Generalized Aircraft Collision Avoidance Through Deep Reinforcement Learning

ACAS1: Dharmesh Tarapore, Vincent Wahl, Kasim Patel, Shantanu Bobhate
dharmesh@bu.edu, vinwah@bu.edu, kasimp93@bu.edu, sbobhate@bu.edu
Boston University CS 542

Abstract

The gradual integration of ADS-B1 into the National Airspace System (NAS) has spurred research into its possible use in collision avoidance systems. While most new systems have shown to be much safer than the existing TCAS II (Traffic Alert and Collision Avoidance System), all of them make comparable assumptions about aircraft capabilities, thus restricting their applicability to highly specific classes of airplanes [1].

In this project, we develop a model derived from one such solution and generalize it to almost all powered aircraft by making conservative initial assumptions about their capabilities and then improving them by extrapolating from state action pairs. This modification allows us to provide a truly comprehensive collision avoidance system that can be used in most powered airplanes.

Introduction

The federally mandated Traffic Alert and Collision Avoidance System (TCAS II) for transport category aircraft has proven remarkably effective at averting mid-air collisions by providing pilots with timely alerts to resolve imminent threats. However, strong assumptions about aircraft capabilities made in the TCAS II logic prevent it from being used in general aviation, where the risk of a mid-air collision is significantly higher.

We investigate the feasibility of using deep reinforcement learning to develop a collision avoidance strategy generalized for airplanes of most classes. In particular, we extend an existing deep reinforcement learning methodology used for unmanned aerial systems by implementing an evolving action space. We evaluate our strategyâĂŹs performance by comparing its risk ratio against that of TCAS II over a series of 15000 scripted aircraft encounters.

Approach

We model the problem as a POMDP, which comprises a state space S and and an action space.An agent in state $s \in S$ chooses an action $a \in A$ to receive a reward r and proceeds to state s' with probability $T(s'|a, s)$. The action taken is chosen on the basis of a policy π. An optimal policy maximizes the expected utility. Given that we cannot have complete knowledge of the function describing the transition between states, we estimate the optimal values for a state action pair, $Q(s, a)$, by using samples of (s, a, r, s') and a learning parameter, α. This yields the modified Bellman equation:

$$Q(s,a) = Q(s,a) + \alpha * [R + \gamma \max_{a' \in A} Q(s', a') - Q(s,a)]$$

We alter this formulation slightly by encapsulating the action space in a set $\mathring{A} \subseteq A$ such that for each aircraft equipped with our system, $\mathring{A}$'s interval evolves (on the basis of state information) to ultimately span the widest possible subset of A. This helps us customize resolution advisories to fully exploit individual aircraft capabilities

Model

Given state information and an imminent near mid-air collision (NMAC) between two aircraft, we use a 3-hidden-layer neural network with rectified linear unit activations and an experience replay [2] mechanism to take in 5 inputs containing state information and available actions and output an estimated Q value for each following each action given the state. Q learning updates are applied on batches of experience pairs obtained by sampling uniformly at random from a set of experience pairs (s, a, r, s') stored in memory.

This ensures that successive updates encompass the entire state space, thus improving the neural networks approximations.

