INDRA

ATC
Tactical Opportunities Recommender

Research and Innovation

Álvaro Quintanar Pascual

22 April 2025 – Fly AI Forum

INDRA

01

About us

Our team

Álvaro Quintanar Pascual

AI Senior Specialist

Javier Herrera Hernández

Senior Manager - Software & Test

Daniel Cuenca Blasco

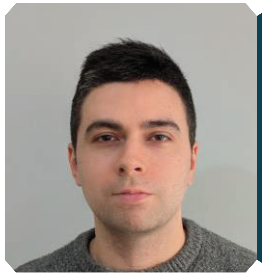
AI & Advanced Analytics Technical Manager

Víctor Manuel Martín García

AI Engineer

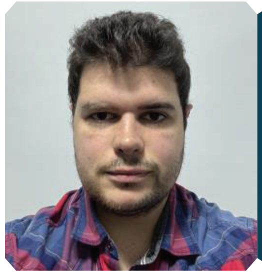
Alex Padros Zamora

AI Engineer



Daniel Arnedo Nieto

Data Architect



Santiago Prieto Alonso

Data Analyst

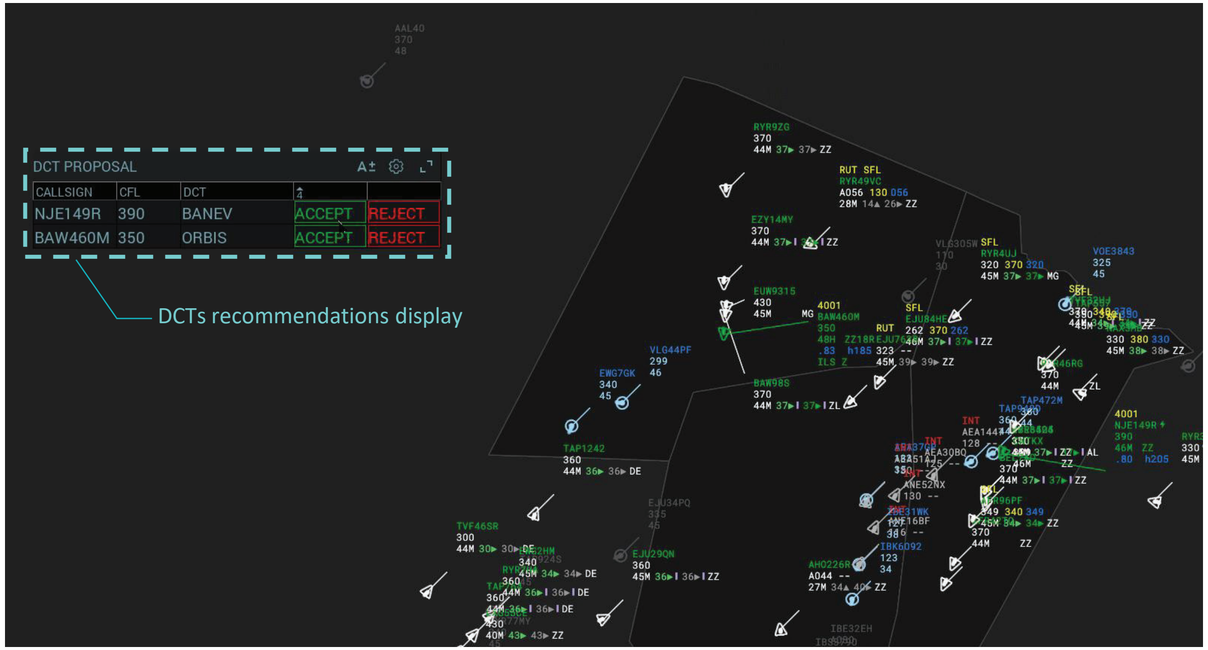
02

Description of the problem & approach

Which are the key aspects of the challenge? How are they being addressed?

Problem overview

Key objectives and challenges



A DCT recommendation system is proposed to optimize the airspace and allow the controller to accept or reject (providing a reason from the available options) these suggestions. As an initial approach, it will be limited to the Bilbao, Domingo, and Pamplona volumes.

This system will display a table of recommendations on the CWP showing the flight callsign and the waypoint to proceed to, along with the options to accept or reject.

