MODELING AND SIMULATION OF AUTONOMOUS UNMANNED AIRCRAFT TO ASSESS RISK OF COLLISION WITH OTHER AIRCRAFT

Thiago Toshio Matsumoto
Escola Politécnica da Universidade de São Paulo – EPUSP
Av. Prof. Luciano Gualberto – Trav. 3, 158, +55 11 3091-5395, ttmatsumoto@usp.br

João Batista Camargo Junior
Escola Politécnica da Universidade de São Paulo – EPUSP
Av. Prof. Luciano Gualberto – Trav. 3, 158, +55 11 3091-5401, joaocamargo@usp.br

ABSTRACT

One important principle of aviation that must be addressed for the operation of an unmanned aircraft is collision avoidance, a traditionally pilot-dependent functionality. In this regard, we present a simulation-based approach for implementing a learning-based autonomous control system for collision avoidance within an unmanned aircraft. Our approach is based on the concept of 'Learning from Demonstration', which is utilized to make the autonomous control system to imitate maneuvers defined by a human in aircraft conflict scenarios. Additionally, we present simulation results with discussion for development of future research in this subject.

Keywords: UAS, safety, detect and avoid, computational learning, simulation.

1. INTRODUCTION

Regarding collision avoidance, the International Civil Aviation Authority (ICAO) by means of its Rules of the Air (INTERNATIONAL CIVIL AVIATION ORGANIZATION, 2005) considers that "Nothing in these rules shall relieve the pilot-in-command of an aircraft from the responsibility of taking such action (…)". Therefore, the absence of a human pilot on board the aircraft may represent the loss of a protection layer regarding collision avoidance, especially when we consider for autonomous flights which can happen in situations of lost of datalink control for example. In this regard, the concept of a Piloting Autonomous System (PAS) was presented by GIMENES et al. (2013). Within an autonomous Unmanned Aircraft System (UAS), the PAS shall be responsible for all the activities traditionally executed by the human pilot on board the aircraft, which must include see and avoid. However, it is worth observing that in the Manual on Remotely Piloted Aircraft Systems (RPAS) (INTERNATIONAL CIVIL AVIATION ORGANIZATION, 2015), ICAO uses the concept of detect and avoid (DAA), which considers not only a *seeing* capability but also means to *sensing* or to *detecting* conflicting traffic or other hazards. As discussed by CONTARINO (2010), the sense capability of a UAS relies mainly on identification data exchanges among cooperative aircraft.

In order to compensate for the pilot removal from the aircraft in an autonomous UAS, MATSUMOTO et al. (2015) proposed the implementation of an autonomous control system capable of granting the aircraft a collision avoidance capability based on maneuvering demonstration provided by a human. The control system is responsible for sensing other aircraft and for avoiding a collision by maneuvering in such a manner as to mimic the human. Therefore, by keeping the same maneuvering capability when autonomously piloted, the UAS would not increase the Mid-Air Collision (MAC) risk, keeping the same safety level observed in manned aircraft. In this regard, the proposed control model would represent a partial implementation of the PAS by (2013), which must be capable of complying to the same quality indicators as a human pilot.

In this paper, we extend the work presented by MATSUMOTO et al. (2015) with results obtained by the implementation of the control model proposed. This paper is structured as follows: in section 2, we present related works with a brief comparison to the one presented herein. In section 3, we present important concepts from the literature used to support the proposed approach. In section 4, we present the defined methodology. In section 5, we present some preliminary results for the proposed approach. Finally, in section 6, we present our concluding remarks with a brief discussion of possible research gaps.

2. RELATED WORKS

The literature presents a number of different approaches to control an autonomous aircraft, including some learning-based ones such as (HERCUS; KONG; HO, 2013), (ROSS et al., 2013) and (WZOREK; KVARNSTR; DOHERTY, 2010). Additionally, it is possible to identify several implementations that are not based on computational learning, such as (LAI et al., 2011) and (BAI et al., 2012). However, it is worth observing that the majority of the aforementioned works focuses solely on the computational problem, with no consideration of ATC concepts.

Regarding the evaluation of collision avoidance strategies, the use of encounter models as presented by KOCHENDERFER (2010) represent a means to estimate collision risk that can be applied to any aircraft type (crewed or unmanned). Additionally, HOLT, BIAZ and AJI (2013) and ALEXOPOULOS et al. (2013) present comparisons of different methods for collision avoidance.

3. MAIN CONCEPTS

Following, we present some concepts necessary to support the implementation of a learning-based autonomous control system for an UAS.

3.1. Regulatory constraints

The regulatory framework for UAS presented by ICAO (2015) only considers the possibility of integration of RPAS into the international civil aviation system. However, as discussed by GIMENES et al. (2013), the lack of scientific research regarding the safety of autonomous UAS constitutes one of the main reasons for such a limitation. Furthermore, it is important to address issues inherent to the RPAS concept. One of those is the need of a wireless communication link between the ground station (where the pilot in command is located) and the aircraft. For a scenario of communication loss, ICAO considers that the preferred contingency action would be to land the unmanned aircraft at nearest appropriate designated landing site (INTERNATIONAL CIVIL AVIATION ORGANIZATION, 2015), which would have to be accomplished autonomously by the aircraft.

