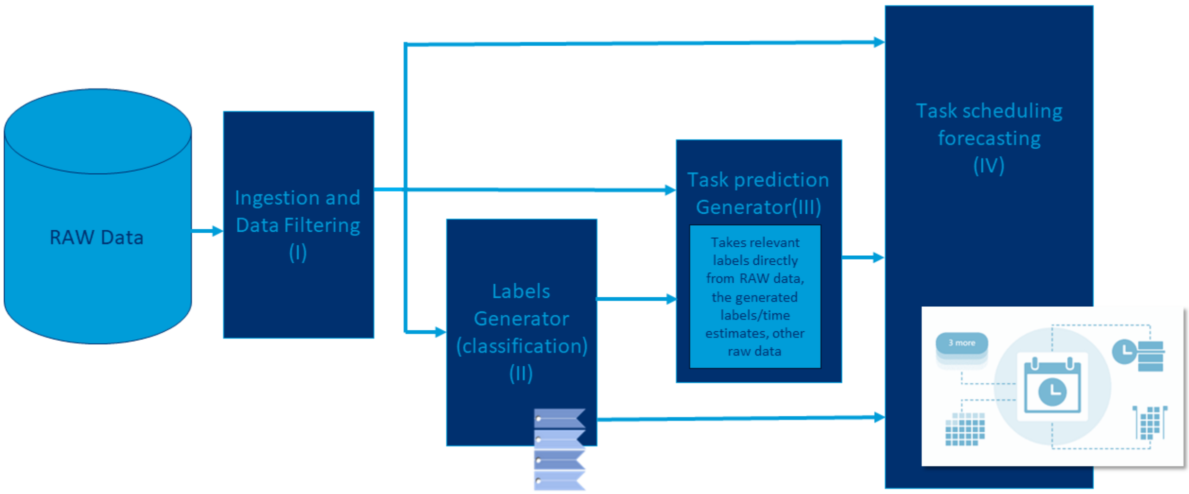


Figure 7: Prediction architecture

The proposed architecture has four phases, to be described in detail in the following subsections.

3.1.1 Ingestion and data filtering

A filtering process is performed on the data to be used for each iteration. It should be noted that a sliding time window will be employed; therefore, temporal filtering of the data will be necessary for each moment this analysis is conducted. Additionally, only flights passing through the sector will be extracted from the dataset.

3.1.1.1 Data Fusion

During the ingestion and data filtering operations, a merge is performed between the raw datasets of tracks, flight plans, and ATCO tasks. Analysing the dataset after this merge, it can be observed that most flights do not have any associated tasks. Only 319 hours of tasks are available, while complete days of tracks and flight plans are available corresponding to those

EUROPEAN PARTNERSHIP
Co-funded by the European Union

319 hours. Most tracks and flight plans are associated to time intervals without ATON recordings and therefore need to be discarded for the training and validation of our model: for each hour with tasks, there are almost fifteen hours of track and flight plan data. The data fusion process discards the not relevant data, makes use of the flight keys and aligns the time formats.

3.1.2 Labels generator

The second stage of the system is referred to as the label generator. These labels will describe flight behavior by comparing flight plans and their tracks. Below is a brief description of each of these labels, which aim to classify the flights. The global aim of this step is synthetizing a collection of features to be able to learn a ML model to predict the occurrence of tasks.

3.1.2.1 ATCO responsibility

This label indicates whether the aircraft falls under the jurisdiction of the air traffic controller (ATCO). Two checks are conducted to determine this: firstly, to confirm if the aircraft is located within the designated sector, and secondly, to assess whether the ATCO has engaged with the aircraft in any manner thus far. Depending on both criteria, the label can take on the following values: INSIDE - UNDER RESPONSIBILITY, OUTSIDE – UNDER RESPONSIBILITY, INSIDE – NO CONTACT, or NO RESPONSIBILITY. Additionally, within this set of labels, the status of interaction transmission, denoted as CTE_SENT, is also included: True/False.

To conduct these verifications, the following data sources are utilized:

- The position designated by the latest trajectory transmitted by the aircraft, determining its sector presence.
- The ATON dataset, logging interactions between the aircraft and the ATCO.

3.1.2.2 Flight Plan (FP) non conformance

This label will facilitate comparison between the current path being followed by the aircraft during the flight and the detailed planned route outlined in the flight plan. To achieve this, the following metrics will be analyzed (and compared with ad-hoc thresholds):

- Lateral displacement: It will assess how much the current route deviates laterally from the planned trajectory.
- Route delay: It will measure the time deviation of the aircraft from the anticipated position on the planned route.
- Aircraft altitude deviation: A comparison will be made between the altitude of the aircraft and the specified altitude limits for the current position on the planned route.

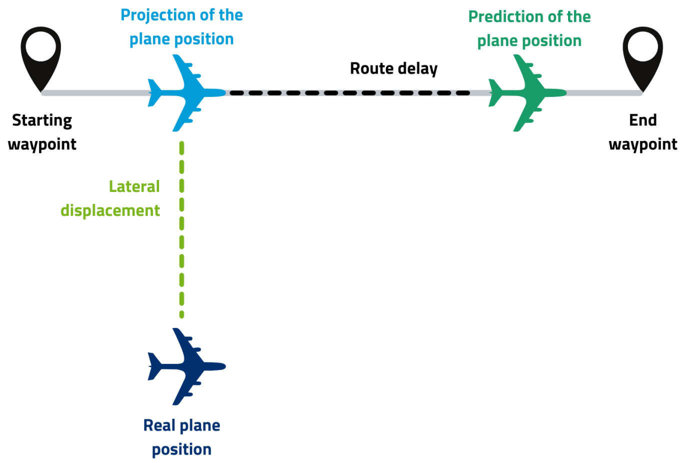


Figure 8: FP conformance classification

3.1.2.3 Delayed flight

This label will provide information on the flight delay in relation to the initial flight plan and the last flight plan updated before take-off. This will be achieved by comparing the expected arrival time at the next waypoint, as indicated in the flight plan, with the planned arrival times at that same waypoint both when creating the flight plan and just before take-off. Additionally, it will indicate if the flight is still on the ground with a marker. The label does not consider the reason for the delay (no consideration of network management and ATFCM is included here).

3.1.2.4 Departure delay

This label displays the departure delay of a flight by comparing the scheduled departure time in the initial flight plan created and the latest update of the flight plan, which presents the actual departure time (if the flight has already taken off) or the expected departure time if the flight has not yet started.

3.1.2.5 Unexpected flight pattern

This label will assess the presence of abnormal behaviors by comparing the current flight behavior with thresholds for maximum and minimum speed, as well as maximum and minimum altitudes. This evaluation will only be conducted for flights within the sector, excluding those that have not yet entered the sector or have already exited it.

3.1.2.6 Aircraft in flow group

This label will identify whether an aircraft belongs to a group of aircraft that are following a common route, close together on the same flight leg, on the previous leg or on the next leg. It calculates the distance between the aircraft and other aircraft on the same legs, as well as on the preceding and following legs. If it finds at least three aircraft close enough, it forms a group and provides information about the aircraft in the group. This function is useful for air traffic analysis and management, helping to identify groups of nearby aircraft for coordination and safety purposes.

3.1.2.7 Congested waypoints

This tag allows to identify the congestion of waypoints in an air traffic sector during a specific time interval. In this way it is possible to know if an aircraft passes through waypoints that are congested.

3.1.2.8 Detect belonging to high traffic flow

This label will determine, both for flight plans and tracks, if the aircraft is following one of the high traffic flows described in section 2.4. This analysis will also check which of the waypoints that form part of the routes defined in the flight plans have been skipped or added within the predetermined sector routes.

3.1.2.9 Congested airspace

This label will serve as a marker for the level of airspace occupancy for the current moment and for future time windows. To achieve this, the airspace is divided into zones or cells as shown in the image. Then, the following data is collected:

- The number of aircraft in each cell (grid) based on flight tracks.
- The number of aircraft that should be in each cell according to flight plans.
- The number of aircraft expected in each cell for each defined future time window according to flight plans.

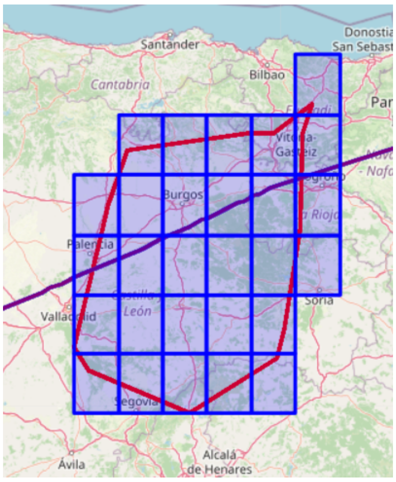


Figure 9: Cells for congested airspace assessment

3.1.3 Task Prediction Generator

The core element of the system is the Task Prediction Generator. The schematic that best describes the task prediction generator is the one presented in Figure 10.

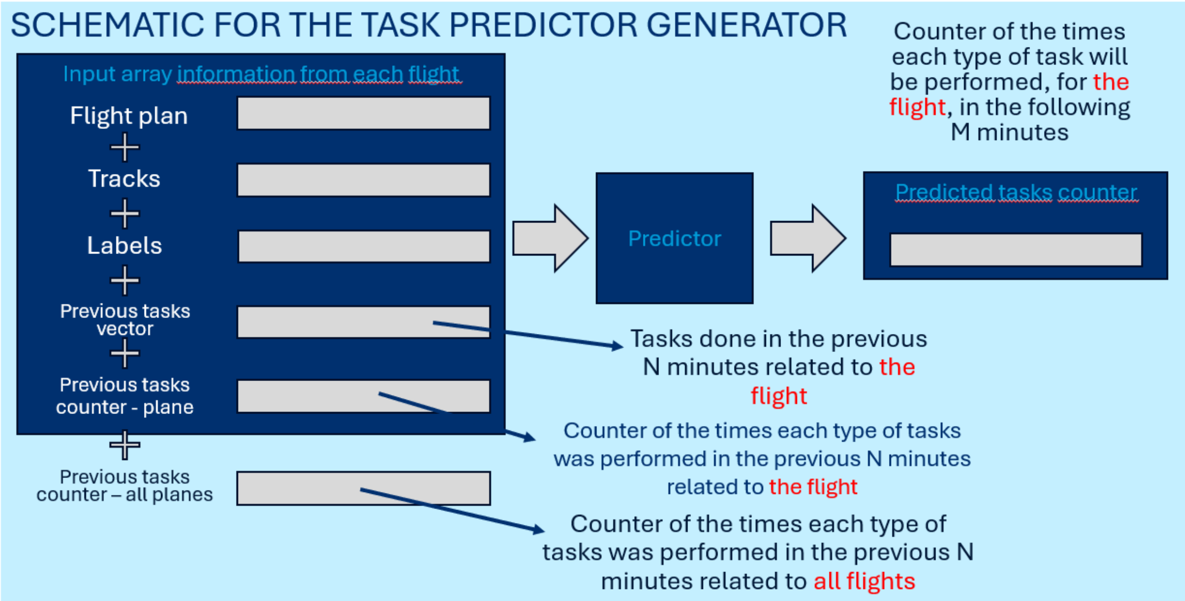


Figure 10: Task prediction generator inputs and outputs schema

ATCO task prediction generator is, therefore, a function which should relate the information of each individual flight and associated labels with the number of occurrences of each individual type of tasks for a short interval in the future. The inclusion of previous tasks for the flight as an input should ideally preclude the occurrence of tasks already performed, while the inclusion of previous tasks for all aircraft would somehow reflect the overall workload and the most typical task types being used by the controller in the current situation.

To implement this function a collection of Machine Learning (ML) approaches to be discussed later have been used. All those ML approaches have been trained using CRIDA database, which has been processed using a sliding window approach to synthesize pairs input-output to enable the use of supervised training approaches.

3.1.3.1 Task Prediction Dataset Generator

To generate the data associated with each flight, it is necessary to extract the labels that describe the flight at the specific moment being analyzed. This will involve evaluating aircraft that are currently or were recently within the sector during the time window for which the labels are being extracted, as well as aircraft that, according to their flight plans, will be within the airspace during the subsequent moments. These time windows, along with the windows for which predictions are to be made, are represented in the following figure.

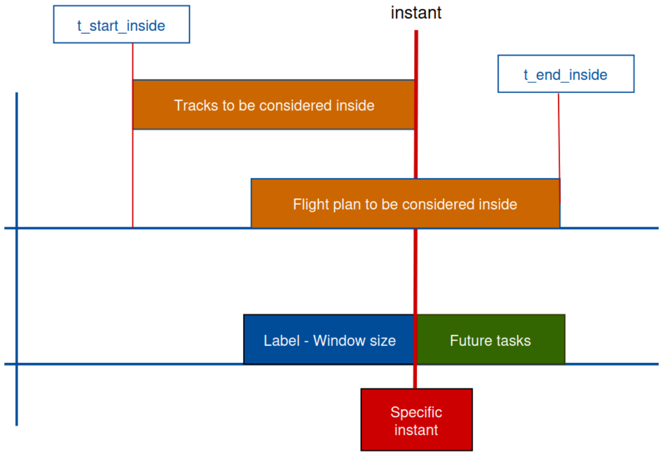


Figure 11: Sliding windows for task prediction generator training

Additionally, this iterative analysis will be performed for each of the time windows for which information about the air traffic controller's tasks is available in the dataset provided by CRIDA. These windows will be analyzed by shifting the times with an overlap with the previous window, as shown in the following image.

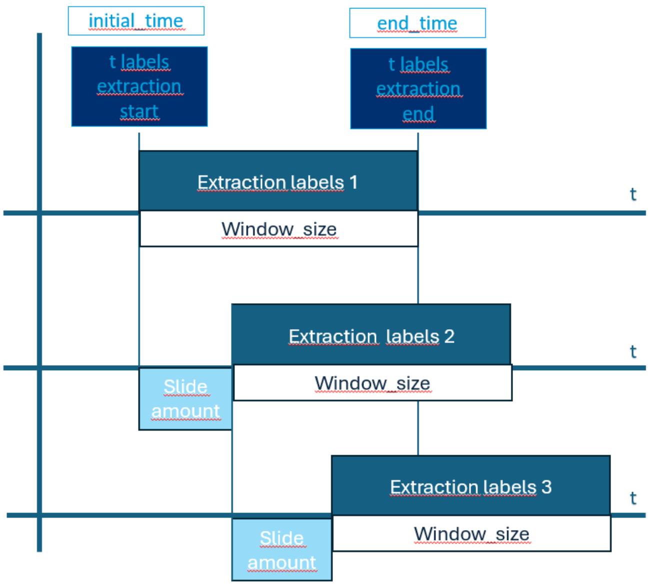


Figure 12: Sliding windows for Task prediction generator training process

After analyzing each flight and obtaining the values of the labels that represent the flight's behavior for the current interest instant, a dataset is generated to predict the number of tasks of each kind that will be performed, associated with a flight, at a specific interval in time.

As a result of this process, a dataset is obtained that contains, for each flight and each time window, a set of input values, called the input vector, with all the real-time flight information (labels plus additional information), and another output vector that contains the number of times a task associated with the flight in question will be performed. The output vector will be the target aimed for the prediction, and the function will be fitted following a supervised training approach.

In our initial implementations of the task prediction generator, we aimed to train a predictor capable of providing the number of occurrences of each individual task type in the CRIDA taxonomy, which resulted in an extremely time-consuming training process with very bad results.

To increase the number of occurrences of each type of tasks and try to improve the previous results it was decided that instead of treating each task type independently, a group of functionally similar tasks would be predicted. Thus, the task information has been grouped with the following structure:

Tasks	Group
CTE + CTE31	Radar Contact
CS + CE32	Flight Released
Ac5 + Ac6 + Ac7 + Ac8 + Ac9 + Ac12 + Ac13	Procedures
S2 + S3 + S4	Vector and weather deviations
A1 + A2 + A3 + A4 + A6	Separation
Co2 + Co5 + Y1 + Y2 + Y3	Coordination

Table 5: Groups of ATON ATCO task types

R1 (transition to arrival) and C1 (transition to flyover) were kept separate, as they appear as tasks in the dataset but are not so closely related to the other groups.

3.1.3.2 CODA ATCO Task Prediction Inputs and outputs

The task prediction inputs that are presented in Figure 10 contains all the information related to the state of the flight at the time of the analysis. In Table 6 we can find the inputs and the outputs used for training and their meaning.