***Current** constraints and features*

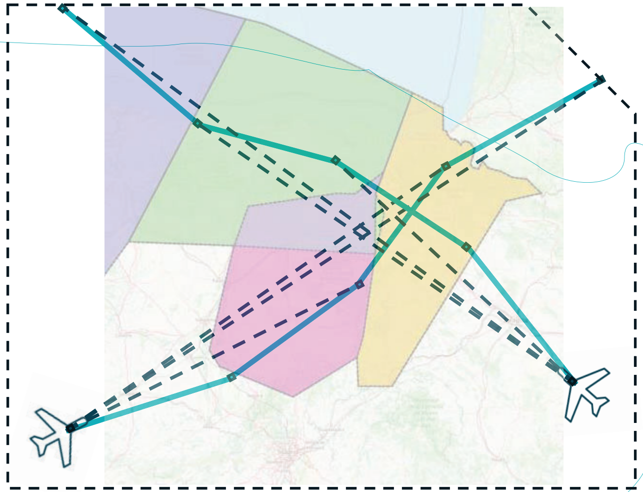
DCT proposal horizon 4 waypoints

Initial point

Single FL analysis and proposal

DCTs independency

Recommendations computed every 2 minutes



Sectors under test - LECMDGU + LECMBLU // LECMPAU

Avoiding causing new conflicts with DCTs

Real time ATCO interaction

Approach developed

* N: number of flights | M: maximum number of waypoints

01. Use case situation

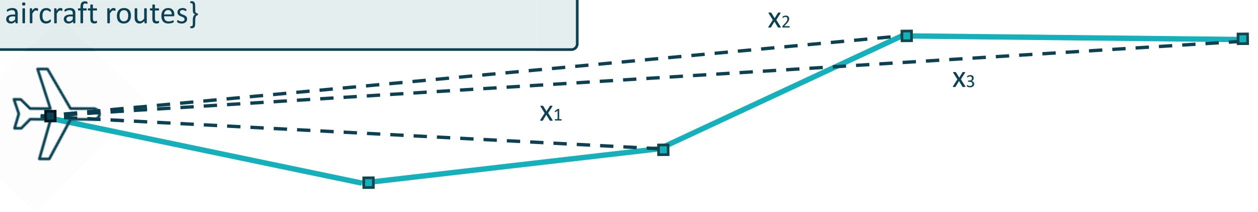
Given a particular airspace, our set of possible actions is:

discrete
{flight_1 > wp_2, flight_2 > wp_M, ... , flight_N > wp_1}*

finite
Limited by the number of flights and the maximum number of waypoints

We define a unique cost function for each airspace as the sum of the cost functions $C_i(x)$ associated to each aircraft.

$\boldsymbol{C_i(x)} \in \mathbb{R} \cup \{\infty\}$ where $\boldsymbol{x} \in$ {possible i-th aircraft routes}



02. Observation

C(x) are functions of finite domain. To minimize such functions, it is sufficient to check all cases and select the most optimal one. Even so, field knowledge reduces significantly the number of cases to check and avoids redundant and useless calculations.

03. Alternatives

Several alternatives with a plug and play approach are proposed, involving learning-based methods:

Reinforcement Learning (RL): The agent(s) is/are taught to take decisions optimizing a reward function in a given environment.

- **Single-agent**: The agent (i.e. the air traffic controller) learns to tell each aircraft which route to take, including the possibility to not take a DCT.
- **Multi-agent**: The agents (i.e. the aircrafts) learn to move as efficiently as possible along their possible routes in the airspace.

Data & inputs

Available data sources

Training

ENAIRE tracks - .csv files from January to June 2023 that include:

- Structure data in airspace sectors → 28k rows
- Flight plan data → 7M rows
- Flight plan waypoint data → 112M rows
- Data from the track points followed by the flights → 567M rows
- Data on the DCT that were taken→ 900k rows

Baseline algorithm outputs are used for training the RL models, serving as a supervised target for the reward and as an environment simulator.

Inference

iTEC data | 9000-type messages

Inference is performed in real time connected to iTEC (executions every 2 minutes)

Algorithm input

CSV

flightData

Column	Format
Timestamp	YYYY-MM-DD hh:mm:ss
Flight Id	String
Callsign	String
Aircraft type	String
ADEP	String
ADES	String
Eobd / Eobt	YYYY-MM-DD hh:mm:ss
Flight Level	Float
Initial Cruising Speed	Float
consumption	Float
flagBloq	Yes/No
blockReason	String

BADA →

CSV

flightPointsData

Column	Format
Flight Id	String
Position	(lat, lon)
Time	YYYY-MM-DD hh:mm:ss
isOnRoute	Yes/No
pointFix	Yes/No
isWayPoint	Yes/No
Initial Point	Yes/No

03

Methods

What techniques have been adopted? How have these been developed?

