Stochastic Dynamic Agent-based Modelling for Evaluation of ACAS and DAA Systems

Sybert Stroeve
Safety & Human Performance
Royal Netherlands Aerospace Centre NLR
Amsterdam, The Netherlands
sybert.stroeve@nlr.nl; https://orcid.org/0000-0002-5294-2601

Abstract*—Validation studies for airborne collision avoidance systems (ACAS) and detect-and-avoid (DAA) systems have strongly relied on fast-time simulation of large sets of encounters, while the impact of sensor errors and (remote) pilot performance variability have been considered to a limited extent only. This paper shows the need of ACAS/DAA validation approaches for feedback to design and approval that account for extended variability in sensor input and human performance. Stochastic dynamic agent-based models are presented that describe the variability of interacting agents in encounter-scenarios. Monte Carlo (MC) simulation results are provided for encounters of manned aircraft equipped with TCAS II or ACAS Xa and for unmanned aircraft equipped with ACAS Xu. The results show that the nonlinear dynamics and stochastic influences in the sociotechnical ACAS/DAA systems can critically affect the aircraft manoeuvres, and the safety and efficiency of the operations. Pilot performance has the largest impact on the overall performance, which is much larger than the impact due to differences between TCAS II and ACAS Xa. Conventional estimates of near mid-air collision (NMAC) probabilities are often lower than the estimates achieved using MC simulation of agent-based models with sensor errors. In remotely piloted aircraft systems (RPAS), strictly following the remain-well-clear guidance of ACAS Xu can lead to livelock conditions, where the RPAS cannot effectively pass each other. It is concluded that stochastic dynamic models describing all elements of the sociotechnical systems encompassing the ACAS/DAA system are essential for reliable validation. This is especially important in the design of artificial intelligence-based systems like ACAS Xa/Xu, where there exists a close connection between evaluation of the performance of the overall system and tuning of meta-parameters of the optimization process.

Keywords—Airborne Collision Avoidance System, Detect And Avoid, Validation, Agent-based Modelling, Safety, Human Factors, Artificial Intelligence

I. INTRODUCTION

Airborne collision avoidance systems (ACAS) and detect-and-avoid (DAA) systems are important means for (remote) pilots of manned or unmanned aircraft to avoid near mid-air collision (NMAC) or loss of DAA well clear (LDWC). An ACAS provides a cockpit crew with traffic advisories (TAs) regarding nearby aircraft that pose a threat, and resolution advisories (RAs) for vertical rates that are needed to avoid a potential collision within the last minute before the closest point of approach (CPA) with a threat aircraft. TCAS II version 7.1 [1] is the ACAS implementation that is used in current commercial air transport operations. Developed since the 1980s, the system uses slant range, altitude and bearing estimated by interrogation of secondary surveillance radar transponders of nearby aircraft and a set of rules to specify its advisories. ACAS Xa is a more recently developed ACAS implementation [2, 3]. While largely using the same types of surveillance data and advisories, the key difference with TCAS II is that it employs a dynamic programming (DP) optimization approach, implemented in a look-up table for its advisory specification.

A DAA system supports a remote pilot (RP) of an unmanned aircraft system (UAS) to observe and avoid nearby air traffic [4]. It can have a collision avoidance (CA) as well as a remain well clear (RWC) functionality. The CA function provides last-resort RAs to the RP to avoid a collision. Differently than ACAS RAs, a DAA can specify both vertical and horizontal RAs. The RWC function provides horizontal and/or vertical flight path guidance to a RP to prevent a conflict developing into a collision hazard. DAIDALUS (Detect and Avoid Alerting Logic for Unmanned Systems) is a DAA reference system of [4]. It employs rule-based logic, inspired by TCAS II, to determine the level of threat of a well-clear volume and to compute horizontal and vertical guidance to remain or regain well-clear [5]. ACAS Xu is a DAA system incorporating the RWC and CA functions, which employs a DP optimization-based advisory logic, similar to ACAS Xa, but for both horizontal and vertical manoeuvring [6].

The DP optimization employed in ACAS Xa and Xu uses Markov decision process (MDP) models, state-space discretisation and reward functions that specify the desired performance. DP is part of the family of reinforcement learning methods, which are machine learning methods that learn what to do so as to maximize a reward signal [7]. Given the broad definition of artificial intelligence (AI) by EASA [8] as "technology that can, for a given set of human-defined objectives, generate outputs such as content, predictions, recommendations, or decisions influencing the environments they interact with" also rule-based ACAS and DAA systems can be seen as AI. Indeed from the perspective of a (remote) pilot there is little difference between an advisory that is based on a (complex) set of rules or an optimization-based system. So both rule-based and optimization-based ACAS and DAA systems can be regarded as AI-based systems. In a human-centric framework for AI in aviation [8] this implies due consideration of human agency and oversight for system certification. Principally, effectiveness and safety should be evaluated for the overall sociotechnical system encompassing AI-based system(s), other technical systems, human operators and the environment in the operational context.

Validation studies for ACAS and DAA have typically used limited sets of human-in-the-loop (HITL) simulations and large sets of fast-time simulations of encounters with aircraft having mixes of equipage. The focus in such fast-time simulations has primarily been on evaluating large sets of encounter configurations, while the impact of sensor errors and pilot performance variability have been considered to a limited extent only. The objective of this paper is to show the need and illustrate the added value of validation approaches that more strongly consider the overall ACAS/DAA sociotechnical systems. This is supported by the development of agent-based models, which explicitly account for

variability in sensor errors and pilot performance, and Monte Carlo (MC) simulation of encounter-scenarios.