3.2. Learn from Demonstration

In order to grant flight autonomy to UAS, we propose the use of the concept of Learn from Demonstration (LfD). ARGALL et al. (2009) present a survey and categorization of the different techniques of LfD. Moreover, ARGALL et al. (2009) presents LfD as a subset of Supervised Learning, which consists in presenting a set of labeled training data to the agent, which will infer an approximation of the function that generated the training data. Specifically in LfD, the training data is obtained from example executions of the task performed by a demonstration teacher.

Based on the LfD concept, we use the trajectory of an aircraft defined by a human pilot (the demonstration teacher) as training data. Then, those data will be presented to the UAS control system so that it can learn an approximation of the function (human pilot control) that produced those data (supervised learning).

3.3. Support Vector Machine

Within machine learning, we assume a model defined up to a set of parameters (ALPAYDIN, 2010a):

$$y = g(x|\theta) \quad (1)$$

Considering a regression problem as assumed herein, in (1), y is a number obtained from the regression, x is the input and θ is the model parameters.

For model g(·), we decided to use the concept of Support Vector Machine (SVM) – a machine learning technique that applies optimization methods to calculate the regression coefficients. The preference for this method over others is based on the fact that, as SVM is formulated as a convex optimization problem, there is no need to consider learning rates selection, checking for convergence and the like (ALPAYDIN, 2010b).

One important characteristic of the SVM is the possibility to add a nonlinearity aspect to the model. As indicated by Smola & Scholkopf (2004), this may be achieved by preprocessing training patterns x_i by a (nonlinear) map $\Phi: X \rightarrow \mathcal{F}$ into some feature space $\mathcal{F}$. By doing so, it is possible to modify the model presented in (1):

$$y = g\left(\Phi(x)|\theta_{\Phi(x)}\right) \quad (2)$$

3.4. Evaluation metrics

In order to evaluate the effectiveness of the LfD-based control system for collision avoidance, we use the approach presented by KOCHENDERFER et al. (2010), which is based on the concept

of Near Mid-Air Collision (NMAC). A NMAC is defined as being a loss of separation 100 ft vertically and 500 ft horizontally. Furthermore, the probability of a NMAC to lead to a MAC is assumed as 0.1 (KOCHENDERFER et al., 2010). Therefore, it is possible to state that the NMAC represent a boundary that indicates whether two aircraft are passing too close (for safety reasons) or not.

For the proposed approach, we expect the control system to be capable of avoiding the occurrence of NMAC as much as a human pilot is. With the collision risk criterion defined based on NMAC, it is possible to execute a verification of the effectiveness of the proposed approach by making the LfD-based control system to execute a number of flights in a simulated airspace (Monte Carlo simulation) with virtual aircraft (after the training process).

4. METHODOLOGY

In this section, we define the scenarios of air traffic conflict to be considered. Additionally, we present the problem definition and the details of the learning-based control system model.

4.1. Autonomous UAS in the ATC system

In order to conceive models that are coherent to the air traffic system, we consider necessary to define air traffic conflict scenarios and collision avoidance maneuvers in accordance to the ICAO Rules of the Air (INTERNATIONAL CIVIL AVIATION ORGANIZATION, 2005). Those rules define maneuvers in three possible scenarios of conflict (approaching head-on, converging and overtaking) for an in-flight aircraft. Herein, we consider the scenario of two converging aircraft.

For the selected scenario, the ICAO's Rules of the Air defines that the aircraft that has the other on its right shall "give way" (i.e. avoid passing over, under or in front). However, in this work we consider that avoiding a collision will be always a responsibility of the autonomous UAS. In fact, we suppose a scenario in which the conflicting aircraft is not aware of the existence of the UAS. Due to validate the proposed autonomous control, the conflict aircraft do not change its own route. Those considerations have the following implications:

- The UAS will always give way to the conflicting aircraft, even if the former is on the right of the later.
- The trajectories defined by the encounter models of KOCHENDERFER et al. (2010) shall be sampled from the uncorrelated model, which is coherent to the fact of the crewed aircraft not being aware of the UAS.

4.2. Definition of a demonstration set

In order to select a set of demonstration maneuvers for the UAS in scenarios of conflict, we sample trajectories from the uncorrelated encounter model presented by KOCHENDERFER et al. (2010) for a crewed aircraft. According to the initial position of the aircraft in each sampled trajectory, we define an initial position for the UAS that characterize a conflict of converging aircraft. From that initial position, we define a demonstration trajectory for the UAS. It is worth observing that, as the UAS will have to pass behind the conflicting aircraft, we do not consider altitude changes. Therefore, we assume that during the conflict timespan both aircraft stay approximately in the same altitude.

