Towards Conflict Resolution with Deep Multi-Agent Reinforcement Learning

Ralvi Isufaj, David Aranega Sebastia, Miquel Angel Piera

Universitat Autònoma de Barcelona

01

Introduction

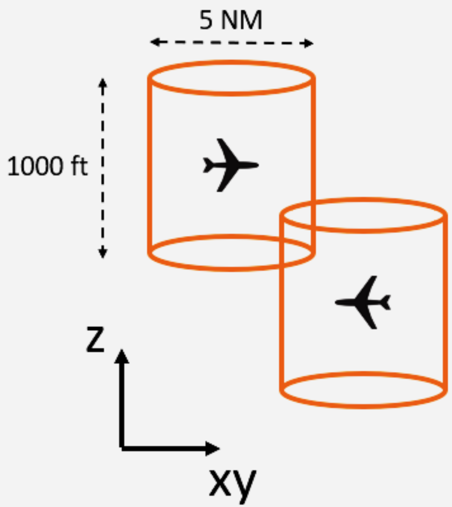
Introduction

ATM ensures that aircraft fly safely

Safety through minimum pairwise separation

Guaranteed by controllers at tactical level

Decision support tools as a pillar of automation



Plenty of existing algorithms

No consideration to factors that affect quality of solutions

Limiting assumptions about airspace

Why Reinforcement Learning?

MAIN OBJECTIVE:

Improve quality of conflict resolutions based on a combination of multiple factors

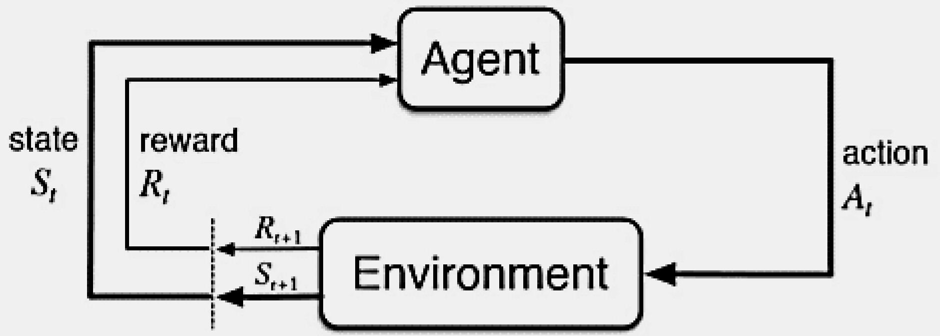
02

Theoretical Background

Reinforcement Learning

Markov Decision Process:

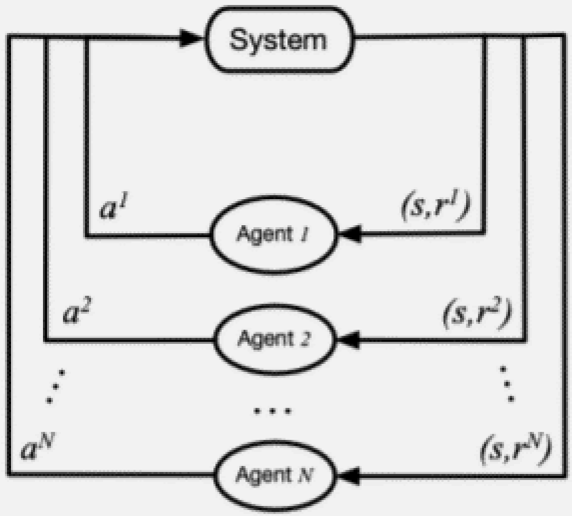
- Agent
- Environment
- Policy
- Reward



Multi-Agent RL

Extend MDP to Markov Games:

- Agent observations
- Cooperative/Competitive/Mixed
- Individual reward is a function of all policies



Zhang, Kaiqing, Zhuoran Yang, and Tamer Başar. "Multi-agent reinforcement learning: A selective overview of theories and algorithms." *Handbook of Reinforcement Learning and Control* (2021): 321-384.