Input	Meaning
window_instant	Instant identifying the window (current time)
flightKey	Unique flight identifier
callsign	Aircraft identifier
actual_lat	Aircraft latitude at the current time
actual_lng	Aircraft longitude at the current time
actual_modo_c	Aircraft mode C at the current time

Input	Meaning
actual_velocity	Speed on the z-axis at the current time
height_pattern	Information about the height pattern based on the vertical movement of the aircraft. Label taken from unexpected flight pattern
lateral_displacement	Information about the lateral displacement of the aircraft with respect to the expected position according to the flight plan. Label taken from Flight plan non-conformal
time_to_intersect_1	Time until entry into the sector (>0 if not entered, <0 if already entered)
time_to_intersect_2	Time until exit from the sector (>0 if not exited, <0 if already exited)
initial_delay_time	Delay respect to the first flight plan that was updated after the take off
departure_delay_time	Delay time with respect to the original flight plan (if there has been a departure delay)
state_delay	Delay status, taken from the label delayed flight
time_delayed_time	Delay time with respect to the flight plan
in_contact	Flight in contact with the ATCO
inside	Flight is within the sector in area (latitude and longitude)
initial_contact_sent	The initial contact task has been performed
scheduled_flow_1	Which Flow from the flows file is being followed according to the flight plan
added_wp_1	Have WPs been added to the flight plan compared to the Flow followed?
skipped_wp_1	Have WPs been skipped in the flight plan compared to the Flow followed?
followed_flow_1	Which Flow has been followed according to the tracks
real_position_real_cell_congestion	Number of aircraft in the cell where the flight is located

Co-funded by the European Union

Input	Meaning
real_position_expected_cell_congestion	Number of aircraft that should be in the cell where the flight is located
expected_position_real_cell_congestion	Number of aircraft that should be in the cell where the flight was expected to be
actual_cell_lat	Cell where the flight is located
actual_cell_lng	
expected_cell_lat	Cell where the flight was expected to be
expected_cell_lng	
future_cell_lat_X	Future cells where the aircraft is expected to be according to the flight plan
future_cell_lng_X	
future_cell_congestion_X	Congestion expected in future cells
congested_wp_X	Set of waypoints exceeding a threshold of the maximum number of aircraft near them
total_congestion	Total airspace congestion
AAA_allplanes_done	Number of times task AAA[3] has been repeated for all aircraft within the airspace during the window_size time
task_done_N	Task number N performed for the flight under analysis
time_for_task_done_N	Time at which task number N was performed
AAA_done	Number of times task AAA[4] associated with the flight under study has been performed
Output	**Meaning**

[3] Here, AAA is a placeholder for each type of task. So, AAA will be CTE, CS, Ac5, … and there would be up to 24 features of this kind. Also, when we group types of tasks, it will be the associated feature for all the associated task types.

[4] Again, AAA is a placeholder for each type of task. So, AAA will be CTE, CS, Ac5, … and there would be up to 24 features of this kind. Also, when we group types of tasks, it will be the associated feature for all the associated task types.

Input	Meaning
AAA_this_plane	Number of times task AAA[5] will be performed in the following minutes defined by the delta_t_future_task window

Table 6: Data for generator training

It should be noted some of the last data in Table 6 are basically vectors, as there are individual counters for each task type. Depending on the iteration, we have typical input vectors of dimensionality 80-130 and output vectors of dimensionality 8-28 (with or without task types grouping).

3.1.4 Task Scheduling Forecasting

In the last phase of the system, the models should, ideally, predict the time instants in which the ATCO will need to perform the predicted tasks. This system would receive as inputs the outputs of the Task Prediction Generator (the predicted tasks for each aircraft), together with the additional tags mentioned above. Based on these inputs, it would analyze the predicted tasks and predict the time interval in which the ATCO would perform each task.

3.1.5 CODA ATCO Task Prediction Outputs

Ideally, the CODA ATCO Task Prediction outputs include a list of tasks (events) per flight, and for each task, a probability associated to that task, a timestamp and a range of time. The prediction is done for each flight and for each time window. To simplify the problem, the outputs have been divided into two parts: task prediction (+probability) and time (+range). A first model predicts the task and a second one the time.

The predictions of timing and time ranges were also affected by the poor performance of the task predictor, and the model currently assumes the window time and duration as the temporal reference for this part of the prediction output.

3.1.6 CODA ATCO Task Prediction Intermediate Outputs

The CODA ATCO Task Prediction model generates several intermediate outputs during its processing pipeline, which can be valuable for system monitoring, debugging, and further analysis. These intermediate outputs include the processed and filtered dataset resulting from the initial data ingestion stage, containing all the labels generated from the initial datasets. From this stage, it is possible to extract additional information such as tasks that never occur, labels that are identical throughout the dataset, or unexpected data types, which is useful for debugging the label generator.

[5] Again, AAA is a placeholder for each type of task. So, AAA will be CTE, CS, Ac5, … and there would be up to 24 features of this kind. Also, when we group types of tasks, it will be the associated feature for all the associated task types.

Another intermediate outputs are the intermediate feature importance scores, indicating which input variables have the most significant impact on task predictions. These scores have been used to refine the model.

3.2 CODA ATCO Task Prediction Generator Model Implementation (v1)

The overall system design was implemented in Python, and a collection of ML models were trained to try to interpolate a function relating the described inputs and outputs, described in sections 3.1.3, and more in detail in section 3.1.3.2.

To be able to have comparative results different approaches from the literature and with reference Python implementations were analysed for initial implementation. Their main features are summarized in next section, and they are later described. Later in the process, and after testing quite a few of the initially selected algorithms other approaches where tested. Those with better performance along the different iterations were kept for further refinement, and therefore this initial list is not completely aligned with the final ones.

3.2.1 Initial Applicable ML models

Table 7 lists the potentially applicable ML models and analyses them comparatively with regards to some attributes of interest for our application.

The selected models, such as those based on regression, decision trees and neural networks, have been chosen trying to find a balance between accuracy, explainability and scalability.

Input	Entry type	Computational cost	Explainability	Precision	Scalability	Interpretability	Robustness at noisy data
Linear regression	Numerical	Low	High	Medium	High	High	Medium
Logistic regression	Numerical/ Categorical	Medium	Medium/High	Medium	Medium	Medium/High	Medium
Decision trees	Numerical/ Categorical	High	Medium/High	High	Medium/High	High	Medium
Random Forest	Numerical/ Categorical	High	Medium/High	High	Medium/High	High	Medium
SVM	Numerical	High	Medium	High	Medium/High	Medium	High
Neural Networks	Numerical/ Categorical	Very High	Low	Very High	High	Medium	Medium/High
kNN	Numerical/ Categorical	Low	Low	High	Low/Medium	Low	Low
Naive Bayes	Numerical/ Categorical	Low	High	Medium	High	High	High
Transformers	Numerical, Binary, Categorical, etc.	Very High	High	Very High	High	High	Very High
LSTM	Numerical, Binary, Categorical, etc.	High	High	High	High	High	High

Table 7: ML approaches comparison

Description of the algorithms.

- Linear and logistic regression: These algorithms are widely used due to their low computational cost and high interpretability. Linear regression is suitable for continuous prediction problems, while logistic regression is effective in binary or multiple classification scenarios. While their accuracy may not be the highest compared to more complex models, their simplicity and interpretability make them valuable in situations where explainability is crucial for human operators.

- Decision Trees and Random Forest: Decision trees are high-performance models that divide data into multiple branches, facilitating the prediction of events from categorical and numerical data. Their ability to be easily explainable makes them an effective tool in the prediction of air traffic control tasks. The Random Forest combines several decision trees to improve accuracy and robustness to noise in the data. These algorithms are suitable for high accuracy prediction, although at a higher computational cost.

- Support Vector Machines (SVM): SVM is a powerful model for real-time task classification. Its accuracy and scalability are strong points; however, the model requires a high computational cost, which could be a limitation in real-time implementations in systems with limited resources.

- Neural Networks and Transformers: Neural networks and transformers are notable for their ability to handle large volumes of data and detect complex patterns. These models are particularly useful in scenarios where high accuracy is sought, such as predicting multiple simultaneous air traffic control tasks. However, they require large amounts of data and considerable computational resources, which can make them difficult to use in real time in certain systems.

- Naive Bayes and k-Nearest Neighbours (kNN): Naive Bayes and kNN are fast and efficient algorithms with low computational cost. However, although they are more suitable for real-time implementations due to their simplicity, their accuracy and ability to handle noise in data are often limited compared to more complex models.

- LSTM (Long Short-Term Memory): LSTM neural networks are particularly useful for sequential and time-dependent data, which makes them suitable for prediction of time-series based tasks such as tracks and flight paths. Although their performance is high, they also have limitations in terms of computational cost and the need for large volumes of historical data.

Limitations of the algorithms.

There are some important limitations of these approaches that must be considered. First, many of the more accurate algorithms, such as neural networks and transformers, require large amounts of historical data to adequately train the models. In scenarios where data is limited or incomplete, these algorithms may not perform optimally. In addition, algorithms such as SVM and Random Forest, although accurate, may present real-time scalability problems due to their high computational cost. Finally, simpler models, such as Naive Bayes and kNN, while lightweight and fast, can suffer in situations with high levels of noise or complexity in the data, resulting in less accurate predictions. These factors underscore the importance of carefully selecting the appropriate algorithm.

3.2.2 Datasets for CODA ATCO Task Prediction ML models training

The datasets used for the training includes all the features described in section 3.1.5, so it directly or indirectly includes all the information coming from radar tracks, flight plans and ATCO tasks. The models have been trained using 2 datasets: the first one generated with a time window of 20 minutes and the second one with a time window of 10 minutes. The datasets have been divided into 85% for training and 15% for validation for all the training session to guarantee homogeneous result and making possible a comparison of the models.

3.2.3 Training process of CODA ATCO Task Prediction ML models

Various algorithms were evaluated in the training process, with those less suited to the specific requirements of our case, such as linear regression, logistic regression, KNN, and SVM, being discarded in the early stages. The focus then shifted to more promising algorithms, including decision trees, random forests, XGBoost, LSTM, and transformer models. For each of these, an extensive hyperparameter tuning process was conducted to optimize their performance and achieve the best possible results. Different modelling approaches were also explored, including the creation of a single multi-output model, as well as experimenting with multiple single-output models to assess whether specialized models could provide better accuracy or generalization.

EUROPEAN PARTNERSHIP
Co-funded by the European Union

3.3 CODA ATCO Task Prediction Model Validation (v1)

3.3.1 Datasets for CODA ATCO Task Prediction Model validation

The datasets used for validating the CODA ATCO Task Prediction Model are the same as those described in section 3.2.2 for training. For the validation 15% of the datasets have been reserved as unseen data. An early stop callback has been used to prevent overfitting and to accelerate the training of all the models created.

3.3.2 Validation results of CODA ATCO Task Prediction Model

The validation dataset has been used to evaluate the model's performance across various metrics such as total accuracy, probability of generating a task when the task occurred in the dataset, and the rate of false alarms. It should be reminded a value 0 in a certain component of the prediction vector means no task of this type is performed/predicted, while a value different than zero means there has been a real or predicted task of this kind.

The metrics are defined and calculated in the following manner:

- **Accuracy**: % of times the model correctly predicts the value of the output vector (number of predicted tasks is equal to number of actual tasks).

$$\frac{\sum \text{correct predictions}}{N \text{ total cases}}$$

- **Probability of correct task prediction**: % of times that the model correctly predicts that the value in the output vector is different from zero (number of predicted tasks and number of actual tasks are greater than zero, so the system can predict an actual task).

$$\frac{\sum \text{ times it predicts } > 0 \text{ when it is } > 0}{\sum \text{ times the actual value is } > 0}$$

- **Probability of false task prediction**: % of times the model predicts that there is going to be a task when in fact there is no task.

$$\frac{\sum \text{times it predicts } > 0 \text{ when it is } 0}{\sum \text{times the actual value is } 0}$$

During the training of these initial multi-output models, several algorithms were discarded due to poor performance, which stemmed from their unsuitability for the type of problem at hand. The poor results in predicting tasks when they occurred (just over 0% accuracy in the cases of R1, C1, and RC) led to the decision to split the multi-output model into several single-output models. To further improve accuracy, although our initial prediction window was of 20 minutes, we obtained better results by reducing it to 10 minutes. The results are reported in tables Table 8, Table 9, Table 10 and Table 11 , which contain the final list of algorithms being tested.

		Decision Tree	Random Forest	XGBoost	LSTM	Transformer
CTE	Accuracy	86.1%	90%	93.3%	92.8%	37.1%
	Probability of correct task prediction	59.8%	60.1%	72.3%	79.8%	45.6%
	Probability of false task prediction	4.9%	5.5%	2.3%	3.2%	53.4%
CTE31	Accuracy	99.2%	99.4%	99.6%	99.8%	78.9%
	Probability of correct task prediction	56.8%	34.6%	52.5%	83.1%	13.6%
	Probability of false task prediction	0.03%	0.1%	0.05%	0.05%	20.5%
CS	Accuracy	91.9%	93.5%	96%	92.7%	71.7%
	Probability of correct task prediction	85.8%	76%	87.9%	81.7%	8.4%
	Probability of false task prediction	1.1%	3.4%	2.8%	2.5%	8.6%
CTE32	Accuracy	92.7%	99.4%	99.7%	99.8%	69.5%
	Probability of correct task prediction	2.3%	40.2%	65.1%	87.7%	22.3%
	Probability of false task prediction	6.7%	1.5%	0.04%	0.09%	29.8%

Table 8: ATCO Contact and Release Tasks Prediction with different ML approaches.

		Decision Tree	Random Forest	XGBoost	LSTM	Transformer
Ac5	Accuracy	96.4%	97.1%	98.1%	98.5%	95.5%
	Probability of correct task prediction	39.7%	35.9%	56.5%	76.2%	10%
	Probability of false task prediction	0.6%	0.9%	0.6%	0.6%	0.9%
Ac8	Accuracy	97.3%	97.726%	98.4%	98.8%	38.4%
	Probability of correct task prediction	33.3%	26%	44.5%	70.01%	58%
	Probability of false task prediction	0.06%	0.5%	0.2%	0.4%	60%
Ac12	Accuracy	100%	100%	100%	100%	78.4%
	Probability of correct task prediction	0%	100%	100%	100%	1%
	Probability of false task prediction	100%	0%	0%	0%	22%
Ac13	Accuracy	99.9%	99.9%	99.9%	99.9%	83.6%
	Probability of correct task prediction	12.4%	50%	72.2%	50%	11.4%
	Probability of false task prediction	0.01%	0.2%	0.02%	0.01%	16.3%

Table 9: ATCO Procedure Task Prediction with different ML approaches.

EUROPEAN PARTNERSHIP
Co-funded by
the European Union

		Decision Tree	Random Forest	XGBoost	LSTM	Transformer
A1	Accuracy	99.1%	98.5%	99.1%	99.2%	62.2%
	Probability of correct task prediction	48.9%	25.4%	48.9%	70.8%	20.1%
	Probability of false task prediction	0.03%	0.5%	0.03%	0.2%	26.1%
A2	Accuracy	99.8%	99.8%	99.8%	99.9%	73.7%
	Probability of correct task prediction	32.6%	18.6%	32.6%	45.8%	23.6%
	Probability of false task prediction	0.03%	0.3%	0.03%	0.01%	26.1%
A3	Accuracy	99.3%	98.9%	99.3%	99.5%	81.2%
	Probability of correct task prediction	46%	21%	46%	74%	17.1%
	Probability of false task prediction	0.01%	0.3%	0.01%	0.01%	17.9%

Table 10: ATCO Separation Task Prediction with different ML approaches.

		Decision Tree	Random Forest	XGBoost	LSTM	Transformer
R1	Accuracy	99.4%	99.4%	99.5%	99.8%	51.5%
	Probability of correct task prediction	61.8%	19.1%	36.8%	72.8%	58.2%
	Probability of false task prediction	0.01%	0.05%	0.03%	0.09%	48.2%
C1	Accuracy	98.8%	98.3%	98.9%	99.1%	68.2%
	Probability of correct task prediction	52.3%	39.3%	56.1%	75.5%	24.9%
	Probability of false task prediction	0.2%	0.04%	0.02%	0.3%	30%

Table 11: Rest of ATCO Task Prediction with different ML approaches.