Besides giving way to the conflicting aircraft, the definition of such demonstration trajectory must be in accordance to the following constraints:

1) UAS dynamic capabilities: the demonstration trajectory must be defined in accordance to the maneuvering limitation of the real UAS. Hence, we selected a Cirrus SR-22 aircraft as the reference airborne platform, as it has been used by NASA for unmanned operation (BARNSTORFF, 2012).

Herein, we adopted the premise of a constant speed for the UAS with turning maneuvers for collision avoidance with the standard turn rate of 3º/s (FEDERAL AVIATION ADMINISTRATION, 2008), as presented in Table 1.

Table 1: Dynamic parameters of the UAS reference aircraft.

Parameter	*Values*
Horizontal speed [kt]	100 (constant)
Turn rate [grad/s]	[-3, 3]

2) Separation distance: for each instant t of the conflict timespan, the position $s_{demo}(t)$ of the demonstration trajectory shall be sufficiently distant from the position $s_{aircraft}(t)$ of the conflicting aircraft to characterize a well clear passage. In the worst case, the horizontal distance $d_{sep}(t)$ must be greater than or equal to 500 ft to not representing an NMAC condition. Therefore, we define the demonstration trajectories by selecting values for $s_{demo}(t)$ that keep $d_{sep}(t) \approx d_{threshold} = 1\,NM,$ but with $d_{sep}(t) < d_{threshold}$, so as to keep the characterization of a conflict condition.

4.3. Unmanned Aircraft Model

In accordance to the formalization presented by ARGALL et al. (2009), the autonomous control system must be able to observe some states from the environment and maneuver the unmanned aircraft accordingly. Such capability would correspond to the see and avoid responsibility of the human pilot, which we replace by cooperative surveillance concepts herein. Such an approach is utilized in the technology known as Automatic Dependent Surveillance-Broadcast (ADS-B), in which each aircraft identify its own position using the Global Navigation Satellite System (GNSS) and broadcast that information to the others. Therefore, we define the sensing model of Figure 1 for the autonomous control system. The proposed model makes it possible to observe the following states:

- Movement speed and self-positioning: GNSS-based positioning sensor.
- Conflicting aircraft location and dynamics (e.g. horizontal speed): datalink interface.

It is worth observing that in the proposed approach we do not address some other essential functionalities for an aircraft, such as stabilization and guidance. Herein, we adopt the aircraft itself (or a subsystem within it) is responsible for implementing such functionalities. Moreover, given a desired value for a dynamic parameter (horizontal speed and turn rate), the aircraft shall be responsible to achieve such set point value.

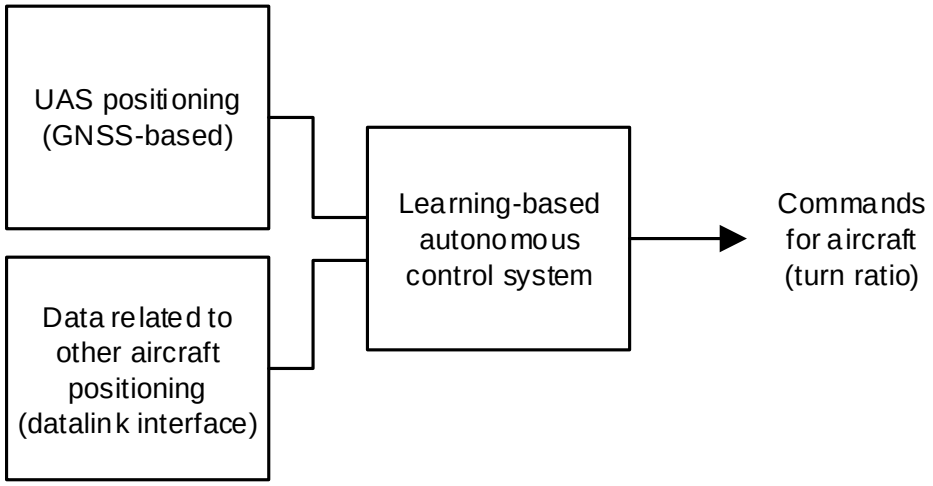


Figure 1: Block diagram of the UAS control system.

4.4. Learning-based autonomous control system model

According to the observed states (position and aircraft dynamics), the LfD-based control system shall take actions, which correspond to defining values of set point for turn rate of the aircraft. However, the direct use of position and movement data from each conflicting aircraft would constitute a problem with a variable number of inputs, depending on the number of nearby aircraft. Therefore, we propose a situational awareness model for the learning-based control system that is based on the division of the UAS perimeter in angular sectors of 36º (Figure 2) defined with respect to the direction the UAS is facing. By doing so, we define the sectors presented in Table 2.

