Human-interpretable input for Machine Learning in Tactical Air Traffic Control

Ir. Tiago Nunes; Dr. Ir. Clark Borst; Dr. Ir. Erik-Jan van Kampen
Delft University of Technology

Carl Westin, PhD
Linköping University

Brian Hillburn, PhD
Center for Human Performance Research

Online, 8 of December 2021

Founding Members
EUROPEAN UNION EUROCONTROL

Contents

MAHALO Partners

SESAR JOINT UNDERTAKING

TU Delft

Technical University of Delft

CHPR

Center for Human Performance Research

LINKÖPING UNIVERSITY

Linköping University

deepblue consulting & research

LFV

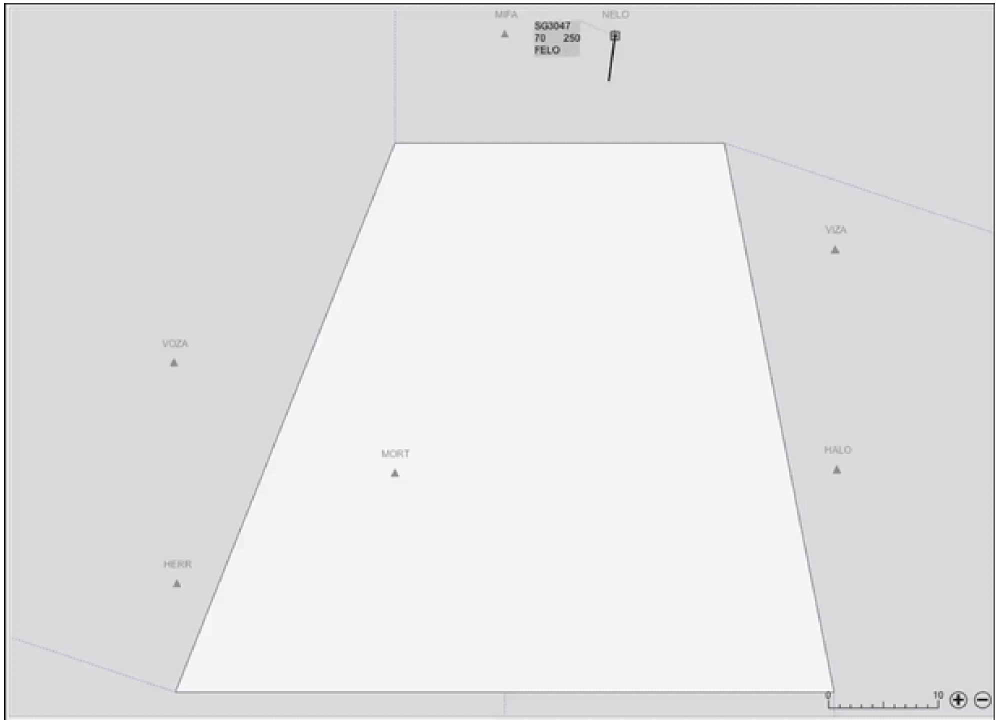
AIR NAVIGATION SERVICES OF SWEDEN

EUROCONTROL

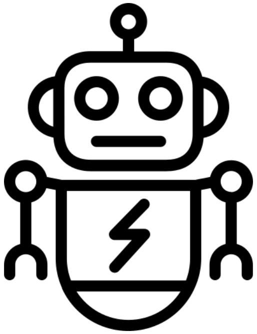
ANACNA

Automation acceptance in ATC

Acceptance



Authority and autonomy



How to build ML?

A simple but profound question: *Should we build automation that is CONFORMAL or TRANSPARENT?*

Conformance— does automation seem to match human strategies?
Transparency— is automation's inner process explainable to human?