It can be observed in the results table that, even though none of the tested models offers strong enough metrics to be considered as a reliable predictor, there are interesting observations to consider. First, the group of tasks with better results are the contact and release tasks, which are not only the most common ones, but also the easiest to explain in a deterministic way. In fact, they are some of the lowest priority tasks. When it comes to predicting less common and more complex tasks, such as procedure and separation tasks, the performance metrics drop quite significantly. In the case of **LSTM**, decent results are depicted, reaching values in the range from **70%-80%** for difficult tasks to predict such as **Ac5, Ac8, A1** ..., with false alarms ratio under **1%**.

The overall scores do not provide a result good enough to be used in a final system, but **LSTM** sheds light on the matter with a somehow promising result which leads to the idea that it could be feasible to develop in the future a model strong enough to be put in the real system. However, this method (**LSTM**), as with most deep learning algorithms, has a limited explainability.

4 Discussion on the limitations of the first iteration of the CODA ATCO Task Prediction Model Design

The initial iteration of the CODA ATCO Task Prediction Model has revealed several limitations and areas for improvement.

Firstly, while the overall accuracy of the model exceeds 97-98%, this high value is somewhat misleading. Most of the flight's passing through the sector do not generate any tasks, effectively transforming the problem into one of anomaly detection rather than task prediction. This characteristic of the data inflates the accuracy metric, as the model performs well in predicting the more common "no task" state. Other performance metrics present a more realistic picture of the model's capabilities. Only the Radar Contact (RC) and Flight Released (FR) task groups (which are almost “deterministic”) achieve probability scores close to 90% when a task occurs, and they have a low false alarm rate. The performance metrics for other task are considerably lower, indicating room for improvement in predicting less predictable tasks. The only method that depicts somehow decent scores is the previously mentioned LSTM.

A significant limitation of the current models is the size of the available dataset. It includes almost one year of tracks and flight plans actualization, but it is limited to only 319 flight hours. The dataset is relatively small for a complex prediction task. This limited data volume may affect the model's ability to learn and generalize across a wide range of scenarios.

An alternative approach would be using a more complex environment (such as a complex TMA) leading to a higher number of interventions. Large ATCO tasks recordings were not available currently for such scenario, but this is a potential area for future research.

Another identified issue is the potentially excessive number of labels generated for each data point (flight + time). The current approach produces 110 total fields as input, which presents challenges for the models in terms of feature handling and interpretation. This high dimensionality may lead to overfitting on the training data and poor generalization to new scenarios.

Finally, the relation between the extracted features and the task predicted is not too clear, and using the same features for all the task types could lead to unnecessary noise in the inputs. Also, the joint prediction of all task types might lead to find local optima where the ML algorithm find trade-offs between different dimensions and struggles to find a common approach. Also, completing the data with other contextual information (weather, airports configuration, etc …) might be interesting, although it may lead to even higher input dimensionality.

An additional limitation of this approach is related to those interventions used for separation/sequencing/conflict resolution. Although the event leading to this intervention may have a relationship with more than one flight, the issue is quite often solved by issuing only one action allocated to one flight. There is not enough information in the recorded data to predict the action allocation from the ATCO, as the complete rules/procedures/preferences are not available. Also, there is no weather data, the configuration of the nearby sectors and airports is not available, and other contextual data which may have an impact on ATCO operations is not available.

These limitations provide some directions for future iterations of the CODA ATCO Task Prediction Model. Firstly, it is essential to identify which labels are relevant for each task type. This approach will enable the training of distinct models for individual tasks, using only the labels that effectively impact the prediction outcome for that specific task. Consequently, this method prevents other irrelevant labels from influencing the result. To achieve this objective, a more in-depth analysis of the available data and the relationships between them will be necessary. It will probably necessitate a revision of certain label generators to modify their logic. The possibility of expanding the dataset beyond 319 hours would also present an ideal condition, providing more data to work with. This expansion would potentially enhance the robustness and accuracy of the models developed for each task.

5 Second Iteration of CODA ATCO Task Prediction Model Design

Following the insights gained from the initial iteration of the CODA ATCO Task Prediction Model, a second iteration has been proposed to address the identified limitations and enhance the model's predictive capabilities. This iteration focuses on refining the approach to task prediction, aiming for more granular prediction architecture, tailored for individual different task or task types. By moving beyond the anomaly detection tendencies of the first iteration, the second iteration seeks to improve task-specific predictions, offering a more detailed and nuanced model that better captures the diversity of tasks an air traffic controller may assign to a given flight. This refined design will help ensure that the model is not only highly accurate but also better aligned with real-world task distribution and operational needs.

A key feature of this version will be to soften the requirement to predict atomic or grouped ATCO tasks, changing it by the estimation of the probability of occurrence of each specific task, tailored to each flight. By leveraging the characteristics associated with each task and flight, the model will provide predictions on how likely it is that a given task will occur in the next few minutes. The associated cost will be not having so accurate timestamping of the predicted tasks and not providing hard decisions on the appearance of these task. This would result in implications on the compatible adaptive automation strategies, but it should be clear that following the first approach we could be providing very erratic ATCO task predictions with very noisy timestamps, which would result in a CODA system not reliable and not acceptable by ATCOs.

If we assume all flights entering the sector have equal probability to "generate" each type of action, for any given flight with no additional data, we could predict, based on 2023 data, that it will have the following associated "default" probability vector (note this is just the information in Figure 5, put in vector form). Probabilities are provided as % in the vector.

Component	CTE	CS	Ac5	Ac8	A1	S2	C1	A3	R1	CTE31	CTE32	A2	Ac13
Percentage	97	97	13	11	5	2	8	5	3	3	3	1	1

Table 12: ATCO Task Prediction probabilities: default 2023 values

Each component of this vector contains the average probability that a given flight demands this type of intervention from the ATCO, which is also equal to the average number of actions of this type per flight (assuming this is a Bernouilli distribution), calculated based on the overall traffic.

An initial key idea in this new approach is predicting a ATCO task probability vector based on ad-hoc features, different from the default one presented before (which can be assumed to be an averaged value for all flights). And a second key idea is splitting the vector-oriented predictor in a collection of smaller dimensionality predictors, one per action type. Each of these predictors would have more tailored small dimensionality feature set.

This approach has some additional advantages over the previous one:

- It deals much more nicely with those types of tasks with a lower occurrence rate.

- It enables an incremental implementation, where those types of actions where a potential prediction function is identified may use this function, and if there is not enough data or the function is found default probabilities may be provided as outputs.

- As it is dealing with average probability values, it is more robust against some of the fundamental problems of the previous approach, as the allocation of actions related to several flights, or the lack of contextual information. For instance, a conflict resolution could lead to an increased probability of action for both flights involved, instead of assigning the action to either of them.

- Ad-hoc predictors contribute to reduce the **dimensionality** by using a set of a few specific indicators that are found relevant for each task. This not only lighten the models but also enhances the explainability of its behaviour.

	Approach version	
Keys	**V1**	**V2**
Output	Binary classification and timestamp for each type of task	Estimated Probability for each task or group of tasks
Number of models	1	Individual model for each task or group of tasks
Dimensionality	Large number of shared features	Reduced number of features for each task
Explainability	Low	High

Table 13: Approaches comparison

At the same time, the approach has one fundamental problem: By not issuing a list of individual tasks for each flight, it is not compatible with automation adaptation strategies demanding this list.

In any case it is compatible with automation strategies where all tasks of a given type are managed by the digital assistant or with others where individual flights are fully controlled by the digital assistant. In both cases it may enable the calculation of an average number of tasks of each kind being managed by the ATCO and the digital assistant, which may be safely used for the prediction of the ATCO workload and the real time management of digital assistant necessary computational resources. It should be noted those average probability numbers may be added for all flights to calculate the average number of actions for each type of task.

5.1 CODA ATCO Task Prediction Architecture (v2)

The initial part of the architecture remains the same. It will filter and pre-process the data to generate a set of labels/features that represent additional information extracted from the dataset. Each label generator will be analysed to understand whether its output is useful for some prediction and/or if it needs some change. The idea will be to have very reduced dimensionality inputs (i.e. dimension < 5), so that we capture the major dependencies, and we are able to train simple and explainable model.

In the final phase of the model, instead of using a single predictor for all tasks, specific predictors will be implemented for each task or type of task (trained/derived using a different collection of features). Based on those features, each of the system predictors will be able to estimate the probability of occurrence of the associated task type for a given aircraft within the next few minutes.

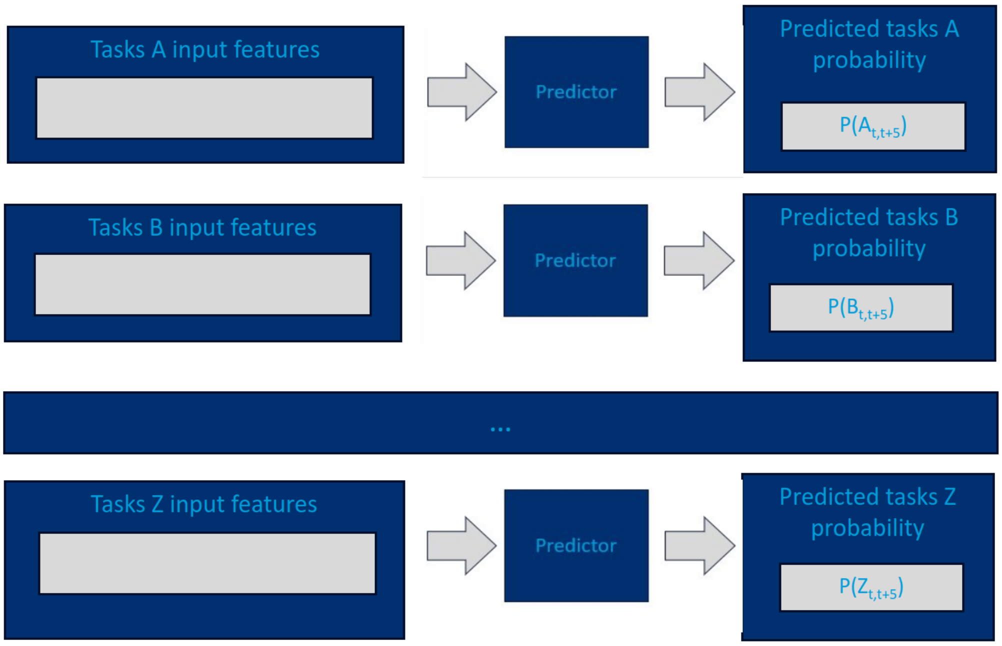


Figure 13: V2 Architecture approach

5.1.1 Ingestion and data filtering

The second iteration of the CODA ATCO Task Prediction Model largely retains the data ingestion and filtering processes established in the first iteration. Given that the dataset remains the same, the spatial and temporal filters applied in the initial version continue to be relevant.

In this iteration, a more stringent filtering process has been applied, incorporating both previously implemented functions and new methods. The goal of this enhanced filtering is to focus exclusively on flights that meet specific time and zone constraints within the sector and provide consistent ATCO task information. After completing this filtering process, the resulting number of tasks is presented in Figure 14.

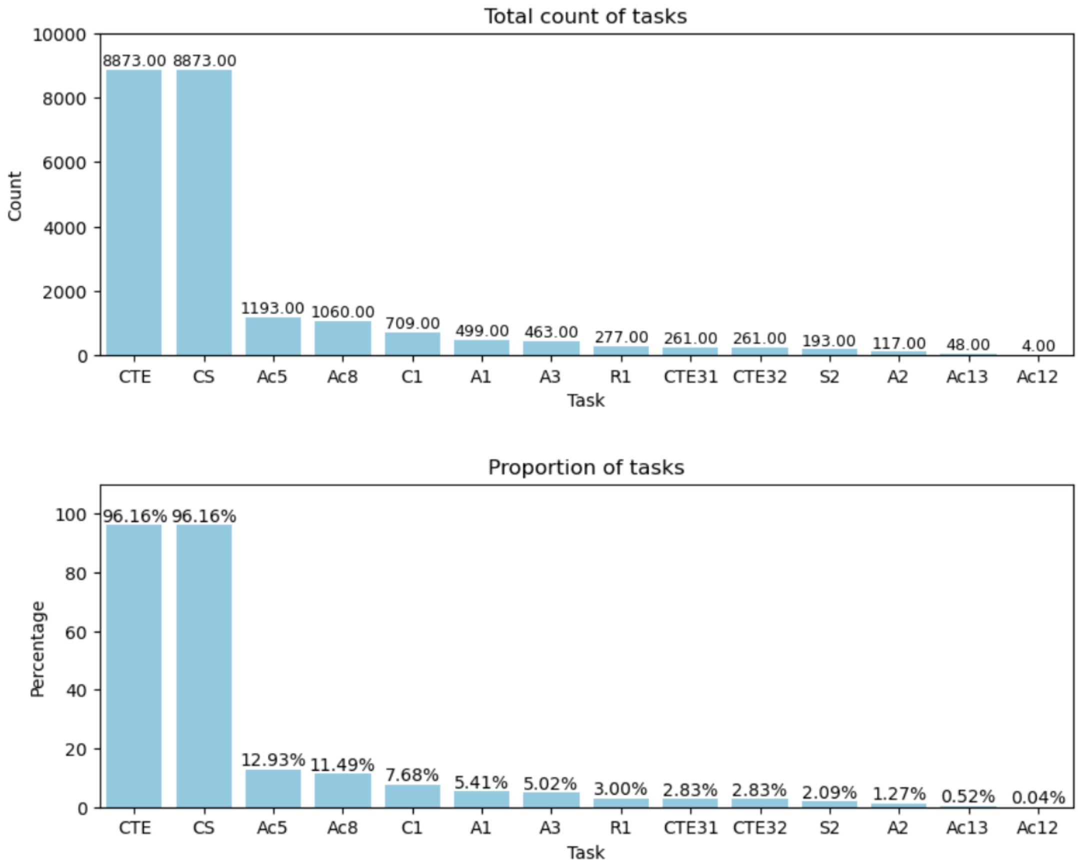


Figure 14: Number of tasks after filtering.

5.1.2 Task Probability Prediction approach

Instead of relying on a single predictor for all task types, separated predictors will be designed for each type of tasks. The prediction process begins with the identification of relevant features for each task type. These features may include flight characteristics from the raw dataset (e.g., altitude, speed, trajectory, etc. from tracks and flight plans), or a subset of the labels generated by the label generators in a similar manner to the first iteration. A more accurate data analysis permits to determine which labels are more relevant to predict a specific task. Only the labels with demonstrated predictive value will be passed to a specific task model, which will be designed iteratively by incorporating new inputs/dependencies. This new approach can be viewed as the use of a different dataset to train each task prediction model. Since it is not known which algorithms are the best, it is possible that some machine learning models initially discarded during the first iteration may now be considered as viable options (e.g. SVM, logistic regression, multilayer perceptron, etc.). Another potential approach would be to define a distance logic between flights (using a feature vector) and interpolate a local probability of occurrence of the action type in the vicinity of the flight under analysis.

A potential example of this approach would be the estimation of the probability of having a CTE task in the next minutes. Two features might be used in an initial implementation:

- Time needed to pass through the sector: in some cases, for flights which very rapidly overfly the sector, no responsibility change is performed. So, it may be expected that a dependency with this feature arises.

- Time to arrive to the sector: depending on the time left, and its relationship with the prediction time horizon, the probability to have such action will change.

In fact, in this case it is even possible both dependencies are roughly independent. So, a model based on data might help interpolate a function to estimate the probability based on these two inputs.

A similar approach might be used for CS. In fact, both estimators will share part of its implementation (given the one-to-one relation between CTE and CS). In this case the features might be the time needed to pass through the sector and the time to leave the sector.

Other examples of ad-hoc features related to probabilities would be those related to flight levels (in the tracks, entry points, exit points, intermediate waypoints) and related elaborations (vertical speeds) and their potential relations with changes in the probabilities of issuing altitude related orders.

5.2 CODA ATCO Task Prediction Model Implementation (v2)

This new approach requires dedicating effort to address each task individually, prioritizing tasks based on logical criteria. To optimize implementation, tasks will be ordered and scoped by their significance in the dataset. Specifically, tasks that occur most frequently in the data are given precedence, as they are not only better represented but also easier to predict due to the larger amount of available training data. Conversely, tasks that occur infrequently will be addressed later since their residual nature in the dataset makes them more challenging to model accurately. Based on this reasoning, the prioritized order of tasks for implementation can be seen in Table 14.

The defined scope is to cover as much tasks as possible, however due to time and data constraints, those tasks with a representation under of 10% will not be considered yet.