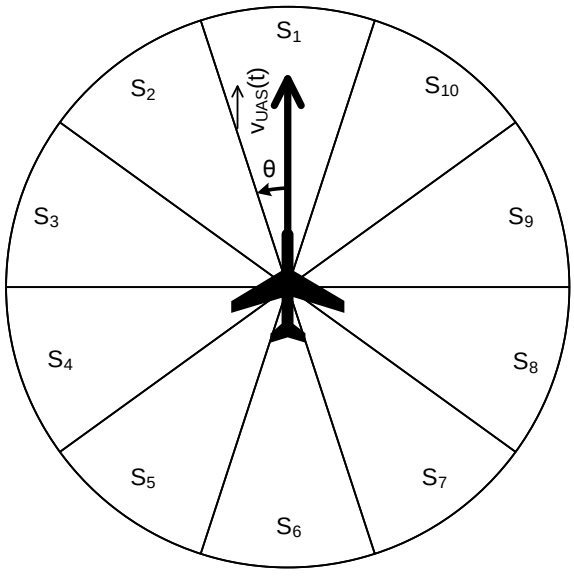


Figure 2: Control system situational awareness

Table 2: Angular sectors.

Sector	*Interval of θ [deg]*
S1	[-18, 18]
S2	(18, 54]
S3	(54, 90]
S4	(90, 126]
S5	(126, 162]
S6	(-162, 162)
S7	[-162, -126)
S8	[-126, -90)
S9	[-90, -54)
S10	[-54, -18)

For each sector S_i $(where\ i = 1..10)$ we associate a pair of binary inputs $(s_i(t), v_i(t))$, related to the position and speed of conflicting aircraft. The values of such inputs are defined as follows:

1) $s_i(t)$ – the value of $s_i(t)$ is defined as:

$$s_i(t) = \begin{cases} 1, if\ there\ is\ a\ conflicting\ aircraft \\ \left(d_{sep}(t) < d_{threshold}\right)\ in\ angular \\ \quad sector\ S_i\ in\ instant\ t \\ 0\ otherwise \end{cases} \quad (3)$$

2) $v_i(t)$ – given the speeds $\vec{v}_{UAS}(t)$ of the unmanned aircraft and $\vec{v}_{conflict}(t)$ of any conflicting aircraft in the instant t ($d_{sep}(t) < d_{threshold}$):

$$v_i(t) = \begin{cases} 1, if\ \arg\left(\vec{v}_{UAS}(t) + \vec{v}_{conflict}(t)\right) \\ \quad is\ in\ angular\ sector\ S_i \\ 0\ otherwise \end{cases} \quad (4)$$

Functions $s_i(t)$ and $v_i(t)$ are responsible for mapping the input patterns (position and speed of aircraft) into a feature space ($s_i(t)$ and $v_i(t)$ are responsible for mapping Φ as described in section 3.3), originating vectors in the form $(s_1(t), \cdots, s_{10}(t), v_1(t), \cdots, v_{10}(t))$. The standard SVM regression algorithm can then be applied in order to generate turn rates for the aircraft maneuvers.

4.5. Control system training and collision risk estimation

For the UAS control, the aforementioned regression shall be executed by the autonomous control system in a regular basis with a period T_{reg}. The selection of this value depends on the cooperative surveillance technology used within the UAS. As an example, according to the transmission rates defined for ADS-B in DO-260B (RTCA INC., 2009), the autonomous control system would receive conflicting aircraft updates twice per second. Hence, it is reasonable not to make a maneuver decision (regression of the turn ratio) within a shorter interval. Herein, we have defined T_{reg} as 1 s.

Therefore, to the proposed control system model, a demonstration trajectory originates N ordered pairs $X = \{x^t, r^t\}_{t=1}^{N}$, where N represents the aircraft conflict duration (discretized with period T_{reg}). Within a training ordered pair, x^t is the t-th training input (in this work, $s_i(t)$ and $v_i(t)$, $i = 1..10$) and r^t is the associate desired output (turn ratio). For each pair, the value of r^t is defined according to the yaw value of the aircraft at the instant $t + 1$: $r^t = [\text{turn_angle}_{UAS}(t+1) - \text{turn_angle}_{UAS}(t)]/T_{reg}$.

From the encounter models, we sample N_{train} trajectories for training the model and N_{test} trajectories for collision risk estimation (according to the NMAC criterion).

4.6. Real-time simulation with a flight simulator

For verification of the maneuver capabilities of the unmanned UAS according to the autonomous control system decision, we use the trained algorithm to send the generated commands (turn rate) to a virtual implementation of the unmanned aircraft within a flight simulator as presented in Figure 3. Such an approach is based on the concepts presented by GIL et al. (2010) for the "Integrated Platform for Testing Critical Embedded Systems" (PIpE-SEC) and makes it possible to identify possible limitations of the control model, especially, regarding an UAS dynamic capabilities.

For such a verification, we selected the flight simulator X-Plane 10 (LAMINAR RESEARCH, 2011), which is certified by FAA for pilot training purposes.