Deterministic Policy Gradient Methods

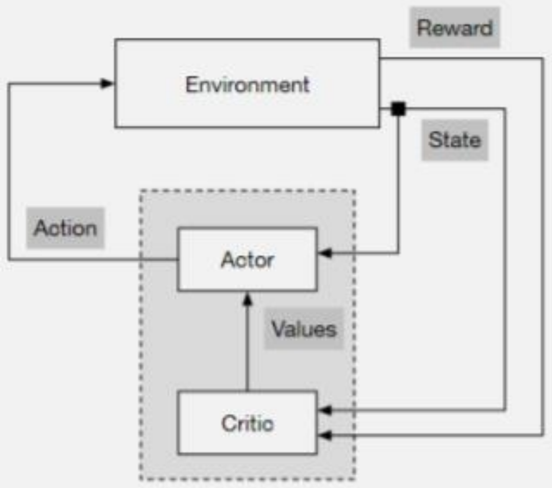
Optimize parameters of the policy so the best action is taken:

$$\theta_{i+1} = \theta + \alpha \nabla J(\theta_i)$$

Target networks to stabilize training

Possible to have continuous action spaces

Actor-Critic paradigm:



03

Conflict Resolution as Multi-Agent RL

Conflict Resolution as Multi-Agent RL

Formalize the Markov Decision Process:

- Action Space
- State Space
- Reward Function
- Goal State
- Partially Observable

Learning Algorithm:

- Multi-Agent Deep Deterministic Policy Gradient (MADDPG)

Scenarios & Simulations

Action Space

Continuous action space:

- Heading chages up to 45
- Speed changes [v – 6%; v + 3 %]

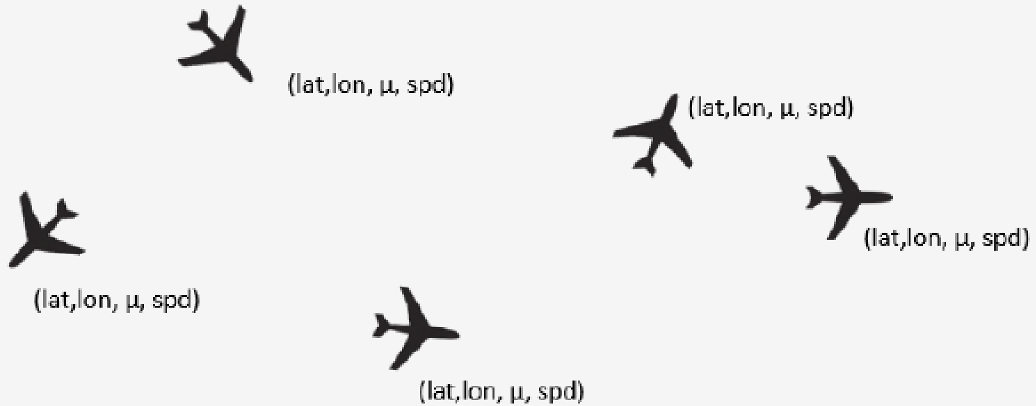
Output:

$$a = [\sin(\alpha), \cos(\alpha), spd]$$

$$\alpha = \frac{\sin(\alpha)}{\cos(\alpha)}$$

Gaussian noise added to ensure exploration

State Space


(lat,lon, μ, spd)
(lat,lon, μ, spd)
(lat,lon, μ, spd)
(lat,lon, μ, spd)
(lat,lon, μ, spd)