Next, Section II describes approaches that have been applied in the development and validation of ACAS and DAA systems. Section III provides an overview of the stochastic dynamic agent-based models developed to support extended validation. Section IV shows a number of illustrative results of MC simulation of the agent-based models; these results are partly based on simulation statistics gathered in [9]. Section V gives a discussion of the findings. Section VI provides the main conclusions.

II. Development and Validation of ACAS and DAA

A. Elements of ACAS and DAA systems

In TCAS II, mode C and mode S secondary surveillance radar transponders of nearby aircraft are interrogated. Based on the replies received, the system tracks the slant range, altitude, and bearing of surrounding traffic. Using this information and a set of fixed rules for alert generation, TCAS II provides its vertical rate advisories. Principally, the TCAS II logic estimates the time to CPA between the ownship and an intruder, based on the measurements of the range and the closure rate in the encounter. It uses an altitude-dependent sensitivity level, which determines threshold values for this estimated time, ranging from 15 s at low altitudes to 35 s at high altitudes, as well as thresholds for the expected horizontal and vertical distances at CPA. Volume II of the TCAS II minimum operational performance standards (MOPS) [10] uses state-chart diagrams to define in detail (over 744 pages) the states, transitions and interactions of all elements of its surveillance filters and threat resolution logic, including their parameter values. The MOPS documents [1, 10] were used by avionics manufacturers as a basis for their system design and software development of TCAS II implementations.

ACAS Xa is a newly developed ACAS [3], which is intended to be a successor of TCAS II with aimed lower collision risk and fewer nuisance alerts. ACAS Xa includes a Surveillance and Tracking Module (STM), which uses enhanced tracking algorithms that correlate surveillance data from multiple sensors, including the transponder-based data also used by TCAS II, but also allowing to use Global Navigation Satellite Systems (GNSS)-based ownship position estimates and Automatic Dependent Surveillance-Broadcast (ADS-B) data. The largest innovation in ACAS Xa is found in its Threat Resolution Module (TRM), which uses optimized decision logic stored in look-up tables to determine its advisories given probabilistic estimates of ownship and intruder states from the STM. The optimized decision logic tables have been determined off-line using a partially observable Markov decision process model (POMDP) and dynamic programming [2, 11]. This optimization process uses (1) probabilistic dynamic models for pilot response and aircraft movements, (2) discretization of the state space, including relative altitude, vertical rates, and time to collision, (3) a reward function with an extensive set of cost components for close proximities and types of advisories, and (4) dynamic programming to maximize the reward function, leading to state-action values that are stored in large look-up tables. For the provision of advisories ACAS Xa extends these precomputed actions with coordination rules for complementary advisories, and with online costs for required system performance, e.g., low altitude inhibitions of descend RAs and RA transition penalties. Volume II of the ACAS Xa MOPS [3] provides the algorithm design description (ADD), which defines (over 489 pages) a total of 362 algorithms of the STM and TRM in the Julia programming language. The ADD thus serves as a basis for the software development by avionics manufacturers. The ACAS Xa MOPS do not describe the POMDP and reward function components that were used in the DP for the generation of the lookup tables. Parameter data item files (PDIFs) for the lookup tables (80 MB in size) are solely distributed by RTCA.

The ACAS Xu DAA system for remotely piloted aircraft systems (RPAS) uses largely the same principles and system architecture as ACAS Xa. In addition to active Mode S/C interrogation and ADS-B, its STM also allows the use of an on-board air-to-air radar (ATAR) for independent surveillance. The TRM of ACAS Xu is based on two independent POMDP models for advisories in the vertical plane and in the horizontal plane. The optimization for the vertical and horizontal dimensions is separated, since the combined problem was considered intractable to solve due to its large (discretized) state space [12]. This implies that no overall 3D optimized advice is achieved. The POMDPs and associated reward functions are solved by DP, resulting in lookup tables that are combined with online costs to specify vertical and horizontal RAs. According to [12], the RWC guidance provided by ACAS Xu is based on a rollout approach [13], which uses the CA POMDP-based cost tables to infer an increase in collision risk in relation to DAA alert timing requirements stemming from RTCA DAA MOPS [4]. In general, a rollout algorithm is a DP-related approach that starts from some given heuristic and constructs another heuristic with better performance than the original [13]. Unlike DP, it is not guaranteed to find the optimal solution of an optimization problem. The ADD of ACAS Xu [14] (764 pages) defines 582 algorithms in Julia. As for ACAS Xa, the ACAS Xu MOPS do not define the POMDP, reward function or rollout approach. The PDIFs (5 GB in size) defining the performance are distributed by RTCA.

In summary, TCAS II, ACAS Xa, and ACAS Xu all are complex systems composed of many interacting algorithms and associated parameters. TCAS II is completely rule-based with performance being fully determined by rule-associated parameters defined in its MOPS. In addition to a broad set of algorithms as defined in their MOPS, the performance of ACAS Xa and Xu is based on large sets of optimization-based parameters, where the optimization details have been excluded from the MOPS.

B. Validation and feedback to design

Validation and feedback to design was important for the development of TCAS II and it is even more important for the design of optimization-based systems like ACAS Xa and Xu. Such validation has included the use of fast-time simulation and HITL simulation of encounters between aircraft with various types of ACAS/DAA equipment (see Fig. 1).