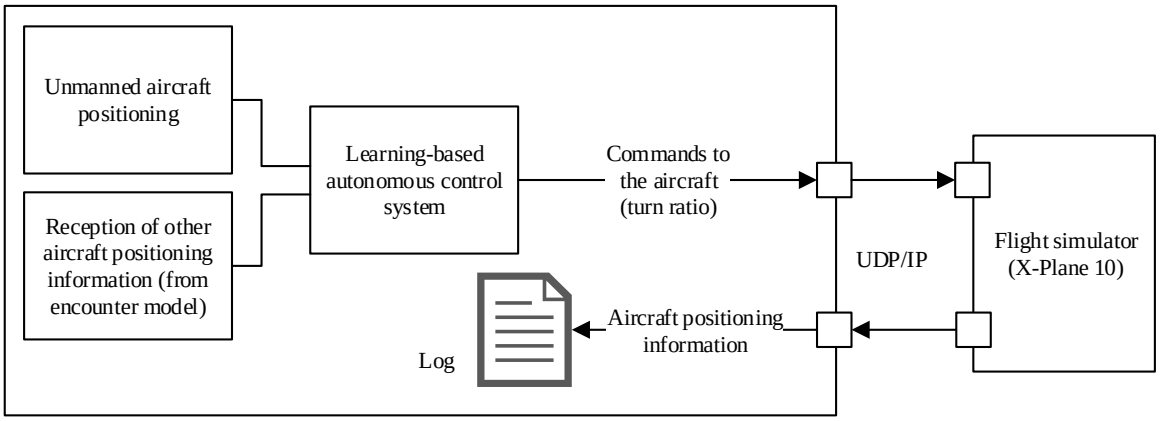


Figure 3: Use of autonomous control system commands to control a virtual aircraft.

5. RESULTS

For training the control algorithm, firstly we sampled 100 (N_{train} = 100) trajectories from the encounter model. Then, for each of those trajectories, we defined 10 possible initial positions for the unmanned aircraft that would represent an imminent conflict condition, resulting in 1000 conflict scenario as demonstration data for the control algorithm. However, it is worth observing that not all those scenarios were NMAC-free, as we defined about 140 demonstration trajectories that lead to a NMAC.

As a preliminary evaluation of the trained algorithm, we simulated the control algorithm with the same trajectories and initial positions used for training purposes. The trained algorithm was capable of avoiding the occurrence of NMAC in 722 conflict scenarios, which indicates a collision rate twice the one observed in the training data. Besides, by comparing the demonstration trajectories and the algorithm-defined ones, we identified the following types of results:

- Both trajectories were NMAC-free (example in Figure 4).
- Demonstration trajectory was NMAC-free; algorithm-defined one was not (Figure 5).
- Demonstration trajectory was not NMAC-free; algorithm trajectory was (Figure 6).
- Both trajectories were not NMAC-free (Figure 7).

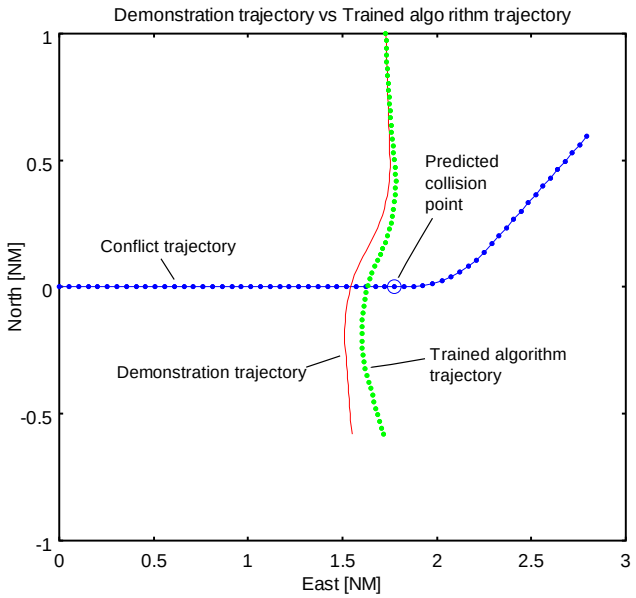


Figure 4: Example of NMAC-free trajectories.

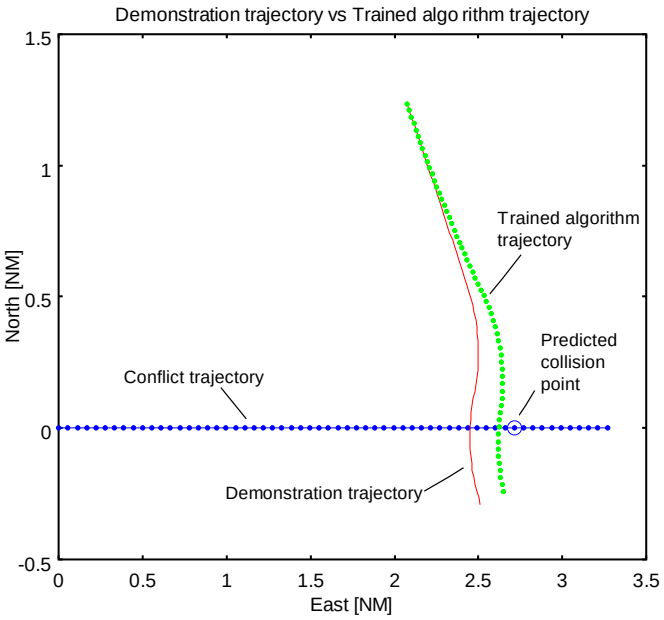


Figure 5: Demonstration trajectory is NMAC-free; algorithm trajectory is not.