Order	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
Task	CTE	CS	Ac5	Ac8	A1, A2, A3, S2	C1	R1	CTE31	CTE32	AC13	AC12

Table 14: Task distribution.

Considering this, the model associated to each task or group of tasks analysed is described more in detail in this section.

5.2.1 CTE (Radar contact)

5.2.1.1 Feature selection

Based on this new approach, the first task to analyse, would be **CTE**, which represents the radar contact between the ATCO and the flight crew. For this task, a statistical approach is an interesting option rather than a classical classification machine learning predictor. The intention is to provide a probability of being contacted giving a certain time and within the next 5 minutes (the 5 minutes time horizon is a decision based on the typical times over the sector and could be changed). To estimate this probability function we will use the following information:

- **Time to intersect the sector**: The contact between an ATCO and the flight crew will be produced in a time range, usually before it enters the sector. This time is computed with the latest available updated flight plan and the 3D intersection point to the sector. With this feature a distribution can be computed that allows to check if the hypothesis is represented by the data, by calculating this time in the instant the CTE is performed. As it can be seen in Figure 15, and as expected, the greater number of contacts given this sample data (which represents a portion of the total) agglutinates in the range of 0 to 5 minutes to enter the sector. This feature not only represents a meaningful predictor for this task but also gives a time perspective that allows to introduce it to the prediction (probability within the next *n* minutes).

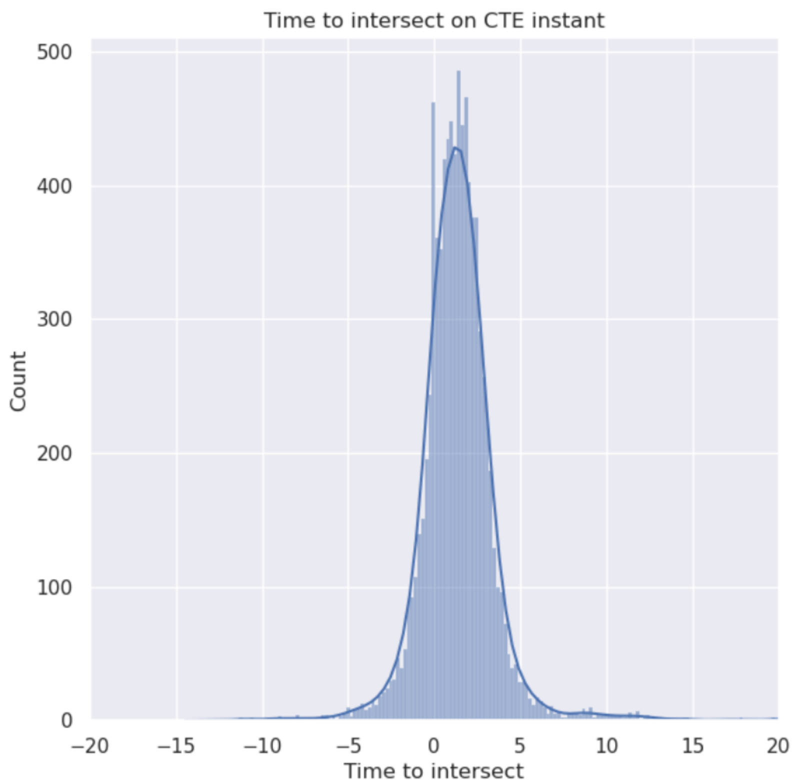


Figure 15: Distribution of time to intersect in CTE.

- **Time inside the sector**: Following the same line of reasoning used for the time to intersect, the total time inside the sector of the flight is calculated. The idea of the potential of this feature is to penalize those cases where the flight tracks through a corner of the sector, and there is no actual change of responsibility associated, as can be seen in the example in Figure 16. In other words, sector skipping is much more probable for those flights with reduced time inside the sector.

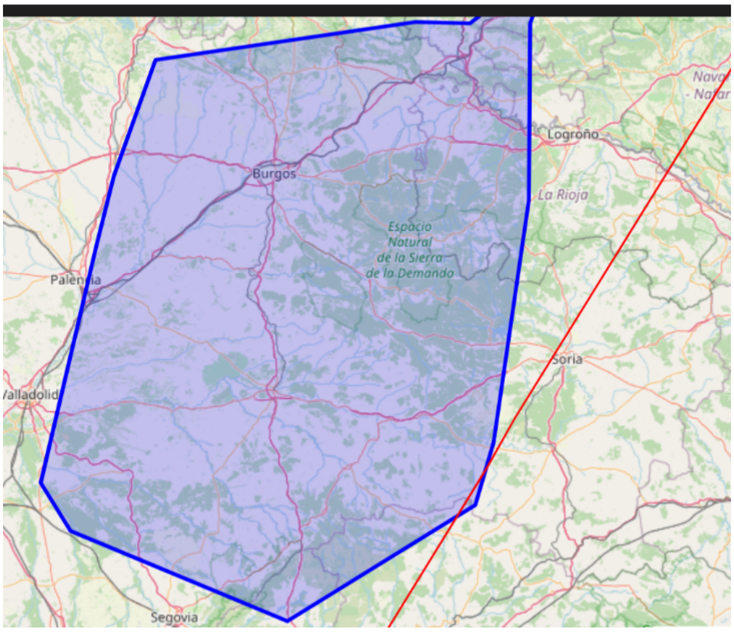


Figure 16: Example of borderline intersection between a flight and the under-study sector.

The time spent within the sector is calculated for each aircraft that intersects the sector. Using this distribution, the probability of a CTE for each time range is determined by dividing the number of CTE cases within that time range by the total number of flights in the corresponding bin. Figure 17 illustrates the overall histogram of feature variable, and the associated probabilities of having a CTE conditioned to each time interval bin.

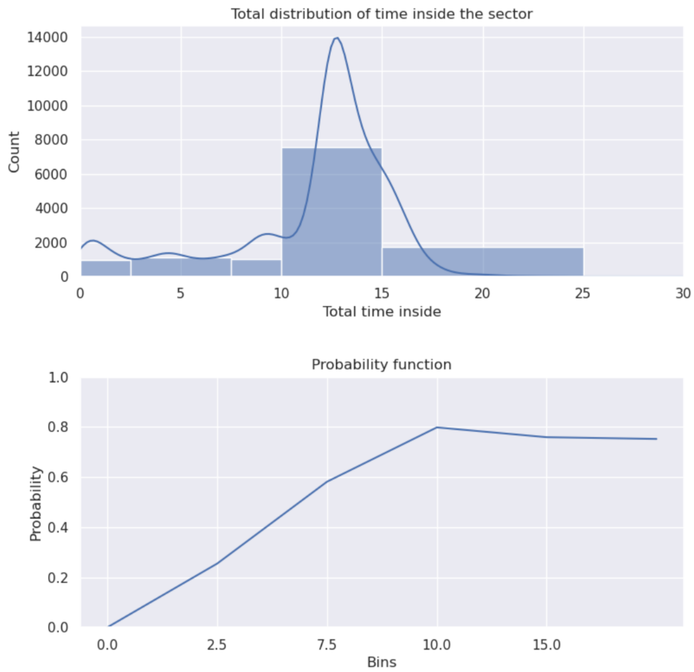


Figure 17: Distribution of time inside the sector feature.

- **Contact state**: This would represent whether the flight has already been contacted or not, it is considered that if the flight was already contacted in the past, it has extremely low (0 probability) chances of being contacted again. This feature would be used to drop the probability once the CTE has been produced.

5.2.1.2 Probability estimation

Considering this set of features, the following probabilities are computed to get the desired estimator.

- **P (time to intersect $_{t,\ t+5}$ | CTE) Probability of CTE in time *t***: assuming that there is CTE, this probability is computed using the distribution of Figure 15. For this, the current time bin and the ones corresponding to the next five minutes are accumulated. Since this probability is considering that there is going to be CTE, the remaining probability to the moment *t* is normalized as is considered to accumulate the prior probability that has not occurred.
- **P (CTE | total time) Probability of CTE for given total time:** in this case, the probability for the corresponding bin is computed. When has already entered the sector, meaning time to intersect < 0, this probability will be calculated with the remaining time (time to intersect + total time inside). This is performed with the intention of dropping the probability as the flight gets closer to leaving the sector.

The output of the CTE probability estimator, having the inputs *[time to intersect, total time inside, contacted],* would be:

- *If time to intersect >= 0* and not contacted*,*

$$P\ (CTE_{\ t,\ t+5}\ ) = P\ (\text{time to intersect}_{\ t,\ t+5}\ |\ CTE) * P\ (CTE\ |\ \text{total time})$$

- *If time to intersect < 0* and not contacted,

$$P\,(CTE_{\,t,\,t+5}) = P\,(\text{time to intersect}_{t-5,\,t} \mid CTE) \;*$$
$$P\,(CTE \mid (\text{total time} + \text{time to intersect}))$$

- *If* contacted or *(total time + time to intersect) < 0,*

$$P\,(CTE_{\,t,\,t+5}) = 0$$

5.2.2 CS (Contact Release)

5.2.2.1 Feature selection

Once the CTE model has been established, the next logical step is to estimate the probability of the complementary task, CS. As with the other tasks, there is a direct correlation between the occurrence of CS and CTE, given that every aircraft contacted through the CTE task must subsequently be released via the CS task (this has been validated by our training data).

Therefore, the probability of the CS task is inherently conditioned on the probability of the CTE task. To enhance this analysis and provide a more precise determination of the timing for the CS task, the following features are expected to be particularly informative:

- **Time to Leave**: It is hypothesized that the expected time remaining until an aircraft exits the sector serves as a strong indicator of the probability of the CS task occurring within the next few minutes. Like the analysis of **time to intersect** in 52 for the CTE task, **time to leave the sector** is utilized here as a predictive feature for CS. This metric provides a temporal framework to anticipate when the task is most likely to be executed.
- **Contact state**: This would represent whether the flight has already been contacted or not, it is considered that if the flight was already contacted in the past, it has extremely high (100% probability) chances of being released.
- **Released state:** Once a CS task occurs, the probability of subsequent CS tasks (CS included) for the same flight immediately drops to zero, as the aircraft has already been handed over and is no longer under the ATCO's control.

EUROPEAN PARTNERSHIP
Co-funded by the European Union

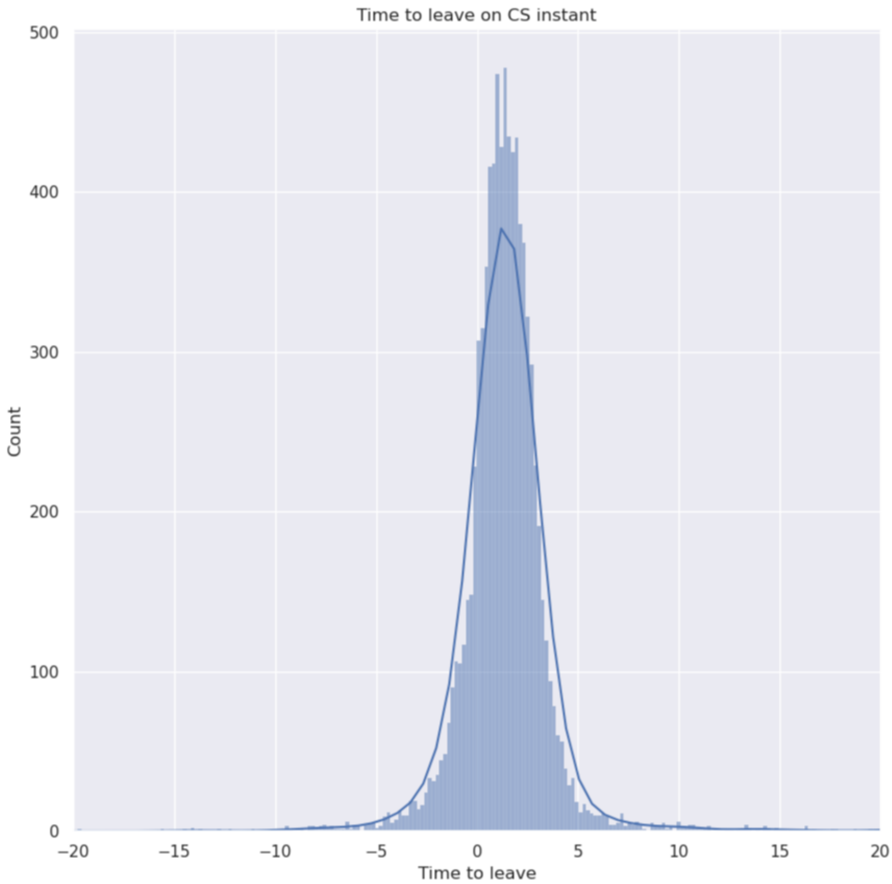


Figure 18: Distribution of time to leave for CS.

5.2.2.2 Probability estimation

As previously stated in the former section, the probability terms for CS estimation are:

- **P (time to leave $_{t,\ t+5}$ | CS) Probability of CTE in time *t***: assuming the occurrence of CTE, and therefore CS, the probability for the CS task is computed using the distribution of the **time to leave** feature, conditioned on the CS event. To estimate this, the current time bin, along with the bins corresponding to the next five minutes, are accumulated. Since this probability calculation assumes the occurrence of CS, the remaining probability at time **t** is normalized. This normalization process accounts for the prior probability that has not yet occurred, ensuring that the model accurately reflects the likelihood of the CS task within the given time frame.
- **P (CTE $_{t,\ t+5}$)**: extracted from the CTE estimator output. Even when the crew has not been yet contacted, we may predict the occurrence of a release, especially in those cases where the flight is expected to be contacted with high probability in the next minutes. Once contacted, this feature should not be used anymore for the output.

Considering this, the **output** of the CS probability estimator would be:

- If *not contacted*:

$$P\ (CS_{\ t,\ t+5}) = P\ (time\ to\ leave_{\ t,\ t+5}\ |\ CS) * P\ (CTE_{t-5,\ t})$$

EUROPEAN PARTNERSHIP
Co-funded by the European Union

- If *contacted*:

$$P\,(CS_{t,\,t+5}) = P\,(\text{time to leave}_{t,\,t+5} \mid CS)$$

- If *released*:

$$P\,(CS_{t,\,t+5}) = 0$$

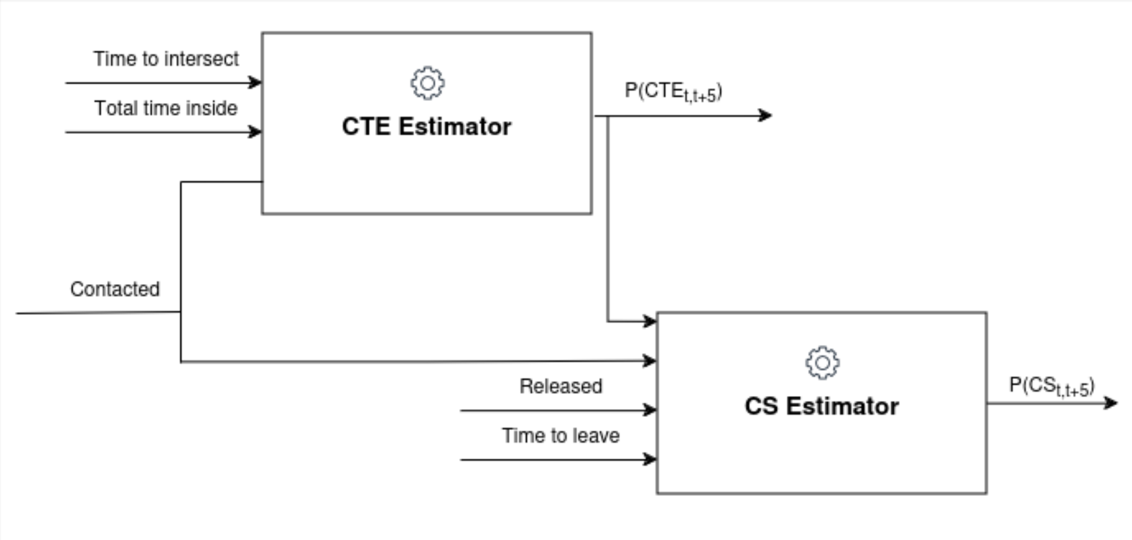


Figure 19: Estimator structure diagram.