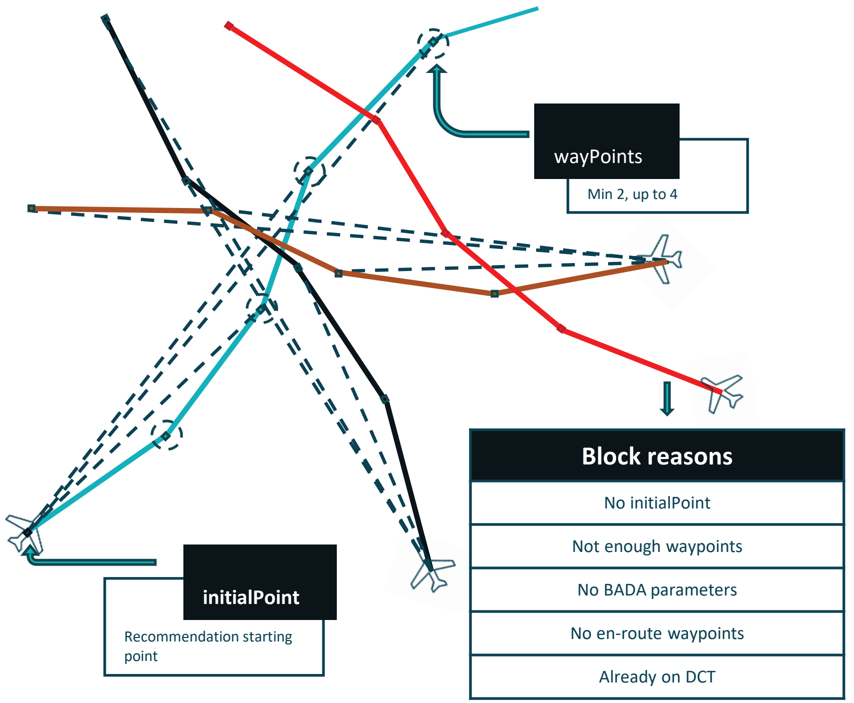
Baseline | Heuristics

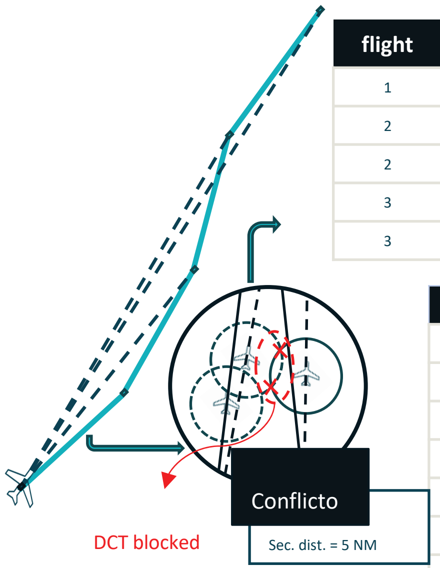
Simulators like BlueSky are not valid for the use case

Baseline as environment for the Reinforcement Learning approach

1. Data input processing (airspace, all the flights within area of interest)
2. Initial point generation, recommendation starting point
3. Given 4 waypoints from the initial point, 3 possible direct routes are considered, looking for the most optimized (cost effective)

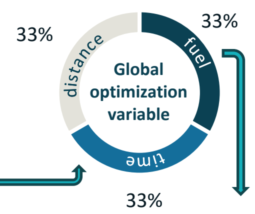
** Blocked DCTs are not considered here.*





flight	DCT
1	2
2	1
2	3
3	2
3	3

Optimization criteria



flight	DCT
3	3
2	3
1	2

Output	Original
Flight ID	Yes
Callsign	Yes
ADEP	Yes
ADES	Yes
EOBT	Yes
EOBD	Yes
Initial_route	No
Initial_point	Yes
Direct_point	No
Turn_direction	No

***Conflict** considered when two aircrafts are approaching within 5 NM

3.1. Cross-check against other aircraft trajectories and their possible DCTs starting from the most **cost-effective** one while **assuring independency**

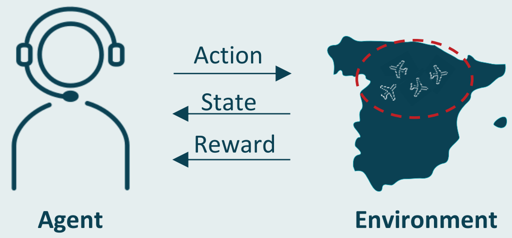
3.2. Computing efficiency regarding distance, time and fuel relative to the trajectory