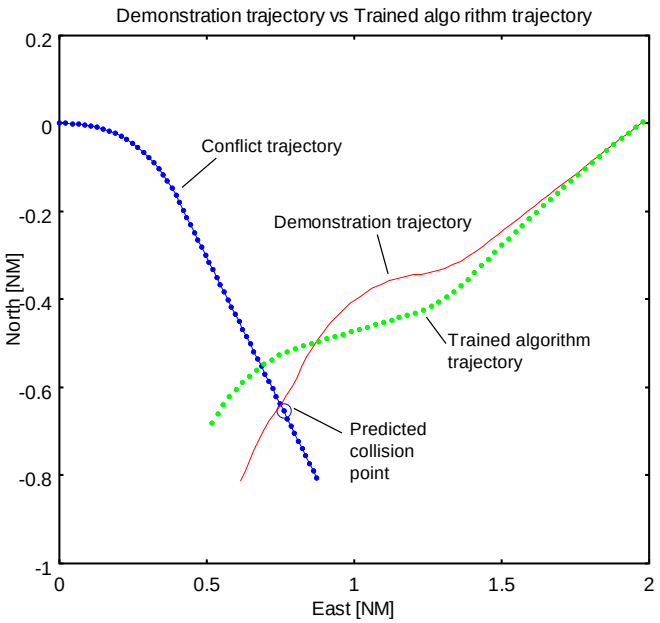


Figure 6: Demonstration trajectory is not NMAC-free; algorithm trajectory is.

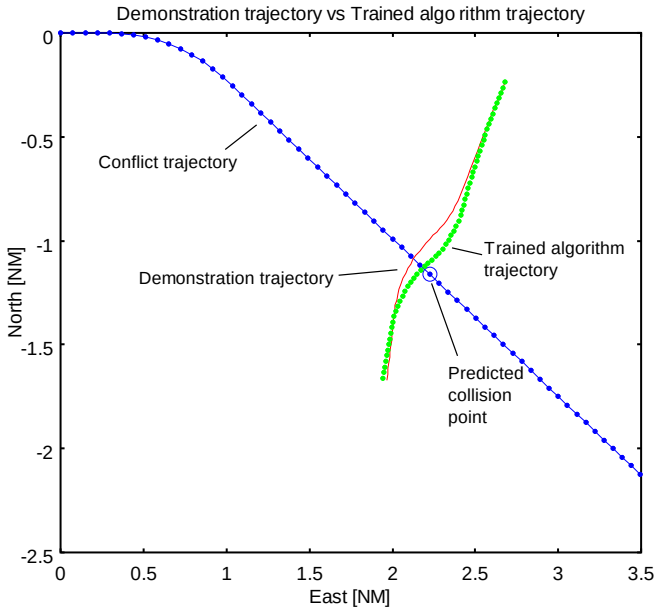


Figure 7: Trajectories that lead to the occurrence of NMAC.

For estimating collision risk associated to the trained control model, we sampled other 200000 trajectories (N_{test} = 200000) and, for each of them, we defined one initial position for the unmanned aircraft to characterize a conflict scenario. By simulating the control algorithm in those scenarios, we obtained 55523 NMAC occurrences, which represents 27.76% of the test scenarios generated for testing. As the probability of a NMAC to lead to a MAC is assumed as 0.1 (section 3.4), the results indicates a probability of about 2.8% of collision.

Additionally, we modified some of the simulating parameters in order to identify possible improvements for the model (regarding collision rate). The most relevant difference was obtained by increasing the separation distance $d_{threshold}$. Such an increase allows the autonomous control system to identify the conflict condition earlier, with additional space for maneuvering. Therefore,

by modifying such distance from 1 NM to 1.3 NM, in the simulation executed within the same 200000 scenarios, we obtained 16252 NMAC occurrences (collision rate of approximately 0.8%).

Finally, we used the commands generated by the autonomous control system to maneuver a virtual aircraft within X-Plane 10. By doing that, we observed that the virtual aircraft describes a trajectory that loosely approximates the trajectories that were computationally calculated within the accelerated-time simulation. However, the virtual aircraft performs less sharp curves due to the delays inherent to the aircraft flight systems and to the aircraft itself. In this case, even with maneuver decisions being correctly performed by the control algorithm, the resulting trajectory may not be sufficiently distant from the conflict path. Still, it is possible to observe that it is possible to compensate for these delays by increasing the distance $d_{threshold}$, since the decisions of maneuver are performed by the algorithm in order to maintain a larger separation distance. One example of trajectory obtained by integrating the autonomous control system to X-Plane 10 is presented in Figure 8.