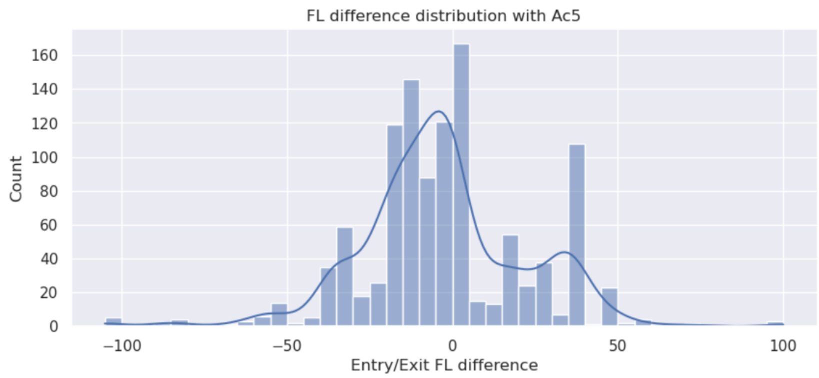
5.2.3 Ac5 (Procedural level change)

5.2.3.1 Feature selection

Task AC5 is related to procedural level adjustments. The difference in flight level between the sector entry and exit points, is a critical parameter for predicting the execution of this type of task. This parameter effectively encapsulates the level adjustments implemented within the sector. To develop an estimator capable of providing the probability function for this task, the following information is considered:

- **Flight Level difference at the sector entry and exit:** This metric quantifies the change in flight level experienced by an aircraft during its transit through a sector. By comparing the flight level at the point of sector entry to the flight level at the point of sector exit, it becomes possible to determine whether a significant altitude adjustment has occurred. This parameter is instrumental in identifying the execution of task AC5, as it facilitates the detection of specific altitude adjustment patterns relative to the aircraft's position within the sector.

  Performed observations indicate that instances where the flight level difference between sector entry and exit is zero are typically not associated with task AC5. This finding underscores the predictive importance of the flight level difference. Figure 20 further supports this conclusion, illustrating the distribution of flight level differences and their relationship to the occurrence of task AC5.



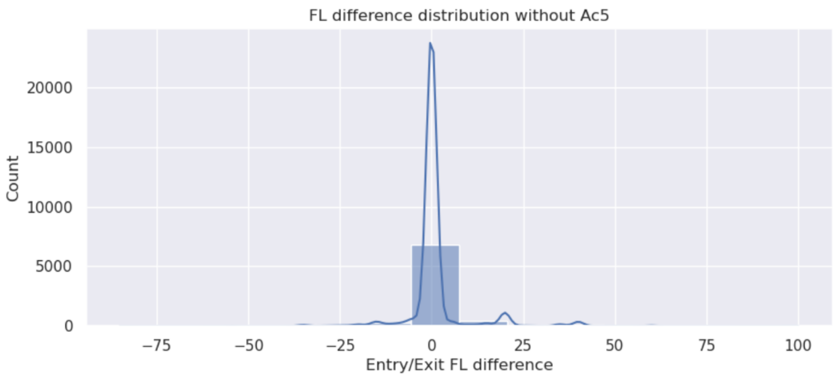


Figure 20: Comparison between flight level difference with Ac5 and without Ac5.

- **Time-to-Leave:** The *time-to-leave* value, calculated also for task AC5 execution, is used to model the probability distribution of task occurrence over the duration of an aircraft's presence within the sector. The remaining time until sector exit is hypothesized to be a strong indicator of the likelihood of task AC5 occurring within the subsequent minutes.

  Analogous to the "time-to-intersect" variable discussed in Section 1, the *time-to-leave* parameter provides valuable insight into the temporal patterns of flight level adjustments. Intuitively, larger flight level adjustments are expected shortly after an aircraft enters a sector, allowing sufficient time for the adjustment to stabilize before exit. Conversely, smaller adjustments are anticipated closer to the sector exit, where opportunities for significant changes are constrained by time.

  This correlation between the remaining time in the sector and flight level adjustment patterns is leveraged as a predictive feature for estimating the probability of task AC5 occurrence. Figure 21 illustrates the computed distribution of the *time-to-leave* value and its relationship with altitude change events.

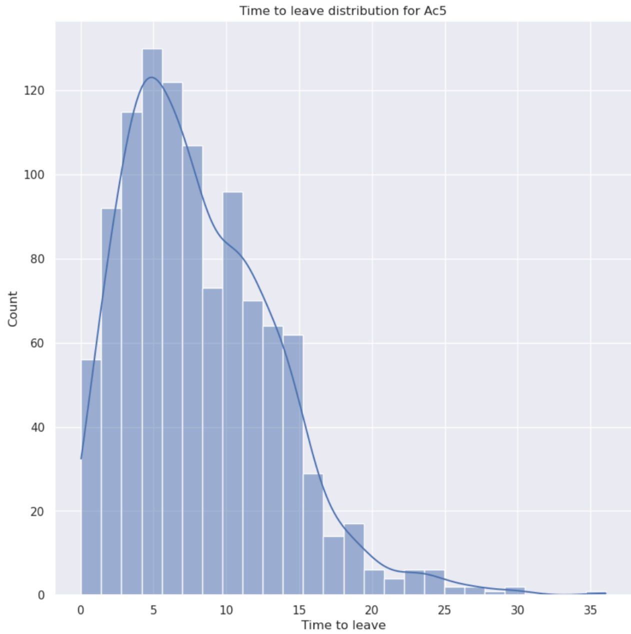


Figure 21: Time to leave distribution for Ac5.

- **Contact state**: This would represent if the flight has been contacted by an ATCO, if the aircraft has not been contacted the probability of having an Ac5 within the next few minutes depends on the probability of being contacted first.
- **Released state:** Once a CS task occurs, the probability of subsequent CS tasks, like Ac5, for the same flight immediately drops to zero, as the aircraft has already been handed over and is no longer under the ATCO's control.

5.2.3.2 Probability estimation

Considering this set of features, for the estimator building the following probabilities are computed to get the desired output of the system.

- **P (Ac5 | CTE ∩ FL difference) Probability of AC5 for a certain FL difference h conditioned to CTE**: the probability extracted from the distribution of FL differences in AC5 conditioned to CTE. This probability is given by the values shown in Figure 22.

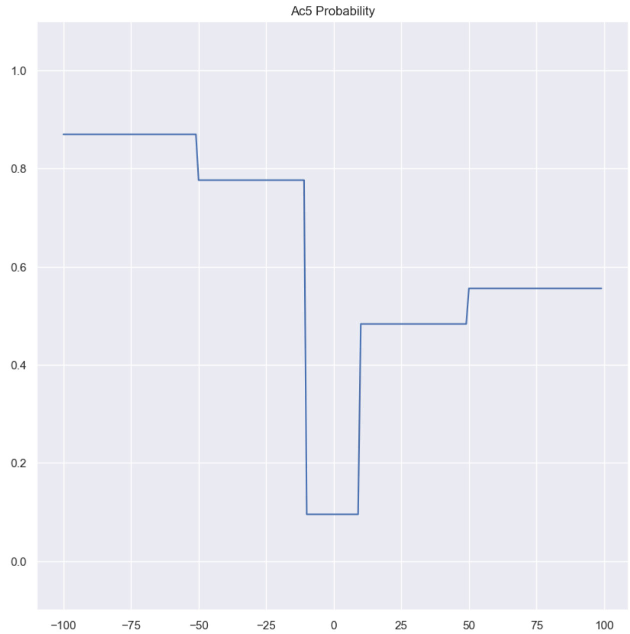


Figure 22: Probability function for flight level difference in Ac5.

- **P (CTE $_{t,\, t+5}$):** extracted from the output of CTE estimator.
- **P (time to leave $_{t,\, t+5}$ | Ac5):** assuming that there is CTE, this probability is computed using the distribution of the time to leave feature conditioned to the AC5 event. For this, the current time correspondent bin and the ones corresponding to the next five minutes are accumulated. Since this probability is considering that there is going to be AC5, the remaining probability to the moment *t* is normalized as is considered to accumulate the prior probability that has not occurred.

Therefore, the output of the system, for precise time *t* and a FL difference *h*, would be:

If *time to leave* >= 0 and not *contacted*:

$$P(Ac5_{t,\, t+5}) = P(Ac5 \mid CTE \cap FL\ difference) * P(CTE_{t,\, t+5}) * P(time\ to\ leave_{t,\, t+5} \mid Ac5)$$

If time to leave >=0 and *contacted*:

$$P(Ac5_{t,\, t+5}) = P(Ac5 \mid CTE \cap FL\ difference) * P(time\ to\ leave_{t,\, t+5} \mid Ac5)$$

If *time to leave* < 0 or *released*:

$$P(Ac5_{t,\, t+5}) = 0$$

5.2.4 Ac8 (Direct route to a point)

5.2.4.1 Feature selection

The AC8 task involves the shortening of a flight's planned route by the ATCO using direct routes between waypoints. Analysing the possible causes behind this type of task involves considering several factors. A key aspect is identifying the objective the ATCO aims to achieve by implementing such measures.

It is reasonable to understand this task as an optimization strategy, typically undertaken when air traffic density is low, allowing the ATCO to allocate time to improve flight efficiency by reducing time and fuel consumption. The motivations for this task depend less on the specific characteristics of the flight itself and more significantly influenced by the broader air traffic context, particularly when compared to the other tasks reviewed. Considering this, the analysis focuses on the congestion level within the sector under study.

Based on this focus, the features computed for trying to predict the AC8 tasks are the following:

- **Number of in contact flights inside the sector:** To estimate the probability of occurrence for this event, the number of contacted flights within the sector under study is computed. This is achieved by dividing each day, based on the available flight data, into time windows of X minutes. Within each time window, the number of flights present in the sector, as well as the number of CTE and AC8 tasks, are recorded. By aggregating this data, it becomes possible to analyse whether there is a correlation between the number of flights or contacted flights and the number of AC8 tasks. This approach enables the identification of patterns and relationships that may help predict the likelihood of AC8 tasks based on the volume of flights and the occurrence of CTE tasks within specific time intervals. Several window times were tested (6,12,30 mins) however none of them seem to provide strong enough evidence to affect this predictor. The number of windows with Ac8 tasks increased as the congestion increases, however, when it comes to the probability for a specific flight, remains almost constant.

- **Progression till next waypoint**: Once the probability of AC8 is determined, based on the number of contacted flights within the sector, an additional feature is required to distribute this probability over the specific moment when the flights pass through the sector. Considering this, **progression to the next waypoint** is considered a relevant indicator. It is logical to assume that the order to perform AC8 would be issued with sufficient reaction time to bypass the next waypoint. Progression is defined as the ratio of the time elapsed since the previous waypoint to the total time between the current waypoint and the next. For example, if the time distance between two waypoints is 100 seconds, a progression value of 0.5 would indicate that the aircraft has completed the first 50 seconds of the journey, with another 50 seconds remaining until reaching the next waypoint. This progression metric provides a dynamic measure of the aircraft's current position relative to its route, offering a useful context for timing the AC8 task.

Co-funded by the European Union

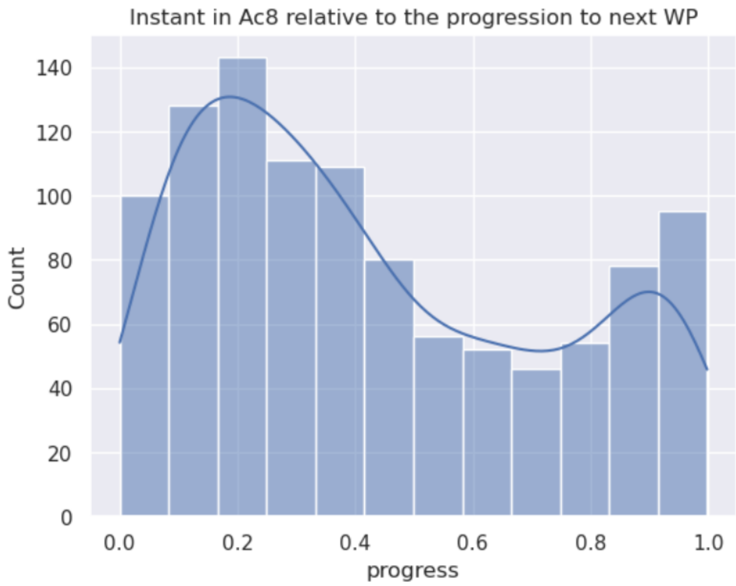


Figure 23: Progress till next WP on Ac8 distribution.

- **Total time to next waypoint**: since the probability targeted in this analysis spans from time t to $t+5$, the time to the next waypoint is calculated to provide a dynamically adjusted window shift for the progression feature mentioned earlier. For example, this approach ensures distinct probabilities for a progress of 50%, depending on whether the travel time between waypoints takes 5 or 10 minutes, thereby adapting to varying travel durations.

- **Correlation with vectoring tasks (S2)**: a vectoring task involves modifying an aircraft's expected direction to avoid potential conflicts (Discussed on section 8). After completing such kind of tasks, ATCOs often return the aircraft to its original route by implementing an AC8 task. This hypothesis is supported by the available data, since the **48%** of the aircraft that receive an S2 task, receive later an Ac8. Even though it does not appear to be such a great cipher, it is considerably higher than the base rate of Ac8s, which barely represent a 11% of the whole set of tasks.
- **Common waypoints:** this line of reasoning stated that there could be waypoints which allegedly receive considerably more directs. During CODA simulations with ATCOs, it was observed that a high percentage of the Ac8 tasks applied by the ATCOs had the same waypoint as target. To achieve this, the ratio of occurrences in flight plans with Ac8 to occurrences in all flight plans was calculated. However, no relevant pattern was found.
- **Contact state**: This would represent if the flight has been contacted by an ATCO, if the aircraft has not been contacted the probability of having an Ac8 within the next few minutes depends on the probability of being contacted first.
- **Released state:** Once a CS task occurs, the probability of subsequent CS tasks, like Ac8, for the same flight immediately drops to zero, as the aircraft has already been handed over and is no longer under the ATCO's control.

Other features were analysed to find an accurate predictor of this task. However, none of them depicted more representative results than the prior described feature. Among these try-outs where the magnitude of the angles between waypoints for each route and the delay of the flight.

5.2.4.2 Probability estimation

Considering this set of features, for the estimator building the following probabilities are computed to get the desired output of the system.

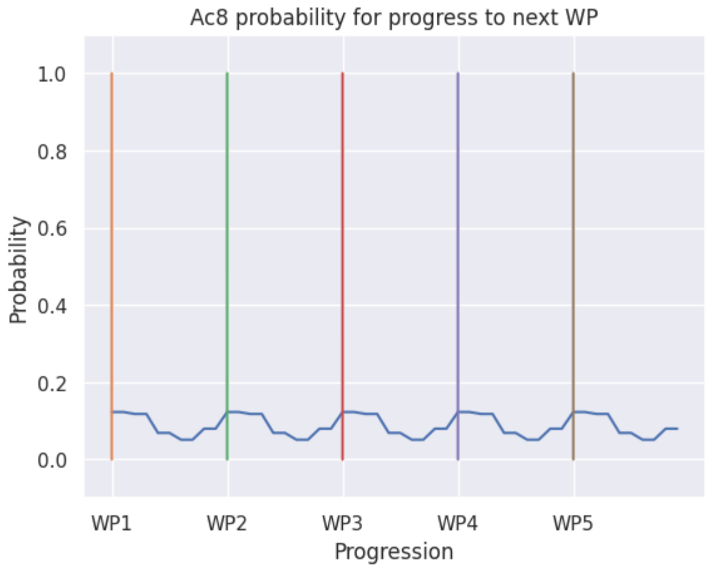


Figure 24: Ac8 probability for progress to next WP.

- **P (Ac8 base rate) = 0.11,** extracted from the data after the filtering process.
- **P (*$Ac8_{t,t+5}$* | *Ac8* ∩ progress and time to next WP)** represents the probability that, knowing there is going to be an Ac8, the task is performed within the next 5 minutes.
- **P (Ac8 | S2) = 0.48**, probability of having an Ac8 after an S2 has been received.