3.3. Global optimization variable considering these criteria

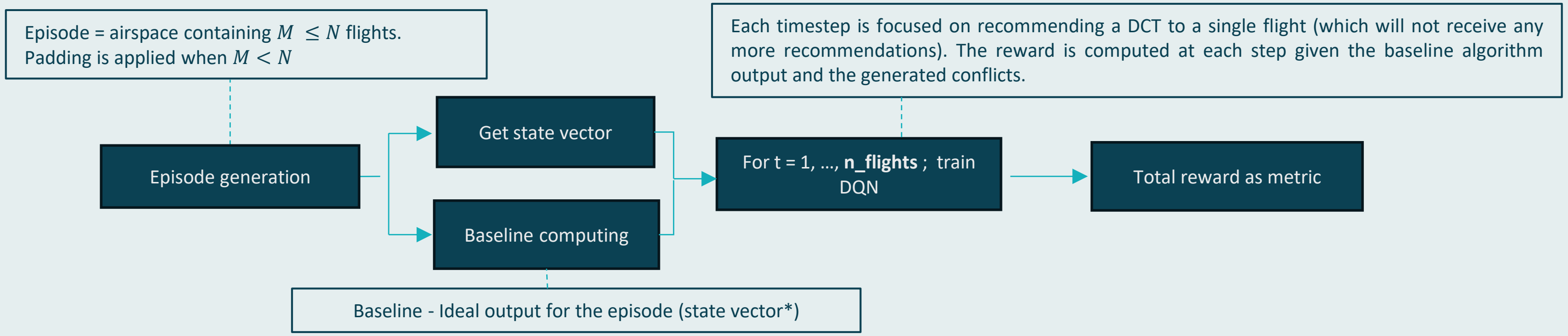
Learning and exploration | SARL

Single-agent Reinforcement Learning | DQN

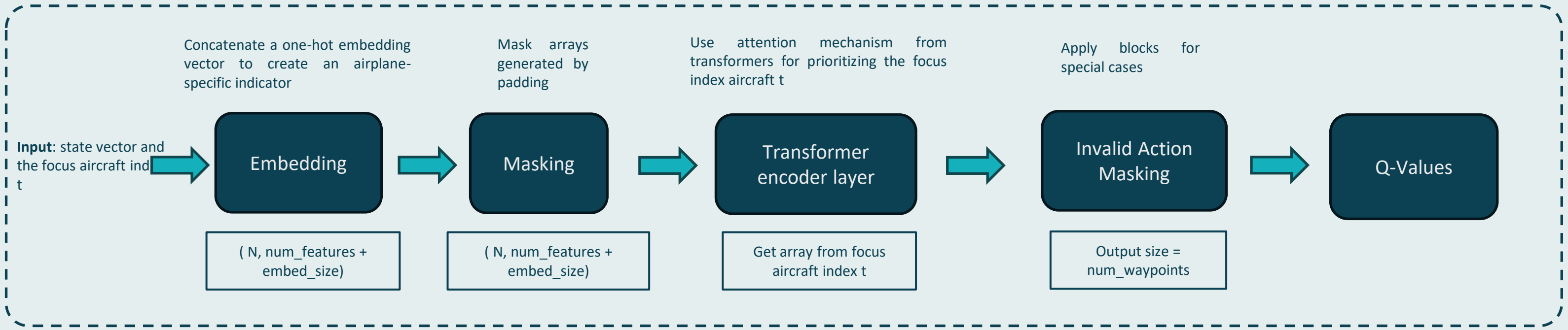
ATCO as agent whose mission is to learn to recommend tactical opportunities among the traffic within the airspace under test.



Let **N** be the maximum number of aircrafts in the airspace, **n_flights** the actual number of aircrafts in the airspace and **n_points** the maximum number of trace points per route (per aircraft). The DQN training pipeline proceeds as follows:



Q-network architecture:



*

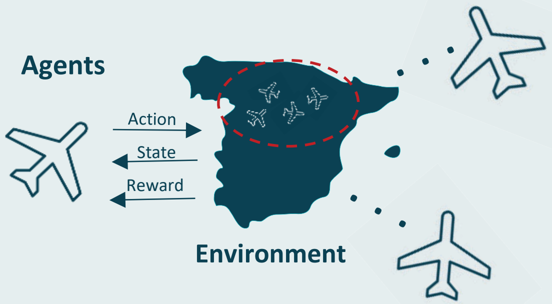
State Vector: 2D-array with trace data of each aircraft

[[lat(1.1), lon(1.1), time(1.1), isWP(1.1), lat(1.2), lon(1.2), time(1.2), isWP(1.2), ..., lat(1.n_points), lon(1.n_points), time(1.n_points), isWP(1.n_points)],
...
[lat(n_flights.1), lon(n_flights.1), time(n_flights.1), isWP(n_flights.1), ..., lat(n_flights.n_points), lon(n_flights.n_points), time(n_flights.n_points), isWP(n_flights.n_points)]]

Learning and exploration | MARL

Multi-agent Reinforcement Learning

Each agent – aircraft- in the airspace will select between the different route options (both routes and DCTs) within a grid space (translated to real coordinates) to reach in the fastest way to its destination, avoiding conflicts (keeping distance with other aircrafts)



Actions:

First action: Route selection

Remaining actions:

1: Move forward on route

2: Slow down

3: Turn left

4: Turn right

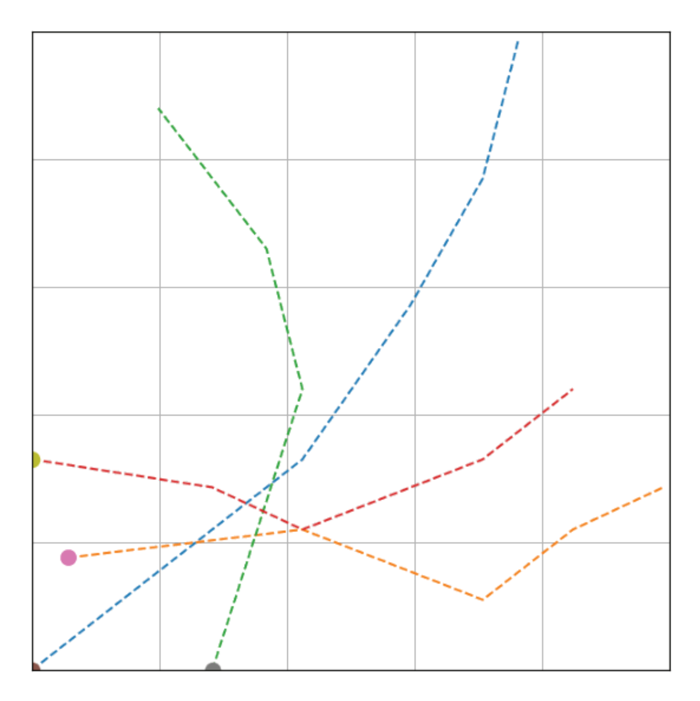
States:

Location of the aircraft at each instant of time within the grid space.

Completion condition: all routes are completed or a conflict is detected.