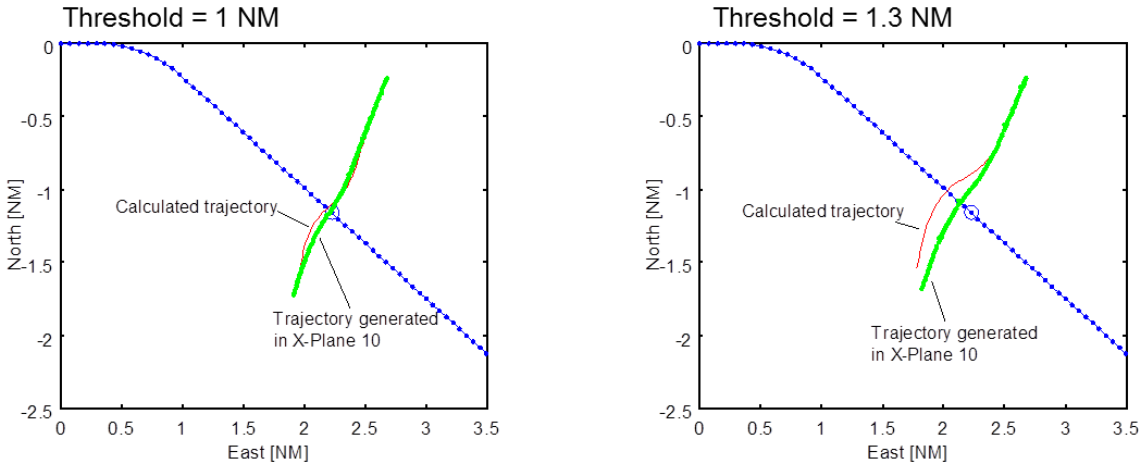


Figure 8: Comparison of trajectories calculated computationally and trajectories defined by virtual aircraft.

6. CONCLUDING REMARKS

We presented a learning-based implementation of an autonomous control system for a UAS focused on collision avoidance. The results obtained by such implementation indicates that the control algorithm was capable of avoiding the occurrence of NMAC for most of the conflict scenarios. However, as demonstrated by the calculated NMAC rates, the autonomous control still does not present the same performance as the demonstration teacher. Therefore, the next step for this work consist of a fine-tuning of the current implementation. As the results suggest, the more promising strategy would be to increase to separation distance $d_{threshold}$, in order to allow the control system to identify the conflict and react accordingly earlier.

Besides, it is worth observing that the use of aircraft trajectory data collected during the training of human pilots would represent a more realistic approach for defining the demonstration data for the algorithm. We expect to be able to use such kind of data in the future by means of cooperation agreements with the local aviation authority.

Finally, it is worth observing that some assumptions adopted in this work may imply in limitations of the model. Some of those limitations represent research gaps that must be addressed for a real autonomous UAS implementation:

1) Conflict scenarios: it is necessary to analyze other scenarios of aircraft conflict, such as approaching head-on and overtaking aircraft. The analysis of such scenarios is especially important for lightweight unmanned aircraft, given the differences in thrust and speed capacity.

2) Flight plan: the presented approach only considered the maneuvers of the UAS during the timespan of the aircraft conflict. However, after avoiding a collision, the UAS should be able to return to its original path. A possible solution would be the inclusion of a flight plan as an input for the autonomous control decision.
3) Safe separation distance: in order to consider the effects of the wake turbulence caused by the conflicting aircraft, it would be necessary to turn $d_{threshold}$ into a function of the conflicting aircraft category (identified according to cooperative surveillance data). Another possible improvement would be the inclusion of a time-to-go parameter similar to range tau used in TCAS.
4) Controlled motion variables: besides the turn ratio, a human pilot is able to control other motion parameters of an aircraft, such as pitch and thrust. For each of those parameters, it would be necessary to associate a model $g(\cdot)$ to calculate the regression of the respective value according to the model inputs.
5) PAS responsibilities: although collision avoidance is an important pilot activity, a system intended to compensate for the absence of the human pilot would have to assume several other responsibilities within an autonomous UAS as defined by GIMENES et al. (2013) for the PAS. Tasks such as communicating with an air traffic controller, navigating according to a flight plan and taking decisions in fault scenarios should be incorporated to the control model presented herein.
6) Control method: as presented in section 2, in the literature, it is possible to identify several other approaches utilized to control a UAS autonomously. The comparison of those methods to the one utilized herein (SVM) regarding collision risk represent another subject for research.
7) Non-cooperative obstacles: a real UAS application may have to consider non-cooperative conflicts as well (e.g. non-communicating aircraft, balloons and the like).