So, the final approach to calculate the Ac8 probability is:

If not *S2* and not *contacted*:

$$P\,(Ac8_{t,t+5}) = P\,(Ac8_{t,t+5} \mid Ac8 \cap \text{progress and time to next WP}) * P\,(CTE_{t,\,t+5}) * P\,(\text{Ac8 base rate})$$

If not *S2*, *contacted* and not *released*:

$$P\,(Ac8_{t,t+5}) = P\,(Ac8_{t,t+5} \mid Ac8 \cap \text{progress and time to next WP}) * P\,(\text{Ac8 base rate})$$

If *S2* and not *released*:

$$P\,(Ac8_{t,t+5}) = P\,(Ac8_{t,t+5} \mid Ac8 \cap \text{progress and time to next WP}) * P\,(Ac8 \mid S2)$$

If *released*:

$$P\,(Ac8_{t,t+5}) = 0$$

For this system it must be noted that if the time till the next waypoint is less than the five minutes window shift, the value of the term ***P*** $(Ac8_{t,t+5} | Ac8 \cap$ ***progress to next WP)*** should be ***1.***

5.2.5 Task of separation or sequencing: A1, A2, A3, S2

5.2.5.1 Feature selection

The tasks grouped in this section are related to various approaches an ATCO may employ to perform two different types of task: 1) avoiding conflicts by managing separation and 2) performing sequencing. Separation refers to maintaining the minimum safe distance between aircraft, while sequencing involves organizing the order of aircraft to ensure a safe flow. Although they are different tasks, they were grouped to have enough data to try to have more representative prediction models, and also because ATON does not make a distinction between them, as it is not simple from an external system to tell what the actual reason for some of the ATCO actions is. Table 15 depicts the individual tasks that form this group. Summarizing, we tried to find a joint model for them due to the following reasons:

- **Respond to the same events**: all the selected tasks for this group are different strategies to apply in a separation or sequencing situation, it would be reasonable to think that they are triggered by similar conditions.
- **Same objective**: each of these tasks are aimed to ensure both minimum safe separation between aircraft and ordered flow, managing airspace safety.
- **ATCO's criteria**: without considering other factors in a potential conflict between two aircraft, it is thought that this potential conflict could be solved by the ATCO using various alternative strategies.
- **Representativeness**: there is a much more reduced number of instances of each task type, which makes the modelling more difficult and the potential results much noisier. However, grouping them this problem may be alleviated.

Task	Description
A1	Change in level for separation or sequencing
A2	Speed adjustments for separation or sequencing
A3	Direct routing to a point for separation or sequencing
S2	Vectoring for separation or sequencing

Table 15: Separation or sequencing tasks.

To find in the data patterns that contribute to the discrimination of flights related to these tasks, the following features have been computed for this estimator:

- **Number of in contact flights inside the sector:** consistent with the reasoning applied in the analysis of the Ac8 task, would be logical to think that sector congestion plays a significant role in these types of tasks. However, increased traffic within the sector does not seem to correlate with a higher likelihood of performing such operations. When there are more aircrafts in the sector, the overall probability of having a separation or sequencing tasks increases, however, the individual probability for each aircraft, which it is aimed in this work remains steady.

- **Closer predicted distance to other aircraft:** The aim of this feature is to forecast all the distances that a current aircraft will have to rest of the ones that go through the sector. At a given time t, and in a window of t+15 min a sampling with frequency of 10 seconds is computed, calculating at each sample the distance to the current nearby aircrafts. It would be logical to think that separation tasks are more prevalent for the flights with reduced predicted separation. However, in our tests this is not reflected in our dataset, and therefore this measure cannot be used as a stable indicator of this kind of tasks.
- **Nearby airport destination:** since sequencing tasks often as used to sort the aircrafts entry to an airport, it was thought to be interesting to verify if there is a correlation between the flights that had as a destination an airport close to the under-study sector, and those aircrafts that received a sequencing task. However, not enough evidence was found in this attempt.
- **Time till landing:** as another attempt to find a pattern for the sequencing task, the expected time till landing was computed for all the contacted aircrafts with the aimed to discover a significant difference between those which receive sequencing tasks and those which do not. The reasoning behind this feature takes the lead of the previous one (nearby airport destination), since it would be logical to think that those aircraft that received such sequencing orders had a short or precise time till landing.

Even though there is still work on this matter with hope to find interesting results, not clear evidence of these tasks was found yet in the calculated features. Not feasible prediction nor forecasting is thought to be achievable in the current state of the performed analysis.

5.2.6 Simulations

This section exemplifies the system functionality through several simulations intended to depict and clarify in greater detail the desired functionality of the system, showing time evolutions of the calculated probability predictions for individual flights.

5.2.6.1 Simulations for CTE (Radar Contact) and CS (Contact Relase)

In this section, the output of the system is computed for both CTE and CS scenarios. Several situations are considered and summarized in Table 16, which displays various times spent inside the sector. In those simulations, the CTE contact, if present, is 2 minutes before entering the sector, and 5-minute windows are used for analysis. For each situation, two plots are displayed: one for the case where there is CTE and one for the case that there is not.

Co-funded by the European Union

Total time inside the sector expected	CTE Instant	Time window interval
1	2	5
2		
7.5		
12		
20		

Table 16: Simulations description.

Also, in Figure 29 and Figure 30, a comparison of the *time window interval* in the output of the system is shown. This parameter represents the number of minutes from a given instant to cover with the prediction. For instance, if *time window interval = 5* the probability computed by the system corresponds to the probability of the given task to happen within the next five minutes.

Simulation explanation

The following figures illustrate the simulations previously discussed, with each one showing two scenarios for the same time within the sector. Taking Figure 25 as an example, which can be applied to all simulations in this section, the y-axis represents the probability derived from the models for both CTE and CS tasks, while the x-axis represents the "time to intersect" feature. This feature measures the flight's progression, where a value of 20 indicates that the flight has not yet entered the sector, and a value of -20 indicates that it has been 20 minutes since the flight entered the sector.

The top plot shows a scenario where a CTE task occurs at a "time to intersect" value of 2, meaning that the task is performed by the hypothetical ATCO two minutes before the flight enters the sector. In contrast, the bottom plot shows a scenario in which no CTE task is associated with the flight. These plots may be interpreted as representing the probability evolution for both tasks in real-time flight conditions. Several scenarios with different times inside the sector are computed to system behaviour for each case.

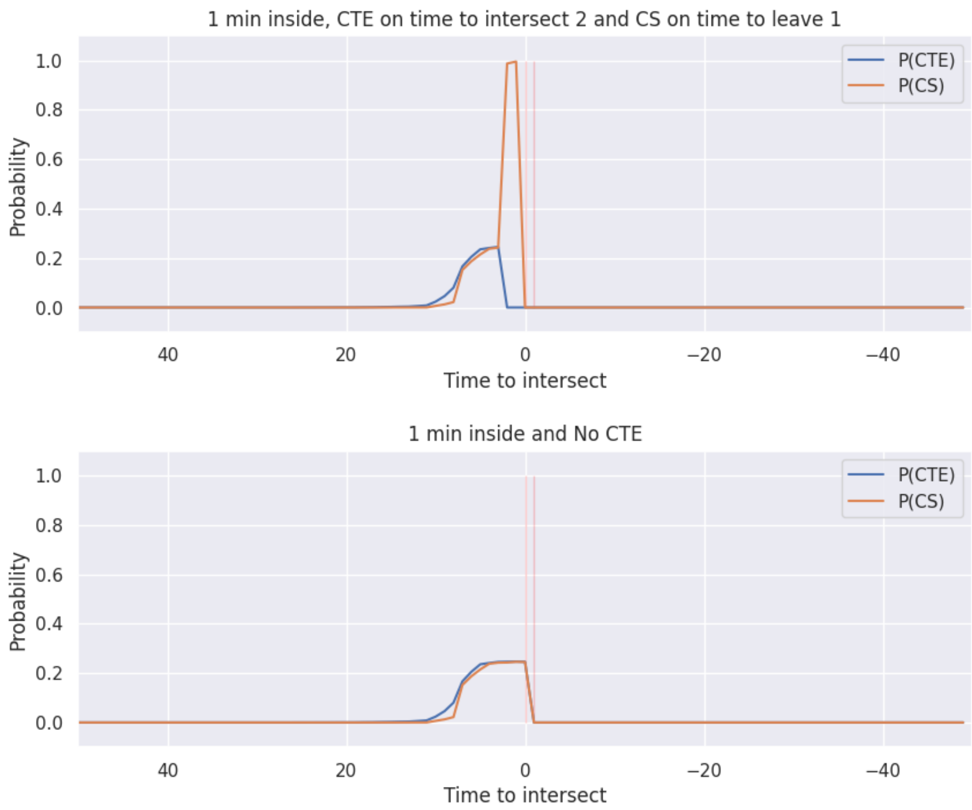


Figure 25: Simulation for time inside 1 min.

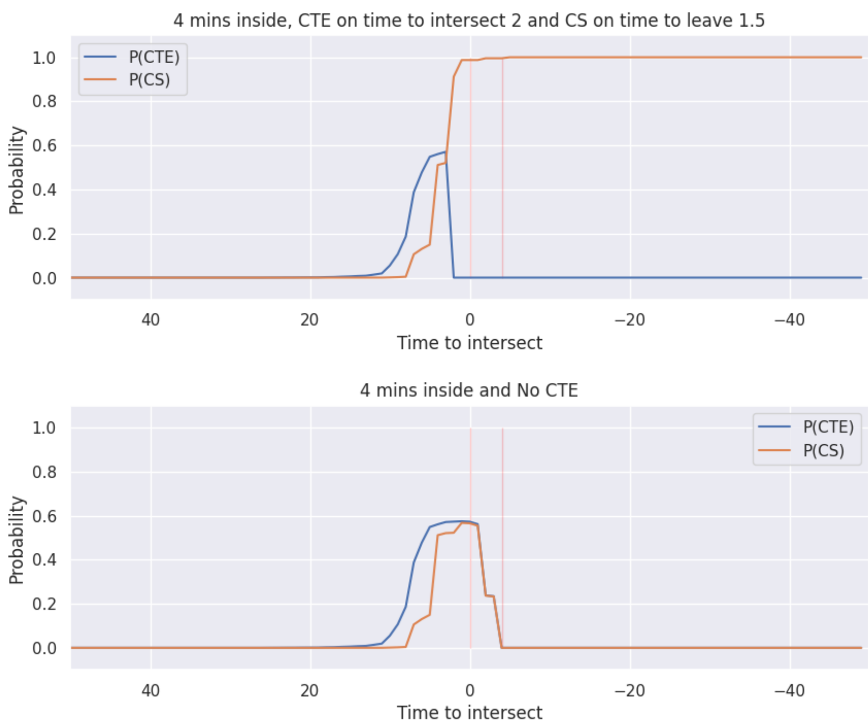


Figure 26: Simulation for time inside 4 mins.

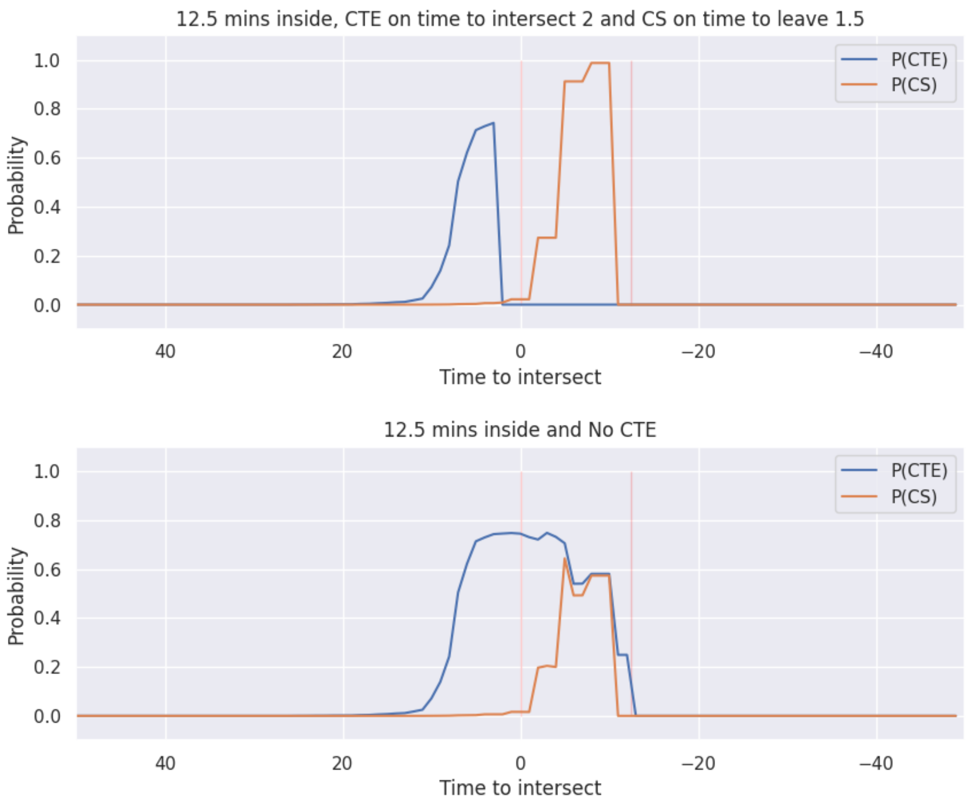


Figure 27: Simulation for time inside 12.5 mins.

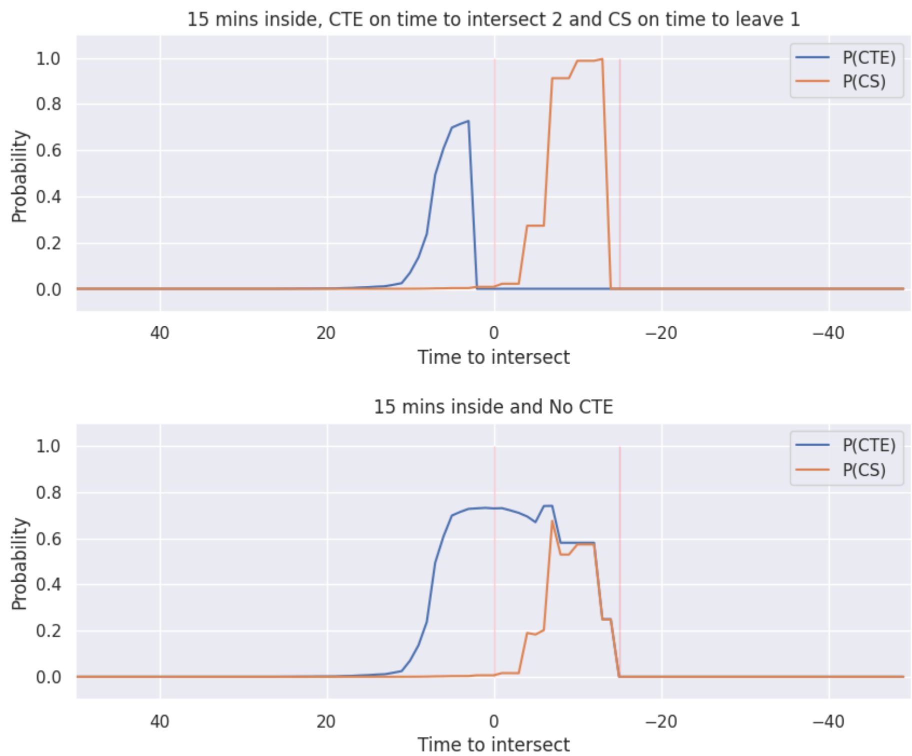


Figure 28: Simulation for time inside 15 mins.

Figure 29 and Figure 30 illustrate the impact of the time window interval on prediction outcomes. As shown, the larger the time window interval, the earlier the predictions begin to rise, as a longer forecast period is being considered.