Reward:

- When selecting a route the reward is computed accordingly with a substantial bonus if valid, or a severe penalty if not
- Moving forward on their routes, agents get additional rewards for progressing and staying close to the predefined path
- Penalties are incurred when slowing down or deviating
- Rewards are incurred for keeping close to their route
- Hard penalty when getting too close to other agents, with penalties becoming more and more severe
- Reward scaling factor that ensures the enforcement of the original route



Algorithms considered

- Proximal policy optimization (**PPO**): An on-policy actor-critic algorithm that uses a clipped surrogate objective to ensure stable, incremental policy updates.
- Twin Delayed Deep Deterministic Policy Gradient (**TD3**): An off-policy actor-critic algorithm for continuous action spaces that reduces overestimation bias by using twin critics and delayed policy updates.

04

Pipeline and integration

How are these processes implemented into a complete processing stream?

Where is the data received from and where is it sent to?

Pipeline

01. Data ingestion

- SWIS service – 9000-type messages
- .txt to .xml conversion

02. Parsing and preprocessing

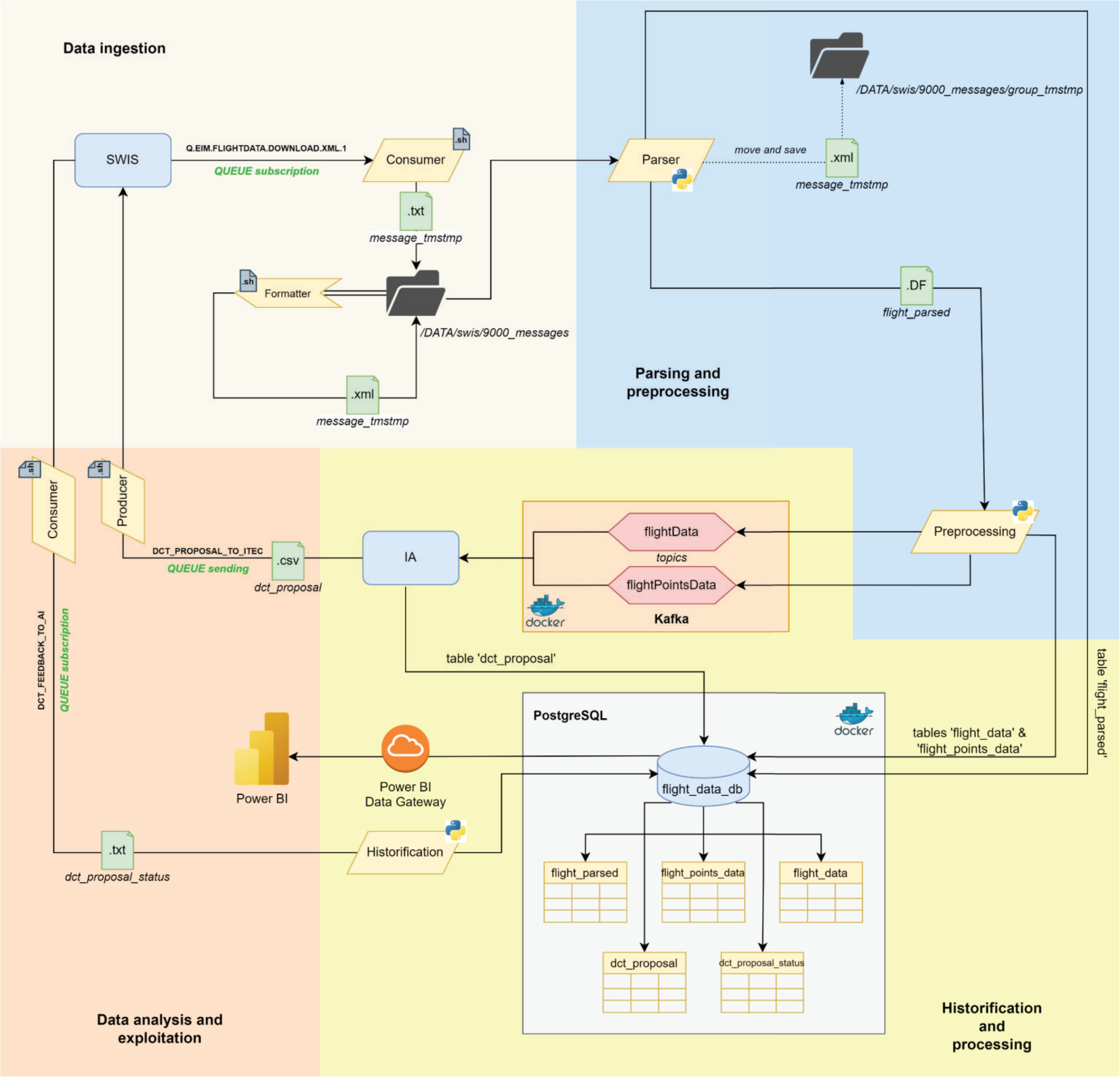
- XML cleaning, selecting only data of interest, which is stored in a PostgreSQL containerized service
- Apply blocking filters to extract valid flights, populating tables to be sent to the model (also storing them)

03. Processing

- Table reading from Kafka containerized service
- Algorithm processing, sending the result to SWIS
- Data storing in the corresponding tables

04. Data analysis and exploitation

- Feedback coming from ATCO via SWIS stored in PostgreSQL
- HMI through Power BI Service enabling post-analysis