The aforementioned limitations indicate that, for the operation of an autonomous UAS, the presented learning-based autonomous control system approach must be expanded. In order to create a model capable of reproducing the necessary abilities to fly an unmanned aircraft without a human in command, it is still necessary to consider a number of other conditions. However, our preliminary results indicate the Learn from Demonstration approach as a valid possible strategy to improve the collision avoidance capability of an unmanned aircraft.

7. REFERENCES

ALEXOPOULOS, A. et al. “A Comparative Study of Collision Avoidance Techniques for Unmanned Aerial Vehicles”. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics. Anais...IEEE, oct. 2013

ALPAYDIN, E. “Introduction”. In: DIETTERICH, T. (Ed.). Introduction to Machine Learning. 2nd. ed. [s.l.] The MIT Press, 2010a. p. 1–21.

ALPAYDIN, E. “Kernel Machines”. In: DIETTERICH, T. (Ed.). Introduction to Machine Learning. 2nd. ed. [s.l.] The MIT Press, 2010b. p. 309–340.

ARGALL, B. D. et al. “A survey of robot learning from demonstration”. Robotics and Autonomous Systems, v. 57, n. 5, p. 469–483, may 2009.

BAI, H. et al. “Unmanned Aircraft Collision Avoidance using Continuous-State POMDPs”. Robotics: Science and Systems VII, v. 1, p. 1–8, 2012.

BARNSTORFF, K. “‘Look Ma, No Hands!’ Flight Tests Completed”. Disponível em: <http://www.nasa.gov/centers/langley/aeronautics/ldcap.html>. Acesso em: 18 feb. 2015.

CONTARINO, M. “All weather collision avoidance for unmanned aircraft systems”. (G. R. Gerhart, D. W. Gage, C. M. Shoemaker, Eds.) In: Unmanned Systems Technology XII. Anais...23 apr. 2010

FEDERAL AVIATION ADMINISTRATION. “Pilot’s Handbook of Aeronautical Knowledge”. [s.l: s.n.].

GIL, F. DE O. et al. “PIpE-SEC: platform for tests and validation of unmanned aerial vehicles (UAVs) operation in controlled airspace”. Journal of the Brazilian Air Transportation Research Society, v. 6, n. 2, p. 47–58, 2010.

GIMENES, R. A. V. et al. “Guidelines for the Integration of Autonomous UAS into the Global ATM”. Journal of Intelligent & Robotic Systems, v. 74, n. 1–2, p. 465–478, 27 sep. 2013.

HERCUS, R.; KONG, H.-S.; HO, K.-F. “Control of an unmanned aerial vehicle using a neuronal network”. In: 2013 IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB). Anais...IEEE, apr. 2013

HOLT, J.; BIAZ, S.; AJI, C. A. “Comparison of Unmanned Aerial System Collision Avoidance Algorithms in a Simulated Environment”. Journal of Guidance, Control, and Dynamics, v. 36, n. 3, p. 881–883, may 2013.
INTERNATIONAL CIVIL AVIATION ORGANIZATION. “Annex 2 to the Convention on International Civil Aviation - Rules of the Air”, 2005.
INTERNATIONAL CIVIL AVIATION ORGANIZATION. “Doc 10019 Manual on Remotely Piloted Aircraft Systems (RPAS)”, 2015.
KOCHENDERFER, M. J. et al. “Airspace Encounter Models for Estimating Collision Risk”. Journal of Guidance, Control, and Dynamics, v. 33, n. 2, p. 487–499, mar. 2010.
LAI, C.-K. et al. “On-board trajectory generation for collision avoidance in unmanned aerial vehicles”. In: 2011 Aerospace Conference. Anais...IEEE, mar. 2011
LAMINAR RESEARCH. “X-Plane 10”. Disponível em: <http://www.x-plane.com/desktop/home/>. Acesso em: 20 feb. 2015.
MATSUMOTO, T. T. et al. “A Learning-Based Autonomous Control System Approach for Collision Avoidance within an Unmanned Aircraft”. In: 2015 IEEE International Conference on Dependable Systems and Networks Workshops. Anais...IEEE, jun. 2015
ROSS, S. et al. “Learning monocular reactive UAV control in cluttered natural environments”. 2013 IEEE International Conference on Robotics and Automation, p. 1765–1772, may 2013.
RTCA INC. “DO-260B - Minimum Operational Performance Standards for 1090 MHz Extended Squitter Automatic Dependent Surveillance - Broadcast (ADS-B) and Traffic Information Services - Broadcast (TIS-B)”, 2009.
SMOLA, A. J.; SCHOLKOPF, B. “A tutorial on support vector regression”. Statistics and Computing, v. 14, n. 3, p. 199–222, 2004.
WZOREK, M.; KVARNSTR, J.; DOHERTY, P. “Choosing Path Replanning Strategies for Unmanned Aircraft Systems”. In: Proceedings of the Twentieth International Conference on Automated Planning and Scheduling (ICAPS 2010). Anais...2010