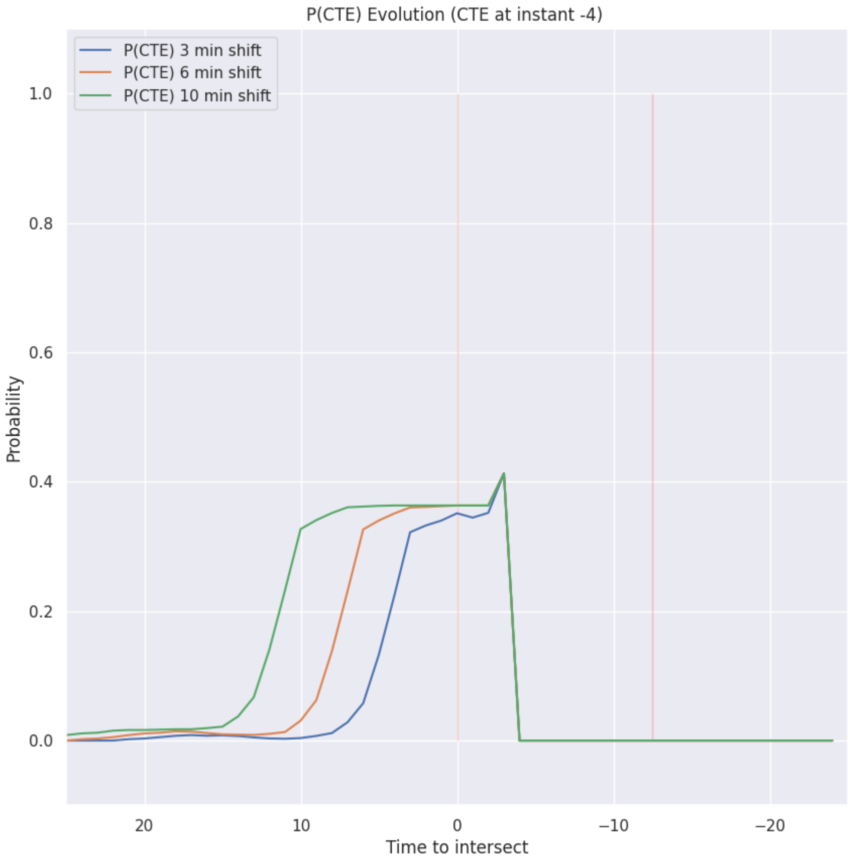


Figure 29: Impact of time window shift on CTE estimation.

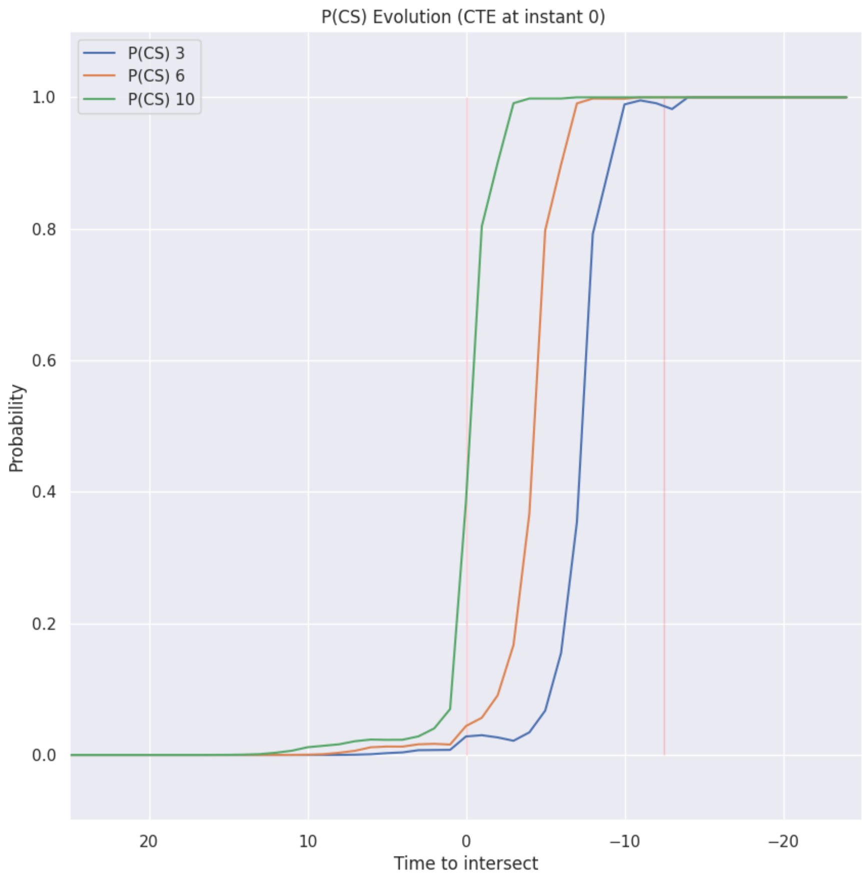


Figure 30: Impact of time window shift on CS estimation.

5.2.6.2 Simulations for Ac5 (Procedural Level Change)

Figure 31 and Figure 32 depict the output of the AC5 estimator in two distinct scenarios: one characterized by a significant FL difference and the other with no FL difference. In both cases, the estimator's output is displayed alongside the Radar contact (CTE) and Release (CS) estimations. This combined visualization facilitates a clearer understanding of the sequence and flow of events, allowing for a comprehensive analysis of the interactions between FL adjustments and communication dynamics within the sector.

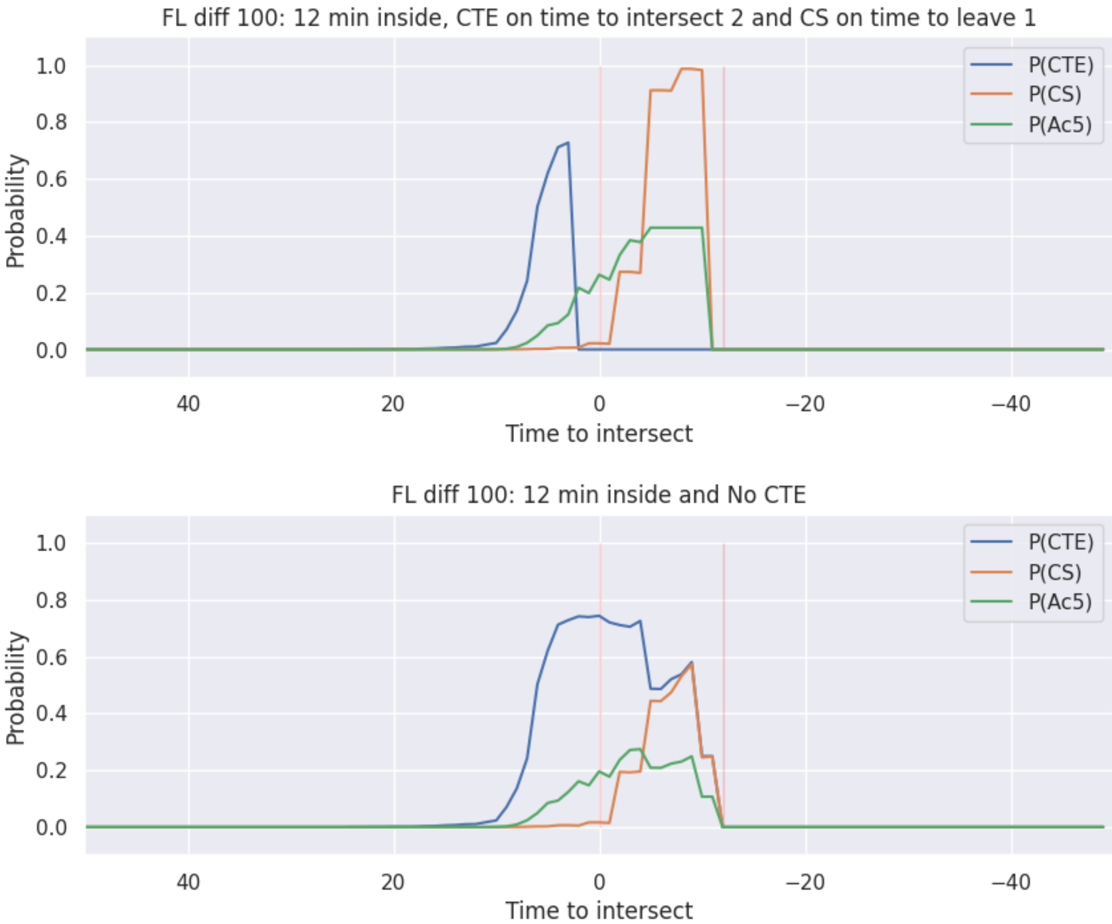


Figure 31: Ac5 probability evolution for FL difference of 100 flight levels.

In Figure 31 the behavior of CTE, CS, and Ac5 tasks can be observed when there is a FL difference of 100 flight levels. The probability of CTE starts to increase approximately 10 minutes prior to the intersection with the sector and drops to zero immediately after. Additionally, the probability of the CS task rises sharply when the flight intersects the sector, provided a CTE has occurred, and then stabilizes. Meanwhile, the probability of Ac5 increases during the intersection and settles at approximately 0.5 after CTE occurs. After the aircraft leave the sector, the probability of Ac5 decreases to zero. Conversely, in the absence of a CTE event, the probability of CTE shows a slight increase prior to the aircraft entering the sector but subsequently diminishes to zero in the absence of radar contact. Under these conditions, the probabilities of CS and Ac5 exhibit modest increases before rapidly returning to zero, reflecting the absence of a CTE event and the aircraft's departure from the sector.

EUROPEAN PARTNERSHIP
Co-funded by the European Union

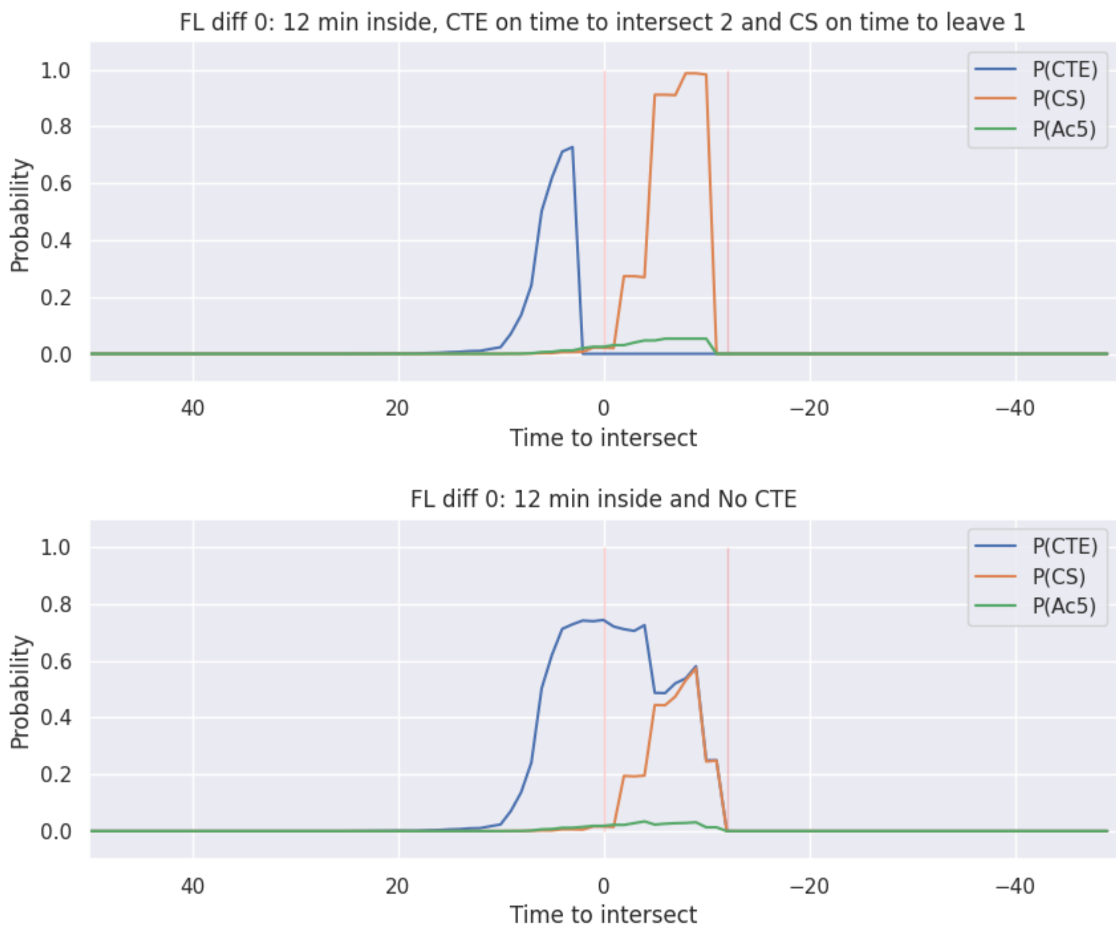


Figure 32: Ac5 probability evolution for FL difference of 0 flight levels.

In Figure 32, it can be analysed the behaviour of CTE, CS, and Ac5 tasks under a FL difference of 0. The probability of CTE begins to rise approximately 10 minutes before the flight intersects the sector and rapidly drops to zero immediately after the intersection. The probability of the CS task increases sharply as the flight intersects the sector (assuming a CTE has occurred) and stabilizes shortly after. Meanwhile, the probability of Ac5 also increases slightly during the intersection and stabilizes at a much lower value. After the aircraft leave the sector, the probability of Ac5 decreases to zero.

Conversely, when no CTE occurs, the probability of CTE still increases slightly before the flight intersects the sector but declines gradually to zero afterward, reflecting the absence of radar contact. In this case, the probabilities of CS and Ac5 show minor fluctuations but remain close to zero, indicating that these tasks are not triggered without a prior CTE and the aircraft leaving the sector.

5.2.6.3 Simulations for Ac8 (Direct route to a point)

The Ac8 predictor probability function has a unique characteristic compared to the others. Since the probability is calculated relative to the distance between waypoints, it generates a periodic pattern while the flight remains within the sector. The simulation is particularly interesting as it highlights the difference between scenarios where a vectoring separation task (S2) is performed and those where it is not. In these cases, the probability transitions from an almost residual value to nearly a 50% chance of the task being performed. In these simulations, the CTE task is performed at 5 minutes before entering the under-study sector, meanwhile in the bottom plot, the S2 task is performed in the instant –5, meaning 5 minutes after entering the sector.

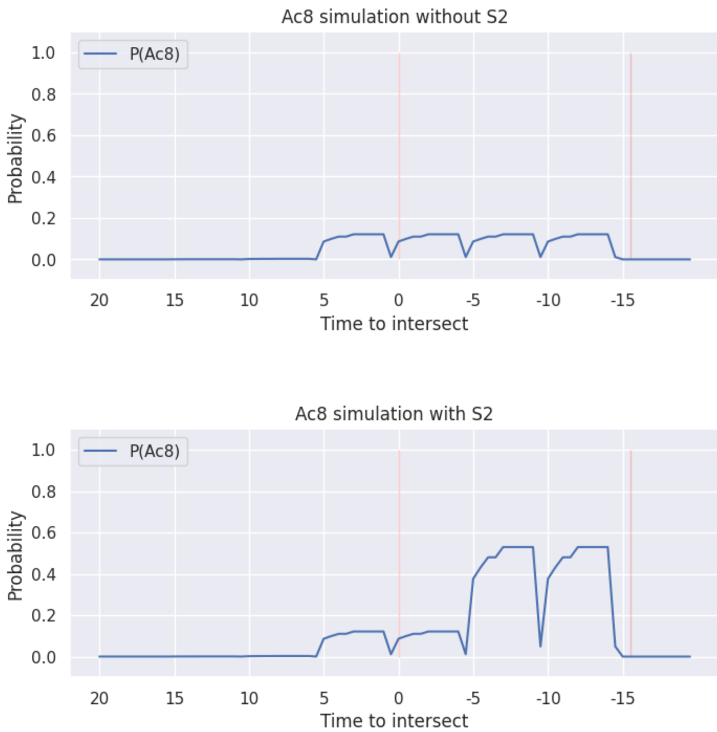


Figure 33: Ac8 probability comparison.

It can be noted that the probability reaches values greater than zero even before entering the sector, this happens because from the moment an ATCO establishes radar contact, other tasks could be performed even if the aircraft is not yet inside the sector. It is also interesting to mentioned that in case that the distance between waypoints of all the trajectory of the aircraft is lower than the window interval of 5 minutes, the output of the system would remain steady as it can be seen in Figure 34.

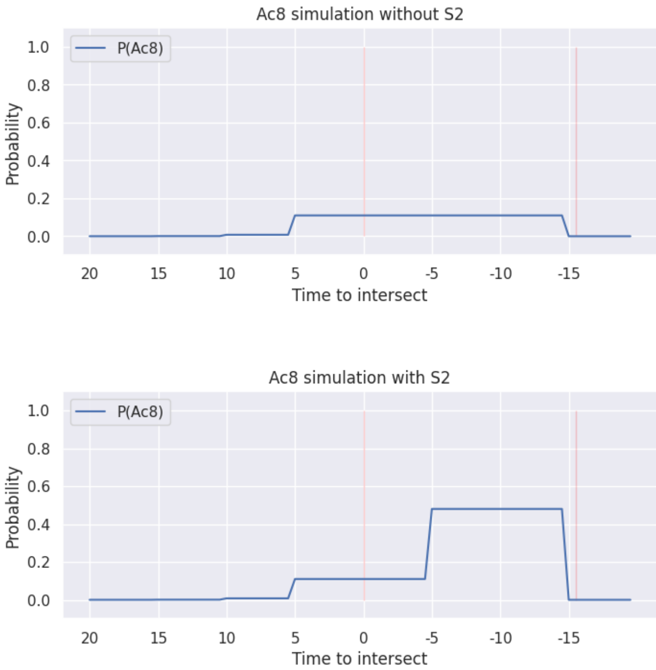


Figure 34: Ac8 probability special case.