In fast-time simulation, especially the evaluation of the impact of large sets of encounters on relevant metrics provides important feedback to the effectiveness and operational acceptability of ACAS/DAA systems. Key safety metrics are the probability of an NMAC, i.e. the vertical miss distance (VMD) is less than or equal to 100 feet and the horizontal miss distance (HMD) is less than or equal to 500 feet, the risk ratio, i.e. the NMAC probability with ACAS relative to the NMAC probability without ACAS, and for DAA, the probability of LDWC as defined in [4]. Operational acceptability metrics

include the rates of RAs and their types, and additional distance flown.

As a basis for such fast-time simulations, considerable emphasis has been placed on the encounter models that specify the probabilities and characteristics of ways that aircraft can come close to each other. These characteristics include the VMD, HMD, relative heading and bearing when the aircraft are at CPA, and the altitude, airspeed, vertical rate, turn rate, and acceleration during the encounter. For ACAS safety studies, a range of encounter models [15, 16] were represented by Bayesian networks, for which the parameter values were tuned using large sets of radar data of manned aircraft. Lacking sufficient radar observations for UAS encounters, Bayesian models for encounters of manned aircraft have also been used for the development of ACAS Xu [12].

In addition to ways that aircraft come close in an encounter, models for aircraft performance, associated avionics, and the performance of (remote) pilots have impact on the trajectories and the performance metrics. For instance, for evaluation of NMAC probabilities in ACAS encounters, the impact of pressure altitude errors has been calculated by accounting for an altimetry error probability distribution in a postprocessing step [17, 18]. For pilot response in ACAS encounters, typically the ICAO standard pilot response model (§4.4.2.5 of [19]) is used as a basis. This model assumes that pilots respond with a delay of 5 s to an initial RA and a delay of 2.5 s to subsequent RAs; if a change in vertical rate is required, the acceleration towards the required vertical rate is 0.25 g or 0.35 g. Additionally it may be assumed that pilots do not always respond to an RA. Also more sophisticated pilot models may be applied.

The performance metrics of fast-time simulations provide important feedback to designers of ACAS/DAA systems. For a rule-based system like TCAS II, designers may adapt algorithms or parameter values of the surveillance filtering and tracking, or of the logic to provide advisories or guidance. For an optimization-based system like ACAS Xa, designers can similarly adapt algorithms or parameters of the STM or TRM, and also they can tune the POMDP models and especially the reward function to achieve the optimized look-up tables [11, 20]. There can be multiple ways to adapt an ACAS/DAA system and they may have opposing effects on desired performance characteristics (like NMAC probability versus RA rate). The choices made require considerable expertise on the ACAS/DAA system elements and the optimization approach. Designers use several iteration cycles where they adapt the design, evaluate the impact in sets of simulations, and learn the sensitivities of design choices. As is visualized in Fig. 1, the evaluated ACAS/DAA performance depends on the combination of the ACAS/DAA system, on the models applied in the fast-time simulation, and on the performance metrics processing. This implies that the design is tuned and may be biased by the assumptions used in these processes and models. Such dependency of the performance of an ACAS/DAA design on the fast-time simulation elements implies that it is important to apply sufficiently independent approaches in the validation for approval of an ACAS/DAA system by certifying authorities following the design cycles.