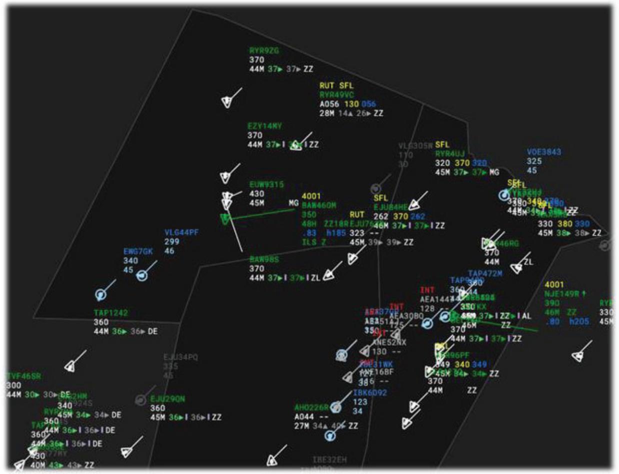


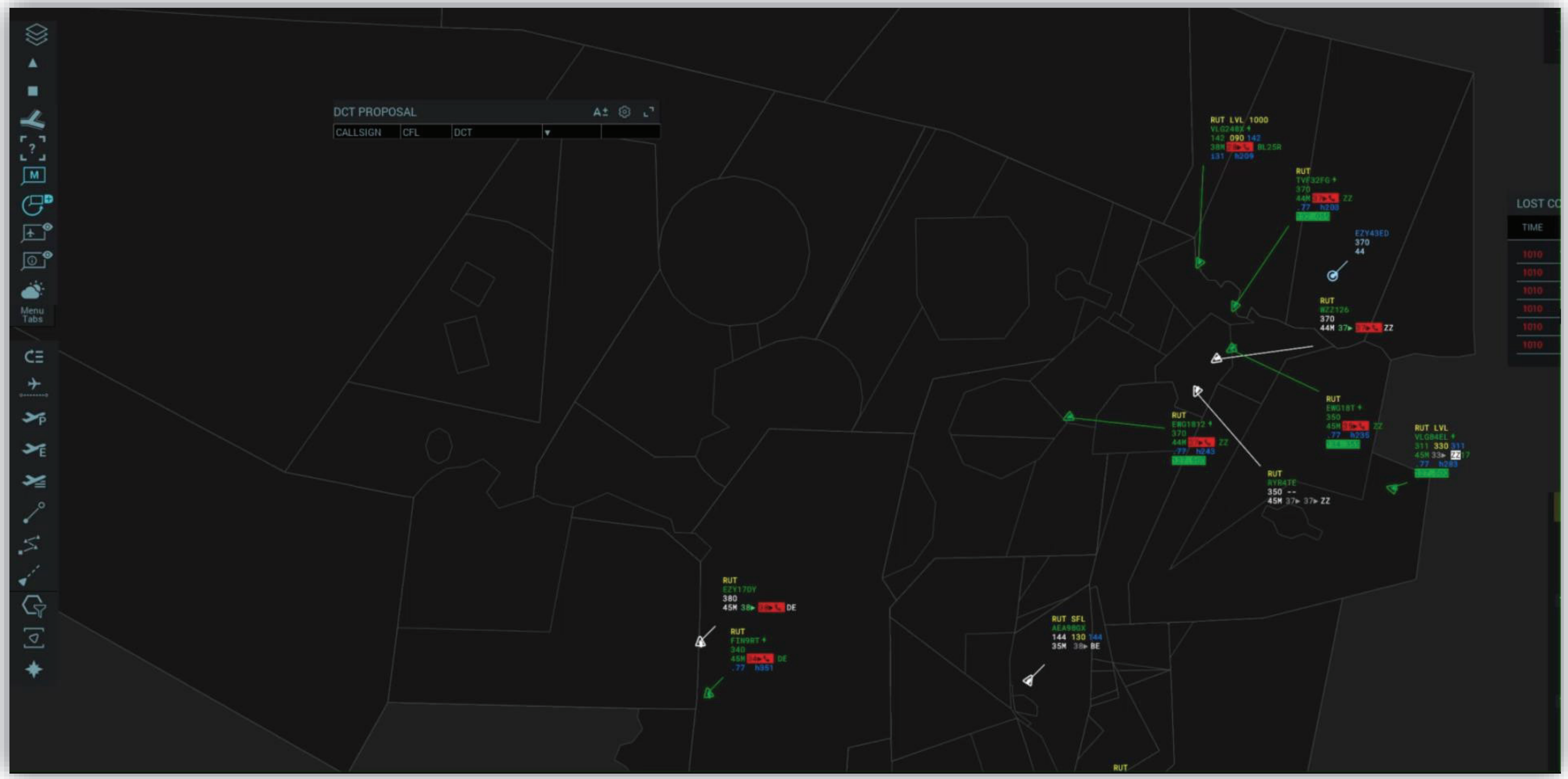
iTEC integration

End-to-end tests in real machines | Full communication

DCT PROPOSAL

CALLSIGN	CFL	DCT		
NJE149R	390	BANEV	ACCEPT	REJECT
BAW460M	350	ORBIS	ACCEPT	REJECT





	flightid	callsign	adep	ades	eobt	eobd	initial_route	initial_point	direct_point	turn_direction
0	8315	BAW460M	EGLL	LEMD	11:28:00	2025-01-31	[DELOG, VADOX, SNR, TITAN, RATAS, NEA, NONTU, ...	(-3.975, 42.9578)	ORBIS	Default
1	8366	TAP1242	LPPT	LKPR	11:11:00	2025-01-31	[NUBLO, RONSI]	(-5.2523, 42.2111)	RONSI	Default
2	8351	NJE149R	LFPB	GMTT	11:04:00	2025-01-31	[ARVID, PPN, NOLSA, ALEPO, VASUM, GARVU, BANEV...	(-2.0778, 42.1711)	BAN	Default

INDRA

05

Results

How is the performance of these systems evaluated?