		TRANSPARENCY	
		Low	High
CONFORMANCE	Low	**Stupid automation:** *"It's doing a strange thing, and I don't understand why…"*	**Peculiar automation:** *"It's doing a strange thing, but I understand why…"*
	High	**Confusing automation:** *"It's doing the right thing, but I don't understand why…"*	**Perfect automation:** *"It's doing the right thing, and I understand why…"*

Objectives

- Create / integrate / demonstrate a **hybrid ML capability** for detecting/resolving ATM conflicts

  Hybrid ML model for CD&R via
  - Supervised Learning (SL)– deep learning– to detect / classify conflicts
  - Reinforcement Learning (RL)– rule based– to resolve conflicts

- Develop a control model and **UI** to foster transparency

  Build a UI to foster transparency & understandability of ML solutions

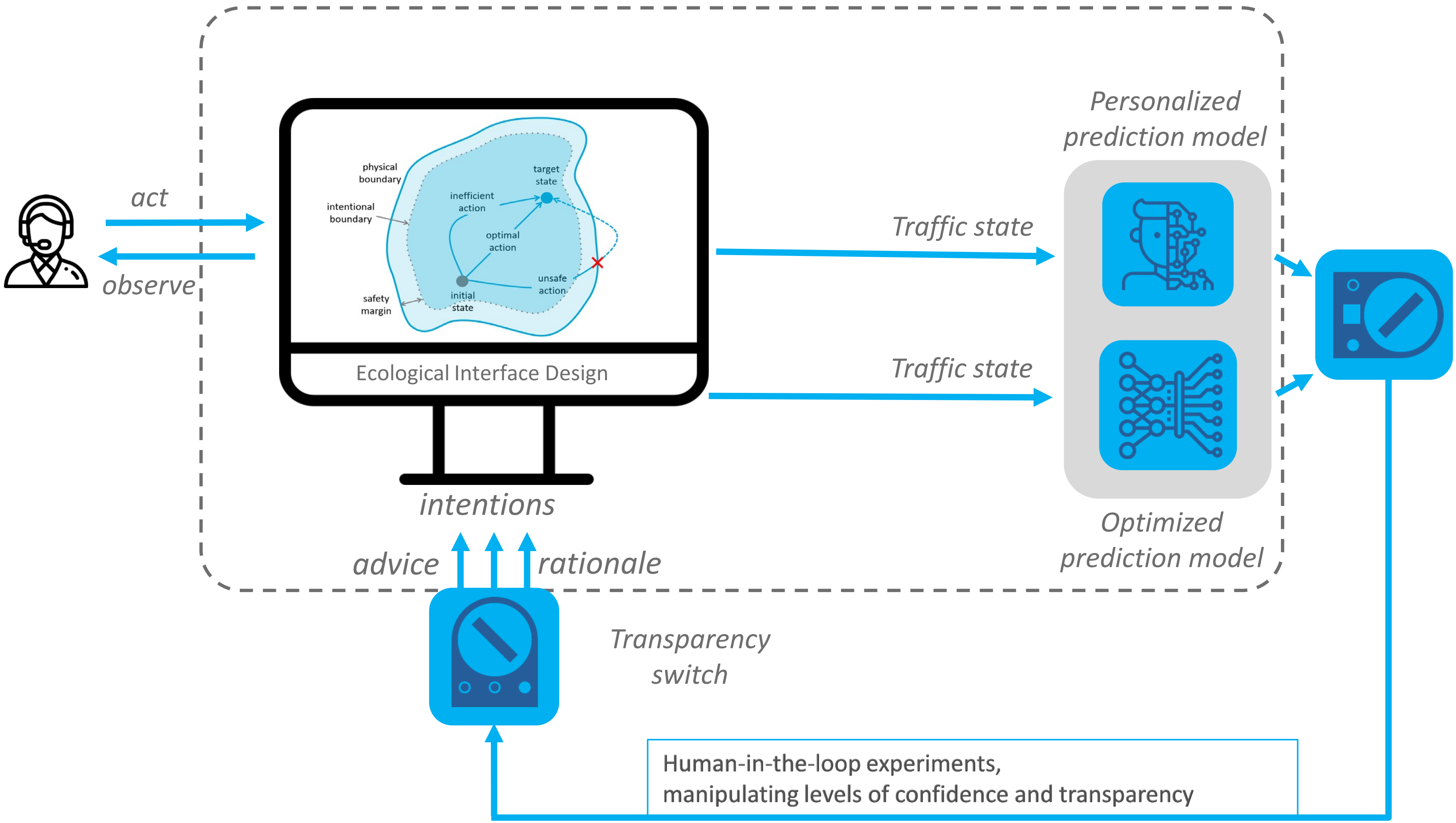
- Experimentally **evaluate impact** of conformance and transparency on performance

  HITL sims: How do Conformance, Transparency, Complexity impact trust, acceptance, etc of ML solutions?