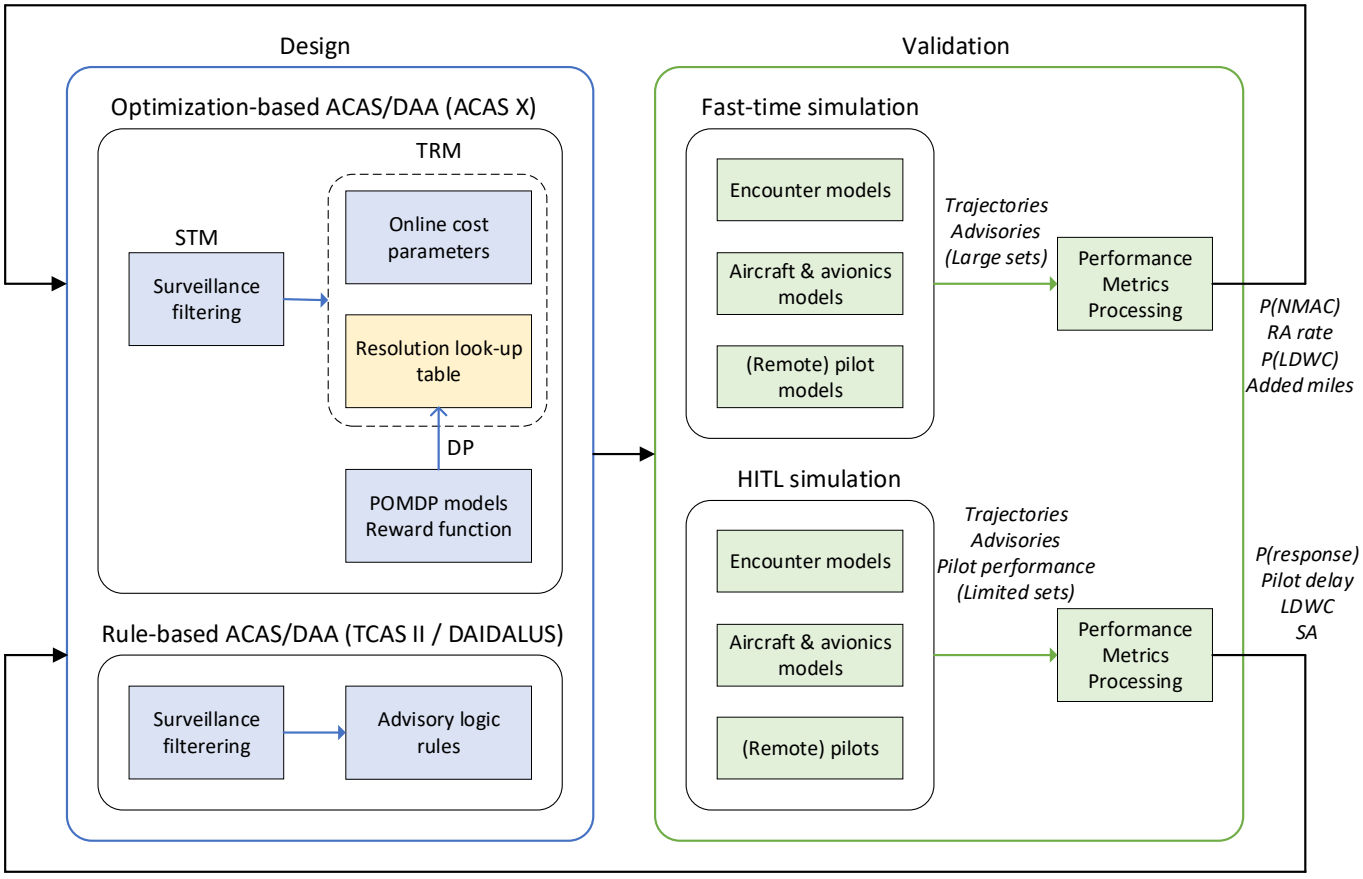

Fig. 1. Role of fast-time and HITL simulation in validation and feedback to design of ACAS/DAA systems

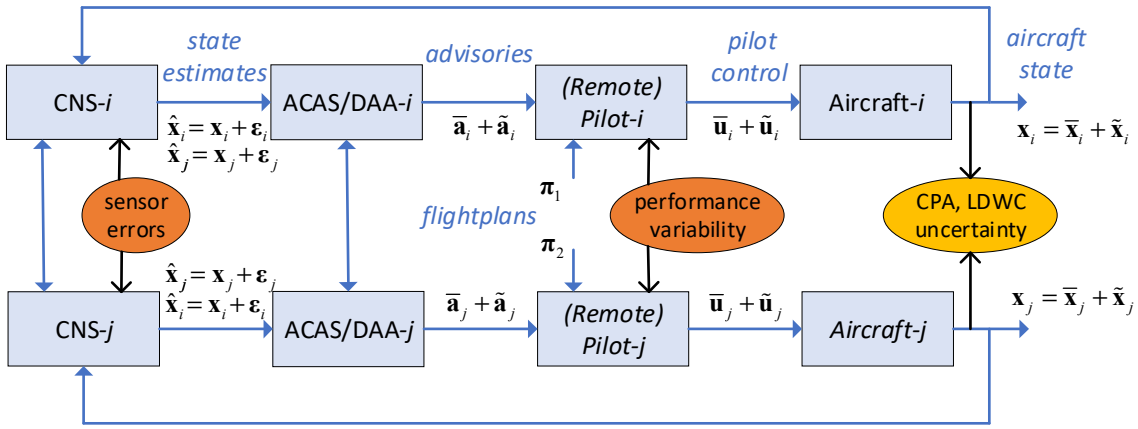

Fig. 2. Interacting uncertainties in ACAS/DAA encounter-scenarios

Given the key role of the (remote) pilot in the ACAS/DAA sociotechnical system, also HITL simulation provides important feedback for design (especially of more mature systems) and for final approval. In particular for DAA systems, pilot response is complex, since the RP needs to decide how to respond to guidance and advisories at a longer timescale and in multiple dimensions, possibly in coordination with air traffic control (ATC). As in fast-time simulation, HITL simulations apply particular encounter models and models for aircraft performance and avionics. Clearly, the number of encounters that can be evaluated is much more limited and the ACAS/DAA HMIs also play a key role. Furthermore, the number of RPs, their experience, the instructions they received for the HITL simulations, and the possible influence of simulated ATC all have impact on the simulation results. Types of results include probabilities and delays of responses, numbers of LDWC conditions and their context, and feedback by RPs on their situation awareness (SA) and overall impression of the ACAS/DAA system, see e.g. [21] for HITL results on ACAS Xu.