5.3 CODA ATCO Task Prediction Models Validation (v2)

Since the second approach relies on statistical descriptions rather than traditional machine learning methods, standard classification metrics typically used for ML model evaluation cannot be directly applied to assess its performance. Instead, the validity of this approach can be related to the analysis of the robustness of the calculated features across the dataset, as the models themselves serve as statistical descriptors without training process.

To ensure the reliability of the features, the process of feature analysis was initially conducted on reduced samples of the data. Only those features that demonstrated consistent behaviour and robustness when applied to the entire dataset were retained for the final model. This matter is critical to maintaining the integrity and explanatory power of the statistical approach, even in the absence of conventional performance validation metrics. The following figures illustrate the distribution of the computed features for different random subsets of the available data to verify that the distribution is consistent.

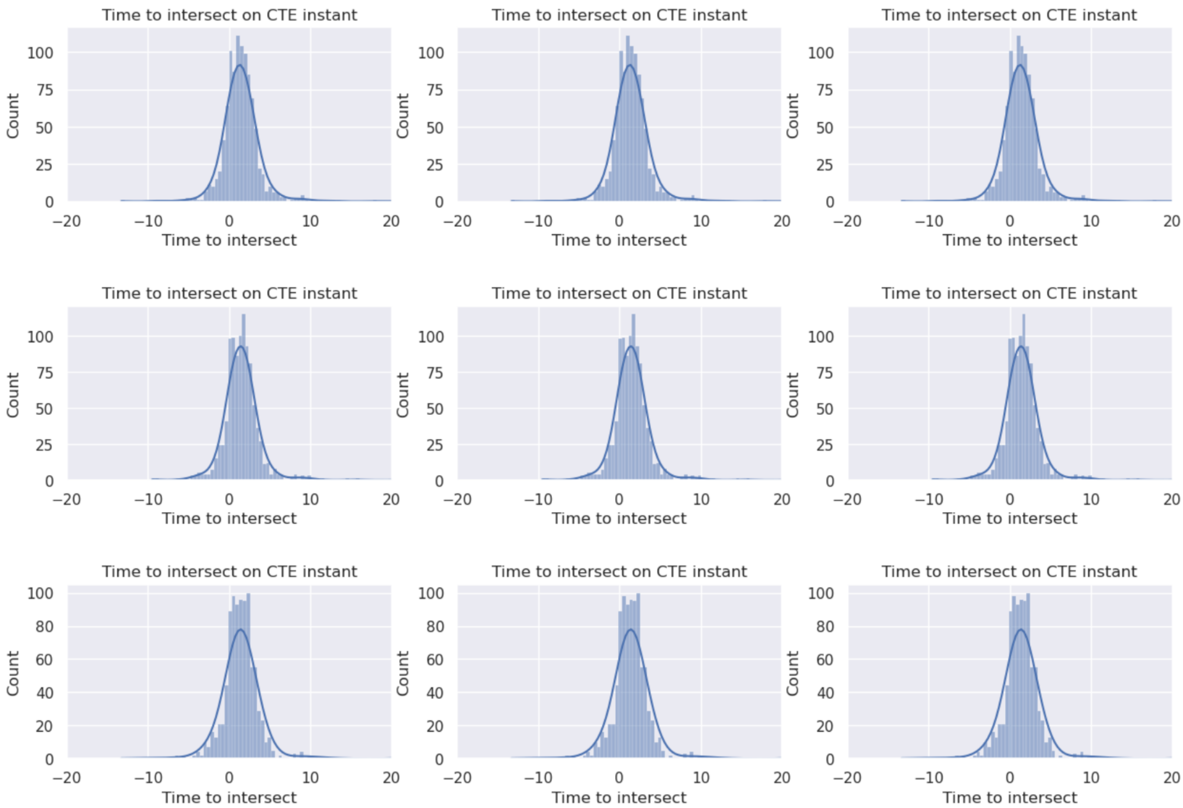


Figure 35: Distribution comparison for time to intersect on CTE.

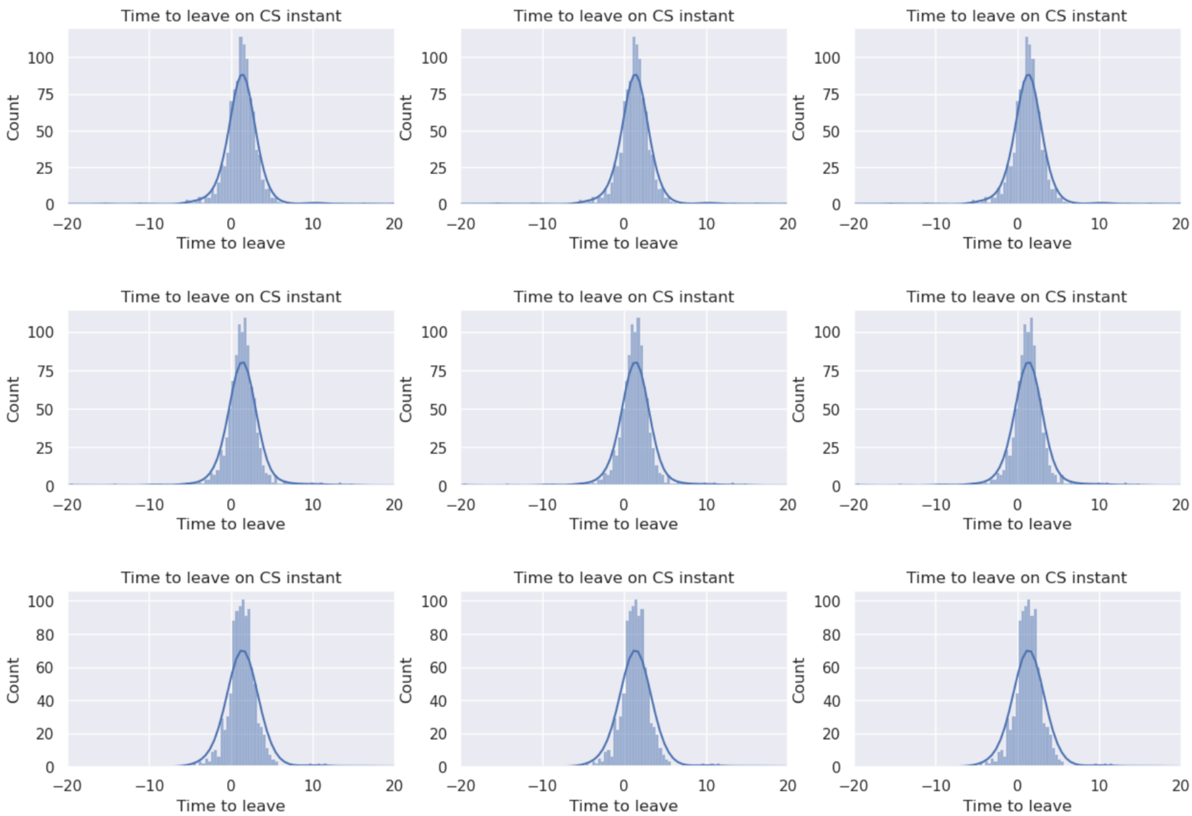


Figure 36: Distribution comparison for time to leave on CS.

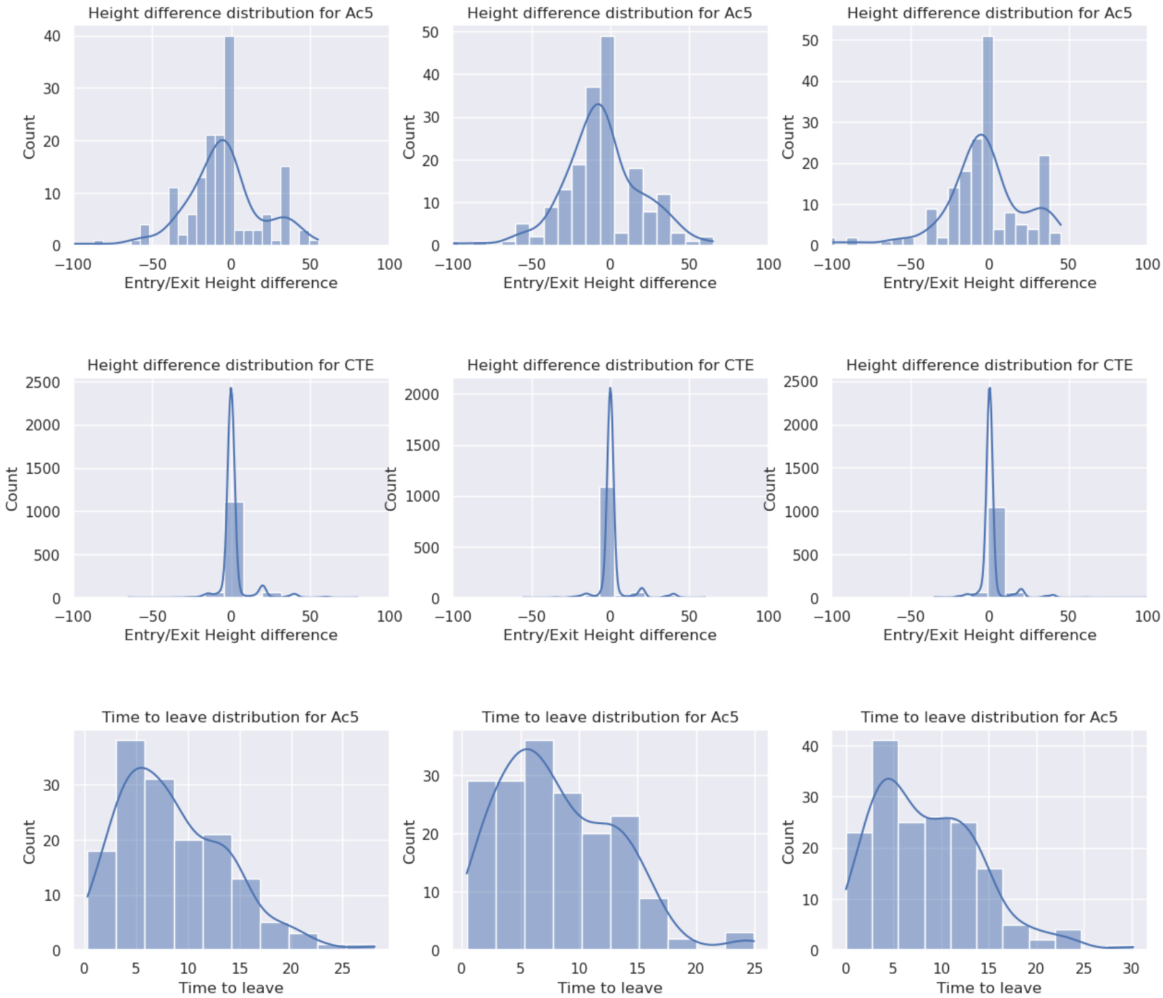


Figure 37: Distribution comparison for Ac5 features.

Co-funded by the European Union

6 Conclusion

The first iteration of CODA ATCO Task Prediction Model revealed both promising aspects and notable limitations. While the model achieved high overall accuracy, it predominantly excelled in predicting common ‘no task’ event, failing the prediction of ‘non-zero’ tasks in several cases and generating false alarms in others. The limitations highlighted by the initial iteration were produced by the imbalance in the dataset, the complexity of handling multiple labels, and the need for better task-specific prediction accuracy. These observations suggested a possible path for improvement in subsequent iterations.

The second iteration of the CODA ATCO Task Prediction Model offers several noteworthy observations. On one hand, for tasks where clear patterns were identified in the data, this approach provides a more stable and interpretable solution compared to Version 1. The models not only exhibit greater simplicity but also reflect logical reasoning behind the execution of each task. On the other hand, for tasks where no sufficiently strong patterns were detected, the predictions rely on the base rate of occurrence extracted from the data, falling short of achieving the desired level of customization.

Future work could benefit from deeper exploration of these patterns, including discussions with ATCOs to better understand and integrate their reasoning into the system, rather than relying solely on the given data. Such an approach could address the challenges associated with predicting ATCOs’ decision-making criteria, as different ATCOs may adopt varying operational strategies for the same situation or task.

In conclusion, the development of the two versions demonstrates that, while challenging, certain ATCO task predictions can be effectively achieved using simple and interpretable models, even without the traditional training associated with machine learning methods. For tasks where no clear patterns were identified through statistical analysis, a hybrid approach combining aspects of Version 1 and Version 2 could be considered. The metrics obtained from Version 1 indicate that **LSTM** models can enhance overall system performance, achieving decent metrics despite a loss of interpretability. As previously noted, the use of LSTM models for tasks lacking discernible patterns represents a viable option. This suggests the potential for a hybrid system that integrates descriptive statistical methods with machine learning models, such as LSTM, to optimize performance while balancing explainability and predictive accuracy.

EUROPEAN PARTNERSHIP
Co-funded by the European Union

7 References

[1] Maynard, E., & Harris, D. (2021) 'Using neural networks to predict high-risk flight environments from accident and incident data', International Journal of Occupational Safety and Ergonomics, 28(2), pp. 1204–1212. doi:10.1080/10803548.2021.1877455.

[2] Chatterji, G., & Sridhar, B. (2001) 'Measures for Air Traffic Controller Workload Prediction', 1st AIAA, Aircraft, Technology Integration, and Operations Forum. doi:10.2514/6.2001-5242.

[3] Delahaye, Daniel & Puechmorel, Stéphane. (2000). Air Traffic Complexity: Towards an Intrinsic Metric. Proceeding of the 3rd USA/Europe Air Traffic Management R and D Seminar.

[4] Isufaj, Ralvi, Thimjo Koca, and Miquel Angel Piera. 2021. "Spatiotemporal Graph Indicators for Air Traffic Complexity Analysis" Aerospace 8, no. 12: 364. https://doi.org/10.3390/aerospace8120364

[5] Karikawa, D., Aoyama, H., Takahashi, M., Furuta, K., Wakabayashi, T., & Kitamura, M. (2013). A visualization tool of en route air traffic control tasks for describing controller's proactive management of traffic situations. Cognition, Technology and Work, 15(2), 207-218. https://doi.org/10.1007/s10111-012-0222-y

[6] Pang, Y., Hu, J., Lieber, C. S., Cooke, N. J., & Liu, Y. (2023). Air traffic controller workload level prediction using conformalized dynamical graph learning. Advanced Engineering Informatics, 57, 102113.

[7] Cano, M., P. Sanchez-Escalonilla and M.M. Dorado. "Complexity analysis in the next generation of air traffic management system." 2007 IEEE/AIAA 26th Digital Avionics Systems Conference (2007): 3.D.4-1-3.D.4-9.

[8] Pham, D. T., Alam, S., & Duong, V. (2020). An air traffic controller action extraction-prediction model using learning approach. Complexity, 2020, 1-19. https://doi.org/10.1155/2020/1659103.

[9] Bastas, A., & Vouros, G. (2022). Data-driven prediction of Air Traffic Controllers reactions to resolving conflicts. Information Sciences, 613, 763-785.

Co-funded by the European Union

8 List of acronyms

Acronym	Description
AB	Advisory Board
ADS-B	Automatic Dependent Surveillance–Broadcast
AIP	Aeronautical Information Publication
ATCO	Air Traffic Controller
ATC	Air Traffic Control
ATFCM	Air Traffic Flow and Capacity Management
ATM	Air Traffic Management
ATON	Air Controller Task (dataset)
COMPAS	Cognitive system Model for Simulating Projection-based behaviours of Air traffic controllers in dynamic Situations
CODA	Controller Adaptive Digital Assistant
eTLM	Enhanced Traffic Load Monitoring
FIR	Flight Information Region
FP	Flight Plan
FL	Flight Level
GIPV	Flight plans dataset
ML	Machine Learning
SSR	Secondary Surveillance Radar
STAR	Standard Terminal Arrival Route

Table 17: List of acronyms

Co-funded by the European Union