- Define **framework** to guide development of AI systems

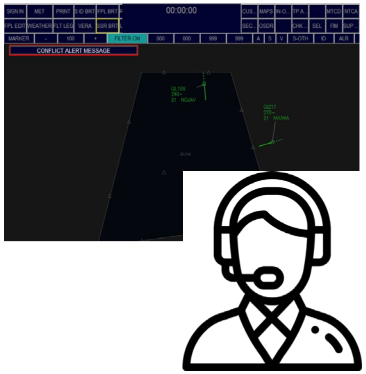
  Lessons learnt: How to build ML for ATM?

MAHALO Overview

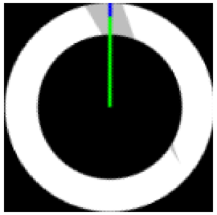


Supervised Learning model

Sector X ATC simulator



Conflict scenarios



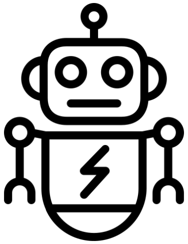
Solution Space Diagram

Human conflict resolutions

Supervised Learning Model

Training a model

AI-supported ATM system



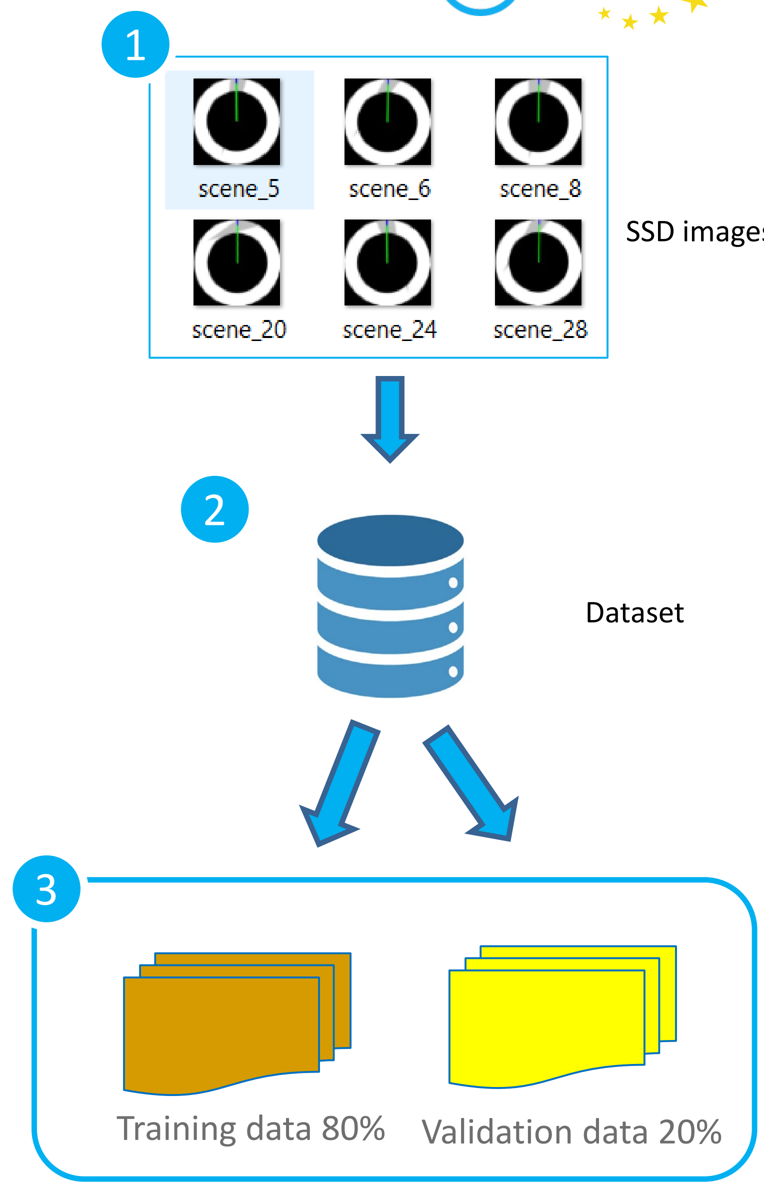
Heading advisories

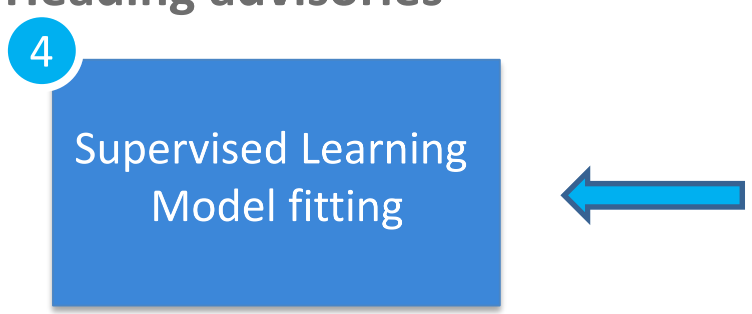
Human AI collaboration

Supervisory controller

Our approach

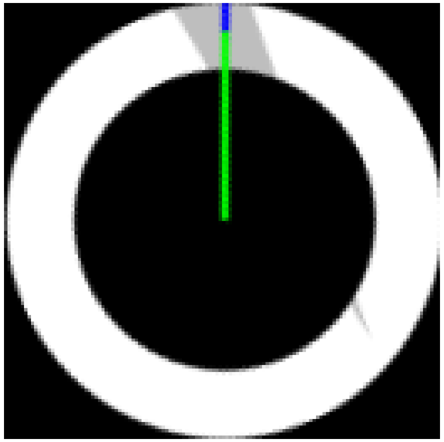
- Train a model on multiple conflicts and ATCO resolutions
- SectorX ATC research simulator
- AI sees what the human sees (same information/situation)
  - Solution space diagram (SSD)
  - Images from the radar screen
  - Heading advisories





Our approach

- Google Tensorflow
  - SSD input -> Heading advisories
- Convolutional neural network
- Data from ATCO students (Sim1)
- Training pipeline



Rooijen, S. J. Van, Ellerbroek, J., Borst, C., & Kampen, E.-J. Van. (2019). Conformal Automation for Air Traffic Control using Convolutional Neural Networks. In Thirteenth USA/Europe Air Traffic Management Research and Development Seminar (pp. 1–10), Vienna.