C. Uncertainty in validation

Given the key role of simulation in design and approval of ACAS/DAA systems, it is important that they properly represent the interactions and uncertainties of the sociotechnical system of ACAS/DAA encounter-scenarios. Fig. 2 provides a schematic overview of interacting uncertainties in an encounter of an aircraft pair (*i,j*). Given particular aircraft states $\mathbf{x}_i, \mathbf{x}_j$ there are various processes that lead to sensor errors $\boldsymbol{\varepsilon}_i, \boldsymbol{\varepsilon}_j$ in the estimates that are input of the ACAS/DAA system, such as pressure altitude errors and errors in the slant range and bearing measurements. These sensor errors lead to variability $\tilde{\mathbf{a}}_i$ in the advisories with respect to the ACAS/DAA output $\bar{\mathbf{a}}_i$ that would be attained in a deterministic simulation without sensor errors. For instance, such variability may manifest itself as a different type of RA, not having an RA, an RA at a different time, or other RWC bands. Since coordination for RAs is used between the systems of aircraft *i* and *j*, variability in an RA can also have impact on the RA of the other aircraft. Next there exists variability in the performance of (remote) pilots in responding to the advisories, e.g. response delay, not responding, responding with different acceleration, or responding with particular margins to the advice. These kinds of aspects contribute to variability $\tilde{\mathbf{u}}_i$ with respect to control input $\bar{\mathbf{u}}_i$ that would be attained in a deterministic simulation. This results in variability in the aircraft state $\tilde{\mathbf{x}}_i$ with respect to a state $\bar{\mathbf{x}}_i$ in a deterministic simulation. As state estimates and ACAS/DAA advisories are regularly updated (typically every second), there are feedback loops from the aircraft states, which can have further impact on the closed-loop dynamics of the overall sociotechnical system. For the aircraft pair this implies, amongst other things, uncertainty in CPA and LDWC conditions.

ACAS validation studies have primarily used deterministic simulations of large sets of encounters, while accounting for some sensor errors and pilot performance variability by postprocessing of these deterministic simulation results [17, 18]. For instance, the impact of pressure altimetry errors on NMAC probability has been evaluated by assuming that the VMD in an encounter equals the VMD observed in a deterministic simulation plus an error chosen from a probability distribution that represents the combined altimetry errors of the involved aircraft (see also [9] for an extended explanation). In the context of Fig. 2 this means that the impact of altimetry errors $\boldsymbol{\varepsilon}_i, \boldsymbol{\varepsilon}_j$ is evaluated based on the altitude differences of $\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j$, neglecting other uncertainties and the nonlinear dynamics of the closed-loop system. The impact of a pilot non-response probability on NMAC probability has been evaluated by combining deterministic simulations where both pilots apply standard response with deterministic simulations, in which one of the pilots does not respond and the other pilot responds standardly [17, 18]. This is only a limited representation of pilot performance variability, which neglects cases where both pilots do not respond, other elements of pilot performance variability, and the nonlinear dynamics of the overall sociotechnical system in an encounter.

III. Agent-based Modelling and MC Simulation

A stochastic dynamic agent-based modelling and simulation (ABMS) approach is useful to systematically describe the dynamics of interacting stochastic processes in sociotechnical systems, such as encounter-scenarios. It provides a modular and transparent way of structuring a model, thus supporting systematic analysis, both conceptually and computationally. An agent-based approach was followed in the development of the Collision Avoidance Validation and Evaluation Tool (CAVEAT) for EUROCONTROL by NLR

and everis/NTT-Data, which supports retrospective and prospective analysis of ACAS/DAA-supported encounter-scenarios [22]. Next, key elements of the models are concisely presented (see details in [23]).

A. Flight performance

The flight performance is modelled by the following entities:

- *Aircraft type*. The aircraft type determines the aircraft performance characteristics and limitations, and whether it is manned or unmanned.
- *Aircraft trajectory*. This is a model of the trajectory achieved by the aircraft, based on flight control input by the automatic flight control system, the pilot flying, or the remote pilot.

B. Flight management system

The flight management system (FMS) is modelled by the following entities:

- *Flight plan.* It is assumed that the flight plan is based on the 4D trajectory of a particular encounter.
- *Automatic flight control system (AFCS).* The AFCS can be chosen as the agent that responds automatically to RAs for manned or unmanned aircraft.
- *Flight instrument system.* This provides information to the pilot flying about flight states and it shows ACAS advised vertical speeds.

C. Ownship state estimation

This set of models represents the estimation of ownship aircraft states that are used as input of ACAS or that are communicated to other aircraft by its transponder (mode S, mode C, ADS-B):

- *Pressure altitude estimation.* Model for estimation of ownship pressure altitude, including error model.
- *Radio altitude estimation.* Model for the estimation of height above terrain, including error model.
- *Heading estimation.* Model for the ownship heading estimation, including error model.
- *GNSS state estimation.* Model for the GNSS-based estimation of ownship geodetic position and ground speed, including error model.

D. Intruder measurement & Data communication

This set of models represents intruder measurement & data communication by transponder-based interaction between aircraft (Mode S, Mode C, ADS-B) and air-to-air radar (ATAR):

- *Bearing measurement by transponder.* Model for transponder-based measurement of the bearing angle of nearby aircraft, including error model.
- *Range measurement by transponder.* Model for transponder-based measurement of the range of a nearby aircraft, including error model.
- *Transponder communication.* Model for transponder-based data communication with nearby aircraft, including mode S, mode A/C.
- *ADS-B communication.* Model for data transfer between aircraft using ADS-B.
- *Air-to-air radar.* Model for surveillance by ATAR providing the relative position and speed of nearby aircraft, including error models.

E. Sensor error models

The ownship state estimation and intruder measurement process models all account for sensor errors $\boldsymbol{\varepsilon}_i, \boldsymbol{\varepsilon}_j$ (Fig. 2). The sensor error models may represent static (bias) and variable (jitter) error components. The bias is chosen from a probability distribution at the start of a scenario, e.g. a pressure altitude bias chosen from a normal distribution with an altitude dependent standard deviation (SD) [1, 3]. The jitter component is updated at every simulation timestep and it is typically represented by a first-order autoregressive process, parameterized by a SD and an autocorrelation factor. Parameter values for the sensor error models are mostly based on performance standards in the relevant MOPS [1, 3, 6].