How successful have they been adopted by ATCOs?

Overview | Post-analysis dashboard

KPIs

Metric samples of proposed DCTs

DCTs

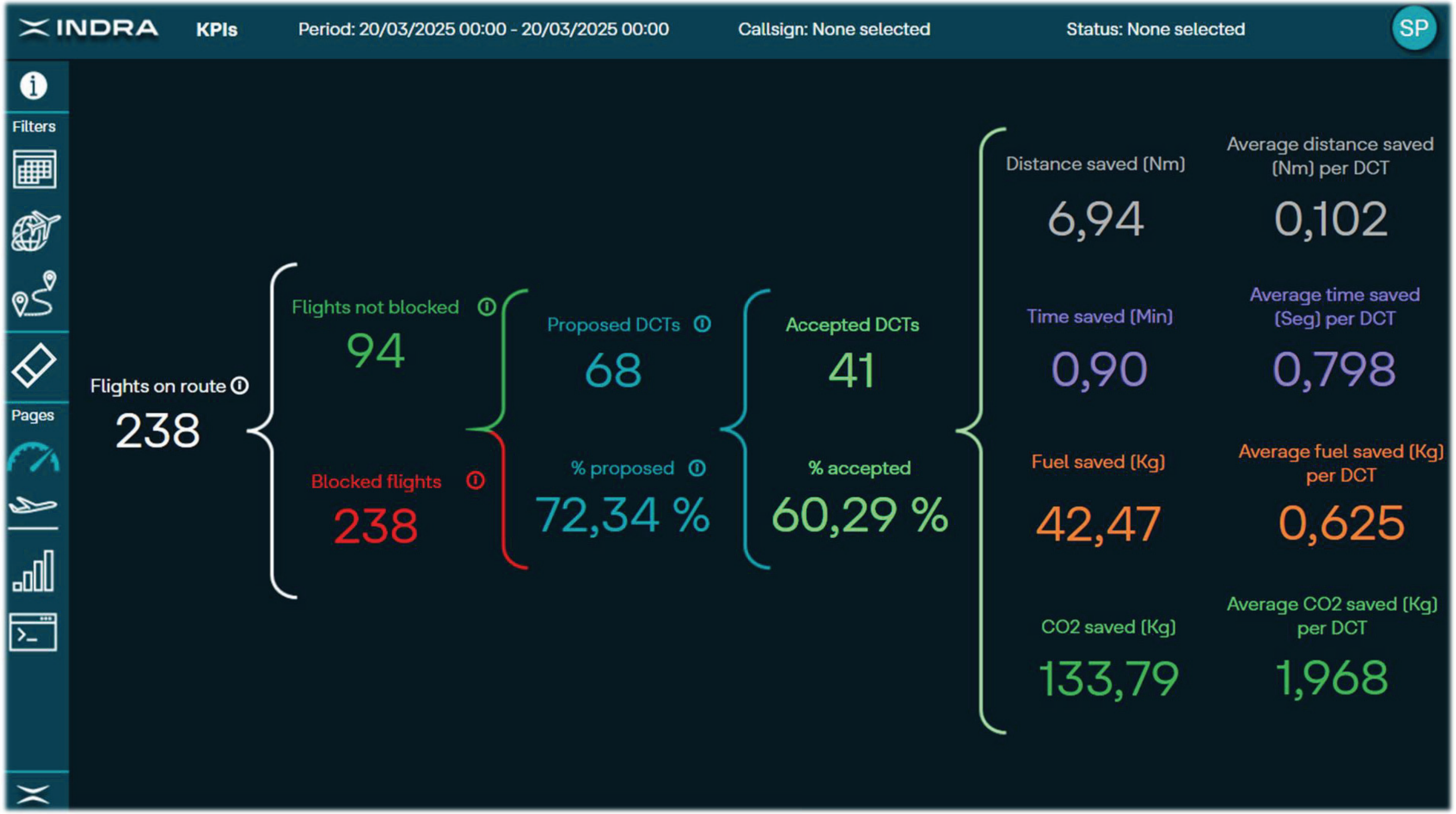
Evolution of the number of DCTs proposed on the time axis and their acceptance status.

Accepted DCTs

Shows the routes where more DCTs proposed have been accepted and the evolution of the assumed efficiency savings on the time axis.