Q-Learning for MVP

- MVP = Modified Potential Value
- MVP has some tunable parameters:
    - Lookahead distance
    - Safety Margin
- A Q-Learning agents learns the optimal parameters for the MVP method given a certain conflict scenario
- A conflict scenario is classified by:
    - Conflict angle
    - Time to closest point of approach (tCPA)
    - Distance at closest point of apporach (dCPA)

MVP

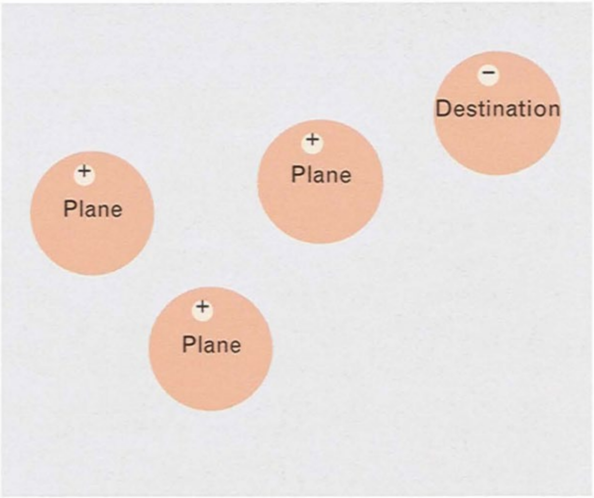


FIGURE 2. Air traffic model based on voltage potential fields. The model exhibits two critical features of successful ATC: aircraft (positively charged particles) tend to maintain separation between each other (because of mutual repulsion) while moving toward their destination (because of attraction to the opposite charge).

From: Eby, M. S. (1994). A Self-Organizational Approach for Resolving Air Traffic Conflicts. Lincoln Laboratory Journal.

Deep Q-Learning from Demonstrations (DQFD): why use demonstrations?

RL tends to be data inefficient

Cumbersome data collection process. Requires a lot of interaction with environment

RL has bad initial performance

Dangerous to use at the beginning of training (if done online in a physical environment)

Q: How do we make a RL agent that is **data efficient** and has a **decent initial performance**?