F. ACAS & DAA systems

The ACAS and DAA systems are the core of the encounter-scenario simulations. The CAVEAT software integrates the following libraries:

- *TCAS II.* C++ libraries of TCAS II algorithms [1, 10] were developed by the MITRE corporation and subsequently adjusted for EUROCONTROL by Evosys.
- *ACAS Xa.* The software of ACAS Xa V15R4 [24] and ACAS Xa Revision-A [3] consists of C++ libraries that were developed and validated by NTT Data and NLR for EUROCONTROL.
- *ACAS Xu*. The software of ACAS Xu [14] consists of Julia libraries that were developed and validated by NTT Data and NLR for EUROCONTROL.

G. Pilot flying

The performance of the pilot flying (PF) is modelled by the following components:

- *Situation awareness.* The SA is a time dependent state which describes the PF's understanding of the ACAS RAs, aircraft states and the flight plan.
- *Response mode.* The PF response mode describes whether the PF responds to an RA. It includes conditional probabilities for response, such as the condition that the pilot responded to a previous RA.
- *Delay.* The response delay is composed of a constant preparation delay of 2.5 s for initial RAs only, and an action delay that is constant (2.5 s) or chosen from a lognormal distribution (mean 2.5 s, SD 1.5 s).
- *Strength.* The vertical acceleration applied by the PF are constants of 0.25 g or 0.35 g (for reversal or increase rate RAs), or they are chosen from a lognormal distribution (mean 0.25 / 0.35 g , SD 0.04 g).
- *Flight control actions.* The PF always follows the planned trajectory in the horizontal plane. For a preventive RA, the PF ensures that the vertical speed remains within the rate limitation. For a corrective

RA, the PF adjusts the vertical speed towards the advised speed. After a Clear Of Conflict (COC) RA, the vertical speed is adjusted to the speed in the flight plan.

H. Remote pilot

The performance of the remote pilot (RP) is modelled by the following components:

- *Situation awareness.* The SA is a time dependent state which describes the RP's understanding of the ACAS Xu RAs and RWC guidance, aircraft states and the flight plan. It also includes decision rules regarding the RWC guidance, such as margins to RWC bands.
- *Response modes.* The RP response modes for CA and RWC describe whether the RP responds to RAs or RWC guidance in one of the following manners: not at all, only horizontally, only vertically, both horizontally and vertically.
- *Delay.* There are different delays for responding to RAs and to RWC guidance. Each delay consists of a constant preparation delay and an action delay, which may be constant or chosen from a lognormal distribution. In responding to RWC guidance there may be an additional ATC coordination delay.
- *Strength.* The model for the RP response strength describes the vertical accelerations and the rates of turn applied for RAs and RWC guidance.
- *Flight control actions.* RP flight control actions describe the integrated impact of the SA, the response mode, the closed loop delay, and the response strength of the RPAS manoeuvres, where it is assumed that horizontal and vertical movements are independent.

I. UA control station

The model for the UA control station represents latencies for command & control (C2) uplink and downlink. These delays contribute to the closed loop delay.

J. Encounters

The encounters for the simulation of manned aircraft scenarios all include two equally ACAS equipped aircraft that come within NMAC range at flight level 120. Aircraft speeds are in a range of 250 to 300 kt, and each encounter is simulated from 75 s before to 15 s after CPA. Four sets with 50,000 randomly generated encounters each are distinguished (see details in [9]):

1. A set where each aircraft has straight trajectories, flying level, climbing, or descending;
2. A set where one or both of the aircraft make a (horizontal) turn;
3. A set where one or both of the aircraft have a vertical rate change, meaning they level off or initiate a climb or descent;
4. A set where one or both of the aircraft make a horizontal turn and have a change in vertical rate.

In these sets no probabilities have been associated with the types of encounters. So the sets are not realizations of probabilistic encounter models like [15, 16], but they rather represent encounters with different degrees of variability.

The encounters for the simulation of unmanned aircraft scenarios all include two RPAS that come within NMAC range, where either one or both RPAS are equipped with ACAS Xu. Aircraft are flying at flight level 80 along straight trajectories with speeds of 110 or 130 kt, and each encounter is simulated from 300 s before to 300 s after CPA.

K. Types of simulation

The agent-based model can be simulated using deterministic or stochastic settings. A deterministic model allows for the traditional type of simulation, where each encounter is simulated once and statistics are gathered over the set of encounters. In a stochastic setting, the model's performance is evaluated by a number of MC simulation runs for each encounter, and statistics are gathered over combined sets of simulation runs and encounters.

IV. Results

This section presents a series of simulation results by the stochastic dynamic agent-based model, which show the diverse impact of sources of variability and interconnections between the elements of the sociotechnical system. Section A presents illustrative simulation runs and statistics for the impact of sensor errors on encounters of TCAS II and ACAS Xa equipped manned aircraft. Section B shows results for the impact of pilot performance variability for such ACAS-supported encounters. Section C provides illustrative results for encounters between ACAS Xu equipped RPAS and the impact of sensor errors and RP performance.