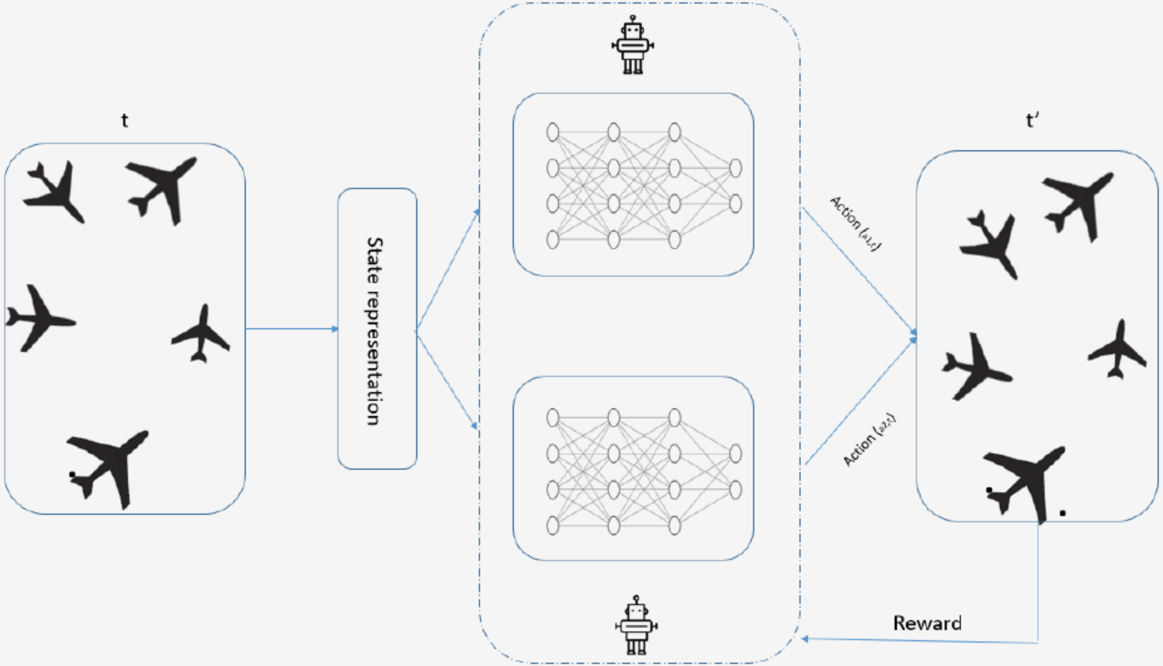
Reward Function

Factors:

- Time until LOSS and CPA
- Difference from track and optimal speed
- Fuel consumption (OpenAP)
- New conflicts
- Airspace complexity (Aerial Ecosystems)

Final reward is a weighted linear combination of all factors

MADDPG



Scenarios & Simulations

Overview:

- Episode lasted 20 minutes
- Pairwise conflicts with surrounding traffic
- Agents take an action every 15 seconds
- Data Augmentation
- Around 1000 training examples, 195 test scenarios

Simulation Environment:

- BlueSky simulator
- Open source
- Trained for 2500 episodes

04

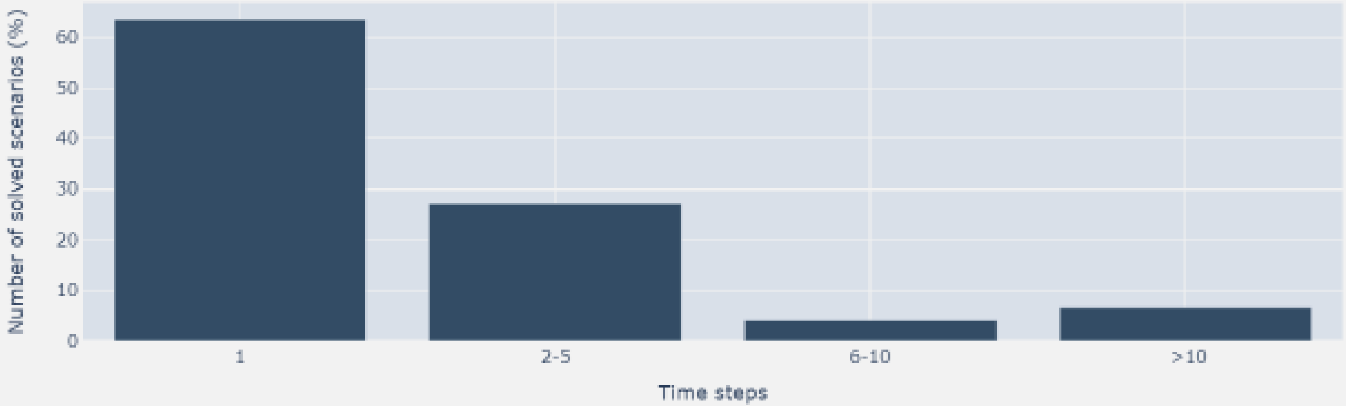
Results & Conclusions

Conflict Resolution

Tested on 195 scenarios

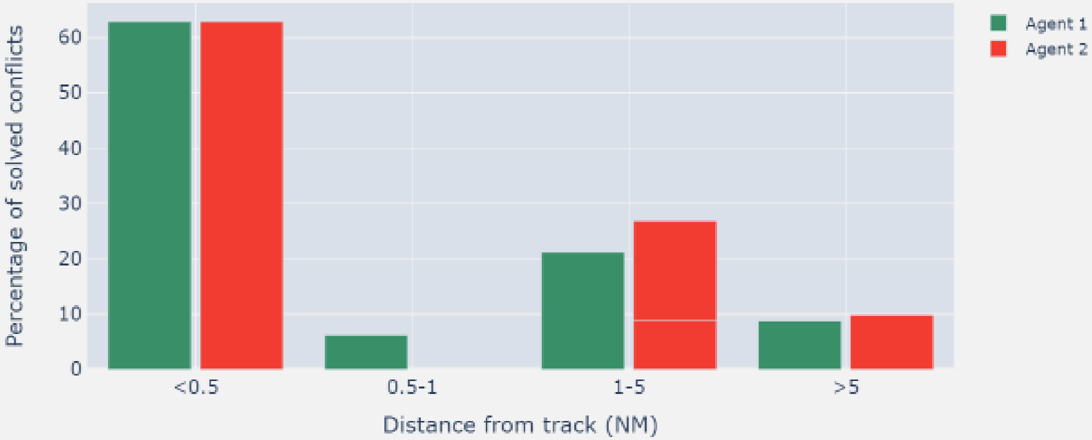
93% Conflicts solved

No new conflicts created

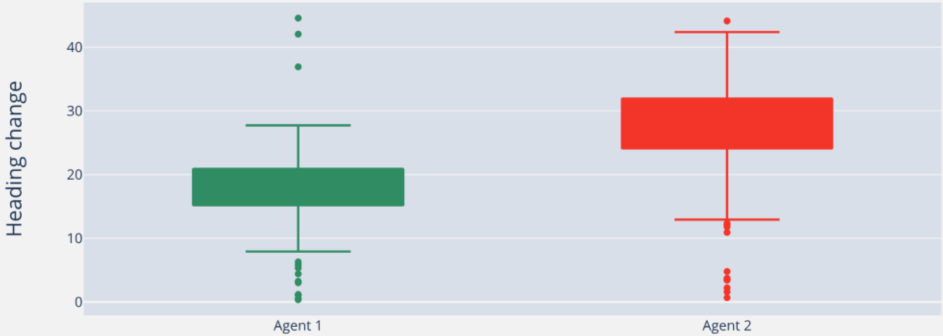


Relation between time to LOSS and time to solve conflict

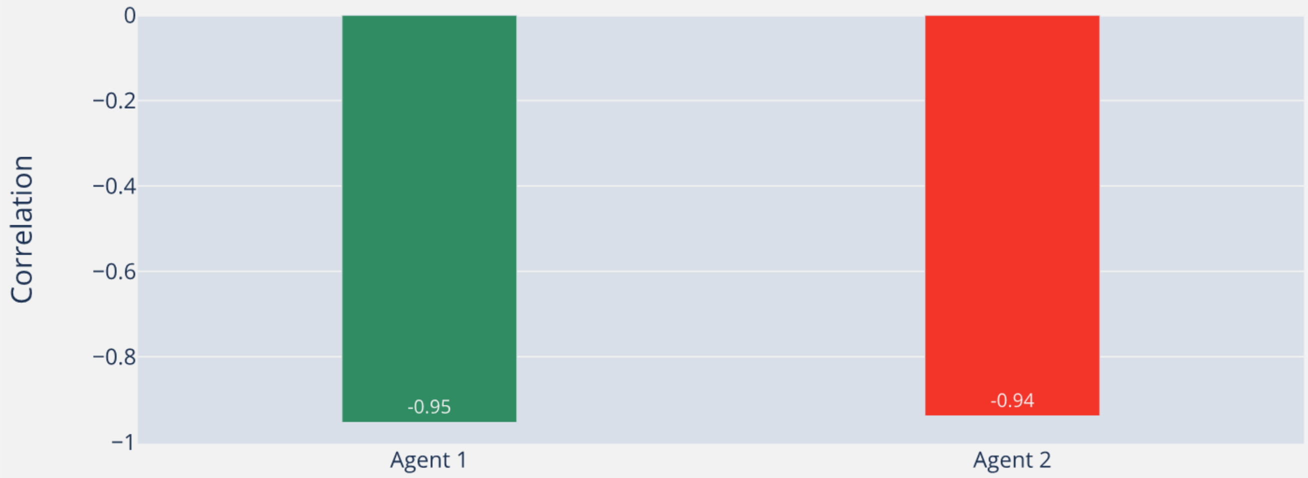
Displacement from track



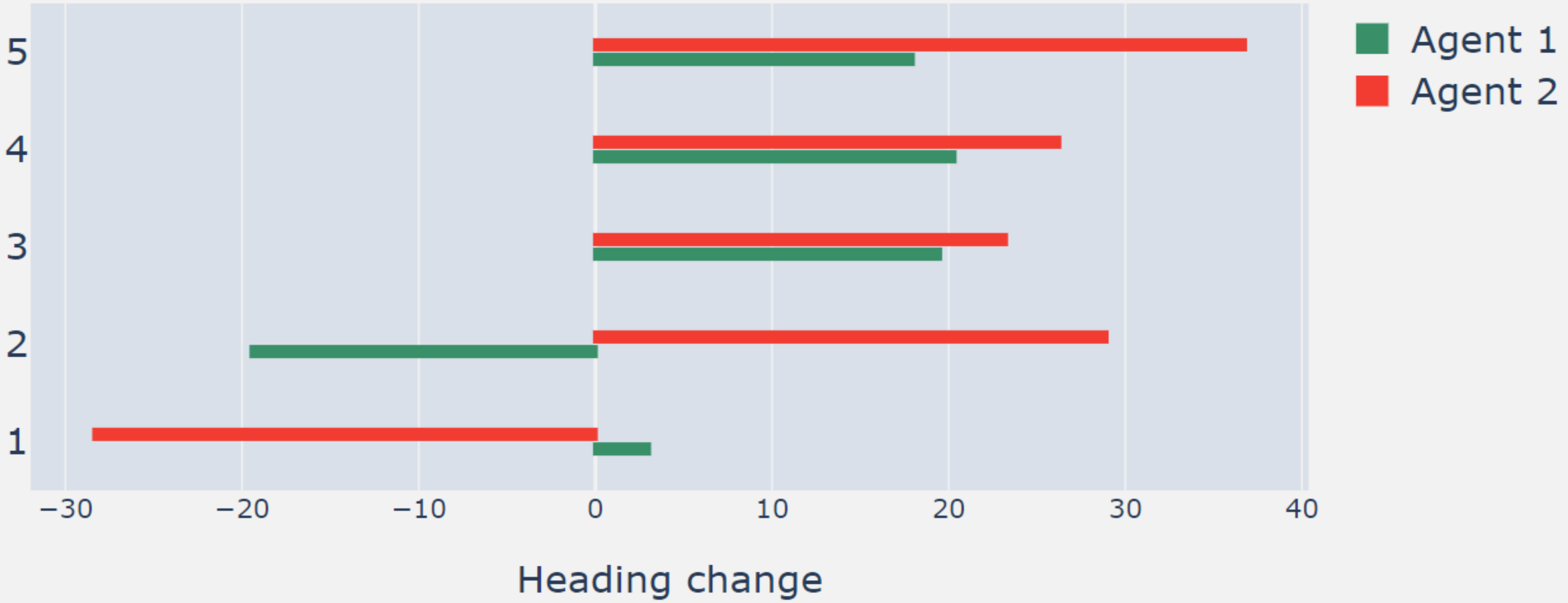
Agent behaviour



Correlation between final reward and time steps



Agent behaviour



Conclusions

- Pairwise conflicts modelled as multi-agent RL
- Novel reward function that attempts to tackle quality of solutions in addition to solving conflicts
- Trained and tested on real traffic scenarios (with data augmentation)
- 93% of conflicts solved
- Agents can learn desirable behaviour

Challenges & Future steps

- Improve counterintuitive behaviour
- Investigate competitive or mixed agents
- Expose models to expert knowledge
- Graph neural networks
- Explainability

Acknowledgment

This research has received funding from the SESAR Joint Undertaking under the European Union’s Horizon 2020 Research and Innovation Programme under grant agreement No 783287. The opinions expressed herein reflect the authors’ view only. Under no circumstances shall the SESAR Joint Undertaking be responsible for any use that may be made of the information contained herein.

THANK YOU!