Deep Q-Learning from Demonstrations (DQFD): why use demonstrations?

A: Use **representative expert demonstrations**

Representative of the state-action space. Encompass diverse scenarios

A set of scenarios solved by a human expert or a simplified 'good-enough' algorithm

What is defined as a sample/demonstration?

↓

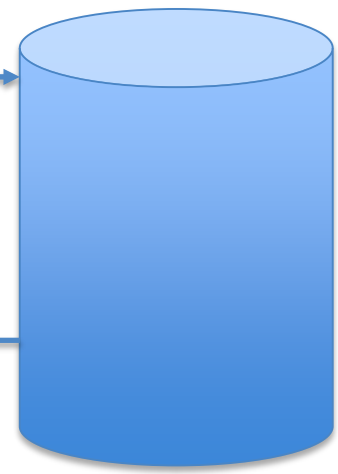
(s, a, r, s') Tuple

Deep Q-Learning from Demonstrations (DQFD): Pre-training

Collect demonstrations. Put them into the **replay buffer**

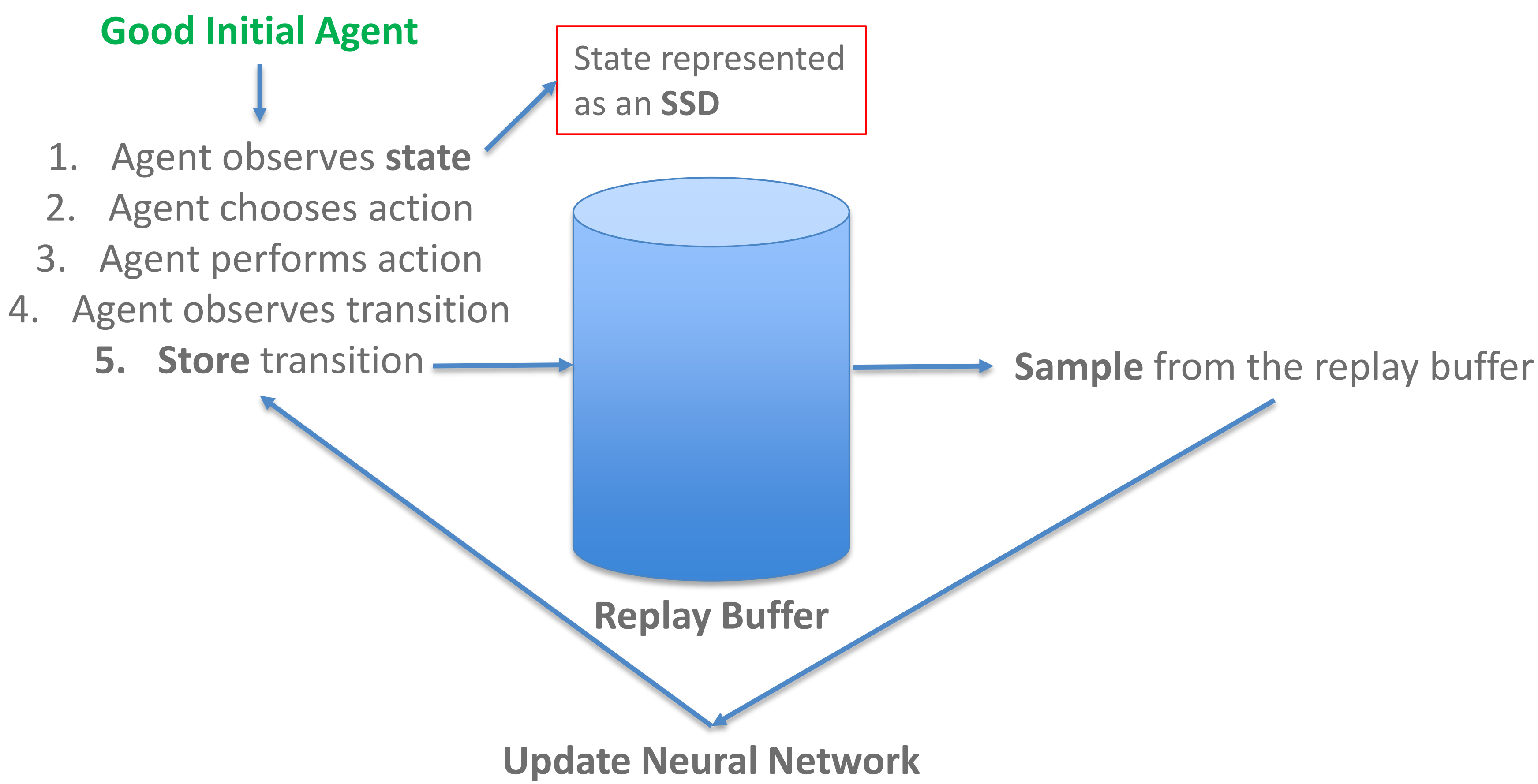
Pre training:
Agent learns by sampling from the demonstrations for *k* time-steps