A. Impact of sensor errors in aircraft encounters

This section presents a number of examples of encounters that illustrate the impact of sensor errors in the agent-based models explained in Section III. In these cases, either the pilots respond according to the ICAO standard model, or a single pilot may not respond. Fig. 3 shows three runs of an encounter between TCAS II equipped aircraft that are flying level with AC2 slightly (84 ft) above AC1. In the deterministic simulation (Fig. 3a,b) this encounter leads to a Descend (DE) RA for AC1 and a Climb (CL) RA for AC2, followed by Level Off (LO) RAs for both aircraft, such that the VMD is increased to 684 ft. In MC simulation runs the same pattern can be observed with slightly adapted timing of the RAs due to the sensor errors and resulting change in VMD. An example is the realization in Fig. 3c where the VMD is 734 ft, which is 50 ft larger than achieved in the deterministic run. A different type of result that can be attained for the same encounter is shown in Fig. 3d. Here the upper AC2 gets a Descend RA and the lower AC1 receives a Climb RA, leading to a VMD of 516 ft (168 ft smaller than in the deterministic simulation). This means that actually Crossing Climb and Descend RAs are provided, which are not indicated as such to the pilots. These can be the result of pressure altitude errors, such that the ACAS is (erroneously) aware that AC2 is below AC1.

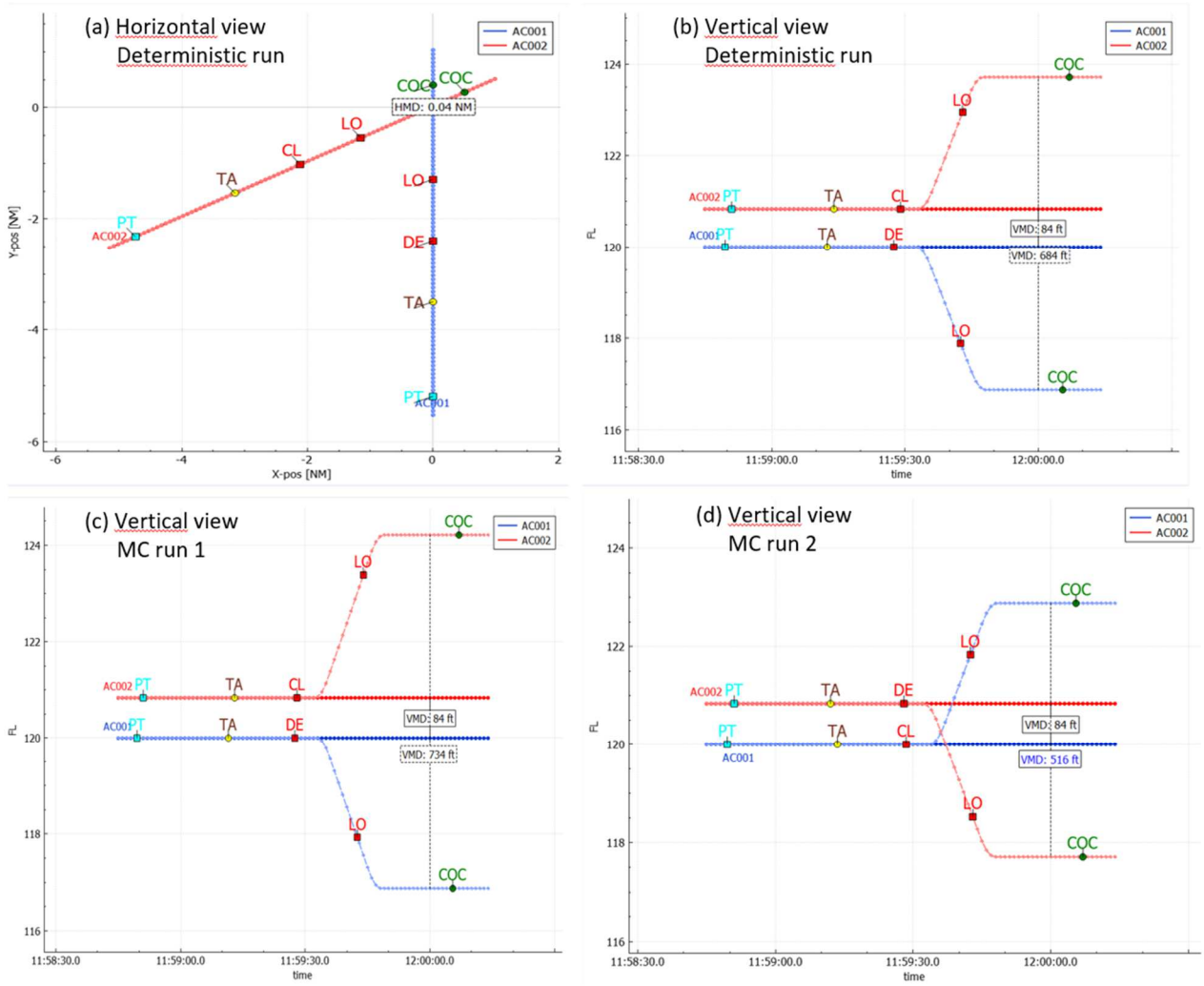

Fig. 3. Illustration of TCAS II [1] advisories and resulting trajectories for simulation of an encounter without sensor errors: (a) horizontal view, (b) vertical view; and for two MC simulation runs of a scenario with sensor errors: (c) vertical view of run 1, (d) vertical view of run 2.