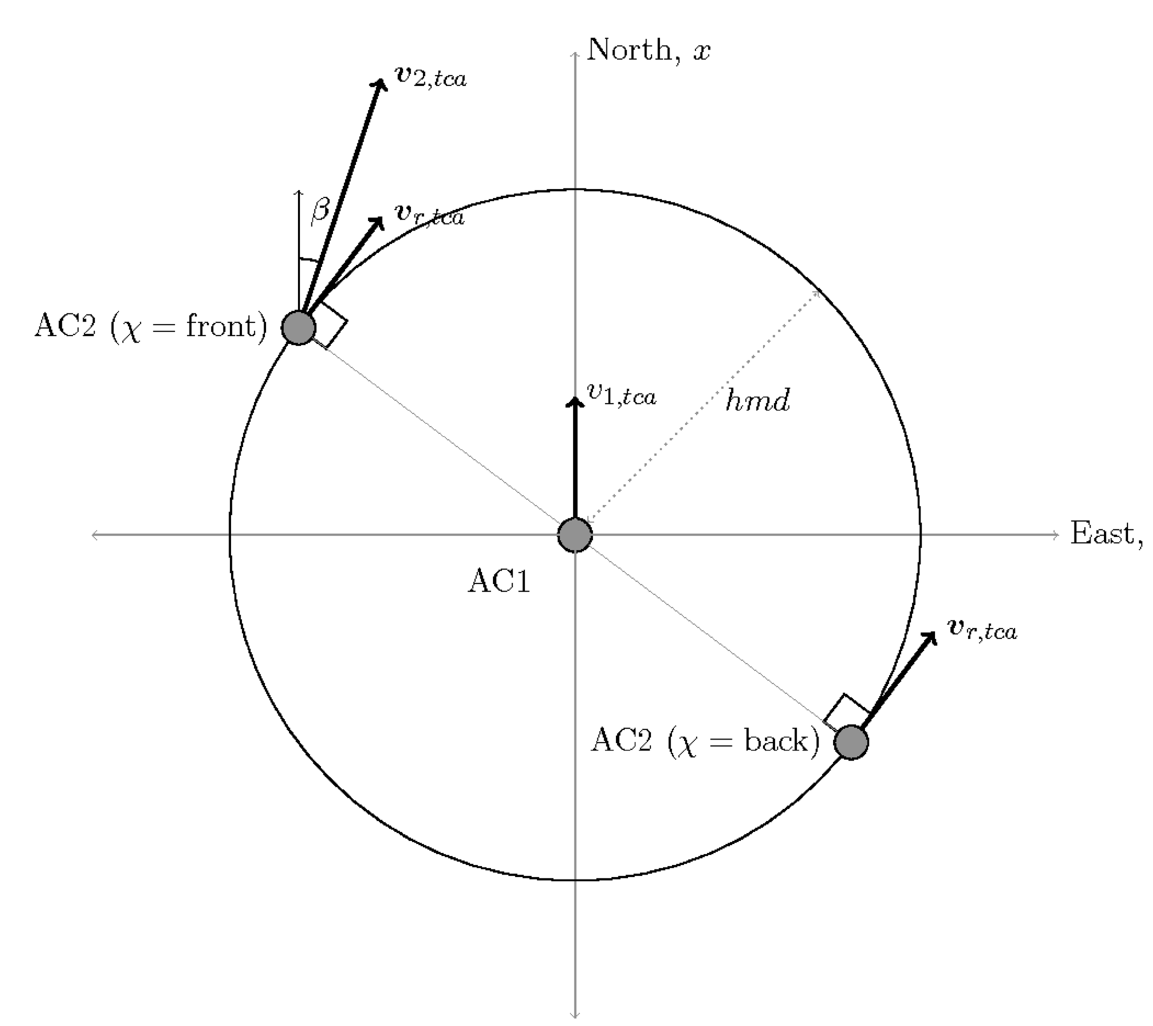

Figure 1: Encounter geometry of 2 airplanes at their closest point of approach

Results

CAS Algorithm	Risk Ratio
TCAS II	0.061230
POMDP based CAS	0.047315
DQN ACAS	0.002071

Table 1: Risk ratios calculated for 3 CAS models

Reward and Loss Function

Reward is thus calculated using the following metrics:

- $R_1 = -(1 + e^{(r_{sep} - r_{min})/C})^{-1}$ where hmd and vmd work together to describe r_{sep}, the distance of separation while r_{min} describes the minimum allowable separation (500 feet vertically and 200 feet horizontally) and C is the smoothing applied to the step function.
- $R_2 = -0.0002\theta_i^2$ where θ_i represents the bank angle in degrees. This penalizes unnecessary maneuvering.
- $R_3 = -0.04$ if $\theta_i \neq COC \vee h_i \neq COC$, penalizing false positives.
- $R_4 = -0.0001h_i$, penalizing unnecessary vertical movement.

The final reward is given by $R = R_1 + R_2 + R_3 + R_4$.

The loss function is given by:

$$L(\theta) = \left[\mathbb{E}(r + \gamma \max_{a' \in \mathring{A}} Q(s', a', \theta^-)) - Q(s, a, \theta)\right]^2$$

An explanation of the variables is available in our project report.

Conclusions

We successfully implemented a primitive airborne collision avoidance system that can easily be generalized to a wide variety of airplanes. It is our hope that with more time and computational resources, we will be able to refine our approach and offer an efficient aircraft-agnostic collision avoidance solution on an embedded device.