Algorithm

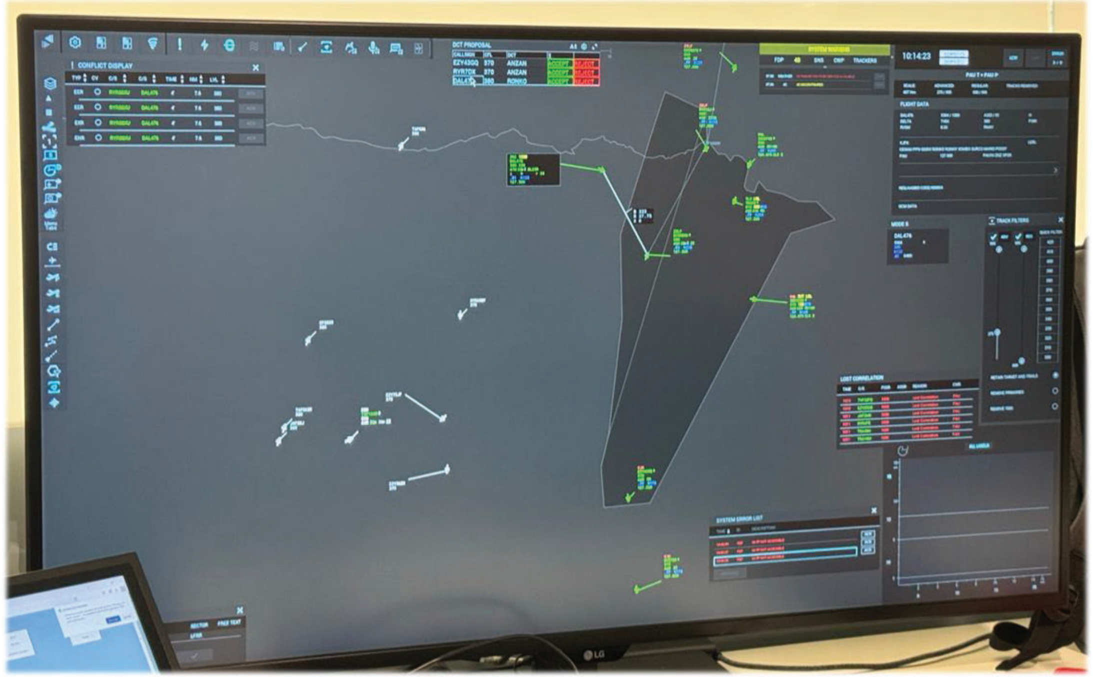
Shows algorithm execution metrics such as execution time and variables that have impacted it



iTEC scenarios – SESAR Dry Run

Main insights - feedback and proposals

- ATCOs have seen this **tool** as **very promising** for improving traffic efficiency in the sectors under test
- Selecting **DCT waypoints from the flight plan** waypoint list is in accordance with regulations
- Several proposals as **increasing** waypoints **horizon** or including **adjacent sectors** have been successfully evaluated
- Minimum savings threshold needs to be considered, **maximizing cost-benefit**
- Including past **ATCO feedback** about DCTs would be useful to know main hotspots and coordination techniques in the sector under test
- Considering DCT to a traffic **already in DCT** would be desirable (according sector constraints)
- **Flight level** filtering: consider **climb** and **descent**
- **Accepting** and **rejecting** DCTs could be more **interactive**
- DCT proposal table would need some add-ons to better **fit into the regular workflow**, such as marking the aircraft and the proposed route when hovering



06

Conclusions and future work

Which are the conclusions of these findings? What are the next steps for further research?

Conclusions

Objective

- Tactical opportunities recommender system development and deployment
- Initial scope: several constraints and limited air volumes
- Data: ENAIRE - Airspace, tracks, FP, DCTs

Approach and methodology

Baseline as environment for RL
- **Heuristics**
- Considering **conflicts**
- Maximizing **efficiency**
  - **Time**
  - **Distance**
  - **Fuel**

Single-agent Reinforcement Learning
- **ATCO** as single agent looking for tactical opportunities
- **DQN algorithm**
- Transformer encoder & invalid action masking

Multi-agent Reinforcement Learning
- **Aircrafts** as **agents** selecting the best **route** option within a grid space to **optimize** their way to the destination avoiding **conflicts**

Findings

Having explored ATC environments and gyms, it is necessary to design and develop a **baseline algorithm** that acts as environment for further develop learning and exploration techniques

Given the constraints already discussed for the use case, this **baseline** algorithm can compute **DCT recommendations** while avoiding simple conflicts within the airspace

Baseline algorithm has proven to be capable of finding tactical opportunities in **real operation** under several scenarios based on real traffic, getting very **positive feedback** from ATCOs

Two approaches based in Reinforcement Learning are currently being explored, identifying several challenges:

- SARL
  - Selecting recommendations among traffic in the airspace
  - Modelling reward
- MARL
  - Gap controller – aircraft
  - Toy example
    - Consistency
    - Transition to real scenario

Future work

Continue exploring Reinforcement Learning approaches

- SARL
- MARL

→

- New data
- New inputs
- Airspaces & horizons

- Algorithms
- Architecture
- Multi-FL environment

- Imitation learning
- Human-in-the-Loop RL

Improving integration with iTEC to increase ATCO acceptance

Improving baseline algorithm

- Set minimum saving threshold | cost-benefit
- Add airspaces & horizon
- Enhance blocking of refused proposals
- Multi-FL analysis & deferred proposals

Exploring hotspots & conflict avoidance

- DCT to solve conflicts within the airspace
- Lookahead for entry traffic and downstream opportunities
- Multi-FL conflict resolution

INDRA

Tech for trust