Good Initial Agent



Replay Buffer

Deep Q-Learning from Demonstrations (DQFD): Training



Deep Q-Learning from Demonstrations (DQFD): Handling Demonstrations

- **Prioritized sampling** is used for the Demonstrations.
- Samples are replaced by newer ones on a **FIFO** basis.
- Should demonstrations be replaced?

- A1: Yes, replaced the same way as other transitions
- A2: No, demonstrations are always kept

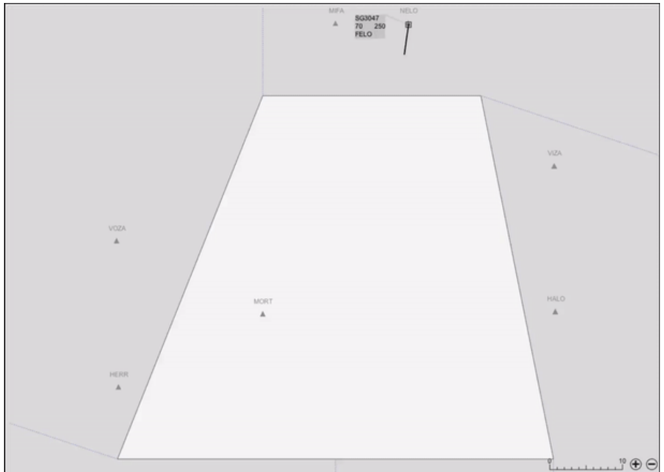
Reward function

- In RL a reward function is used to *define* objective optimality

- An optimal policy maximizes the expected sum of future rewards

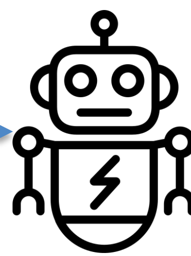
- How to design this reward function???
    - Collision avoidance
    - Extra distance flown
    - Extra time used
    - Fuel usage
    - Workload
    - Etc.

Resulting automation



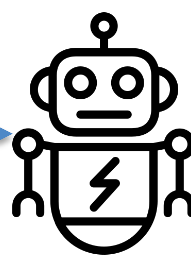
Observe

Conformal



Suggestion coherent with **previous solutions** by the ATCO

Optimal



Suggestion based on **best performance**

Results

- Current results relate to the development of the different ML agents, interfaces and presentation within SectorX
- Proof of concept simulation (Sim1) successfully performed
- Sim 2a currently underway (experiment involving ATC professionals)

Results: Preliminary findings

- Sim 2a pre-test underway:
  - ATCOs find the traffic scenarios complex and a bit unrealistic without being artificial
  - ATCOs gave additional feedback on the interface
  - No boredom reported so far

Conclusions

- Article presented an overview of the current approach being taken by the MAHALO team
- A combination of SL and RL agents are used to achieve conformal and “optimal” performance, respectively
- Main focus is to study effects of conformance and explainability on acceptance
- A Human-Interpretable input is used for the ML agents
- Currently in the HITL stage of the project

Future Work

- Surpassing current limitations:
    - Larger action space
    - More complex traffic scenarios and resolutions (ALT+HDG+SPD)
    - Inclusion of adverse weather
    - Expansion of work to other interfaces (Travel space representation, for example)

Human-interpretable input for Machine Learning in Tactical Air Traffic Control

Thank you very much for your attention!

This project has received funding from the SESAR Joint Undertaking under the European Union's Horizon 2020 research and innovation programme under grant agreement No [number]

t.m.monteironunes@tudelft.nl

stefano.bonelli@dblue.it

http://mahaloproject.eu/

SESAR
JOINT UNDERTAKING