Fig. 4 shows another example of differences in RAs for a same encounter due to sensor errors. In this case AC1 is flying level and AC2 is descending from above. In a deterministic simulation for TCAS II (Fig. 4a,b), AC2 receives a Level Off RA and AC1 receives a Descend RA, leading to a VMD of 1192 ft. Similar results can be obtained in MC simulation runs, such as in Fig. 4c leading to a VMD of 1242 ft (50 ft above the deterministic simulation). However the sensor errors can also lead to the case that AC1 does not get any RA as shown in Fig. 4d, leading to a VMD of 980 ft (212 ft below that in the deterministic simulation).

In the examples of both Fig. 3 and Fig. 4 the sensor errors lead to small differences in VMD (50 ft) in comparison with the deterministic simulations if the same RA sequences exist (with some timing variation), whereas they lead to large VMD differences (168 and 212 ft) if different RA sequences exist. If one would assume that differences in VMD can be described by a normal distribution with standard deviation of 76 ft (which is the combination of altimetry errors of both aircraft), then differences of 50 ft are very likely, while the likelihoods of attaining the large differences are small. However, the assumption that the VMD can be described by a normal distribution does not hold, because of the nonlinear dynamics in the encounter-scenarios, as manifested by the differences in the RA sequences.

While the above examples highlight the impact of sensor errors on VMD differences, Fig. 5 shows an example where sensor errors can result in an NMAC. It considers an encounter-scenario with ACAS Xa equipped aircraft where both aircraft are descending and only the pilot of AC1 responds to RAs. In the deterministic simulation (Fig. 5a,b) AC1 first gets a Maintain Descent (MDE) RA, but the RA is updated to a Reversal to Climb after 5 s, finally leading to a VMD of 1376 ft. In MC simulation runs with sensor errors, similar RA sequences and VMDs can be attained, such as illustrated in Fig. 5c. However, the sensor errors can also lead to an NMAC as shown in Fig. 5d, due to the subsequent RA of the already descending AC1 being Increase Descent (IDE).

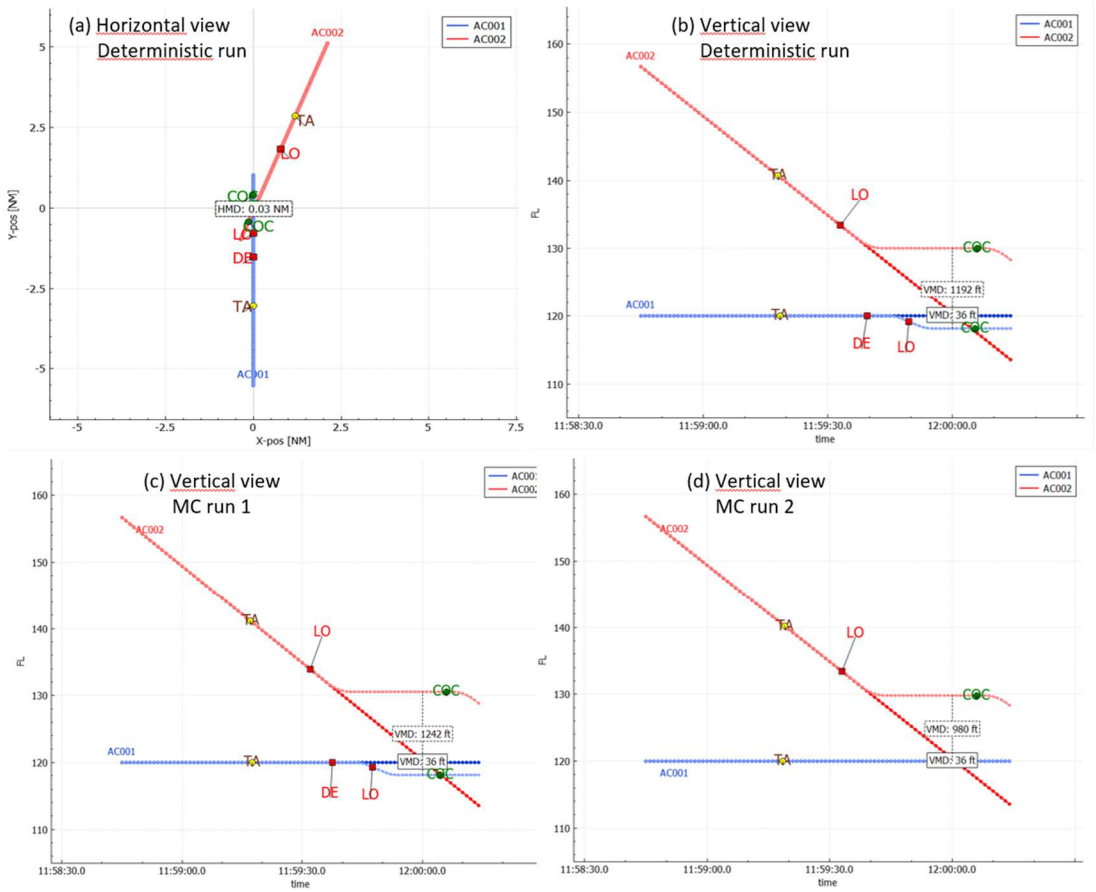

Fig. 4. Illustration of TCAS II [1] advisories and resulting trajectories for simulation of an encounter without sensor errors: (a) horizontal view, (b) vertical view; and for two MC simulation runs of a scenario with sensor errors: (c) vertical view of run 1, (d) vertical view of run 2.

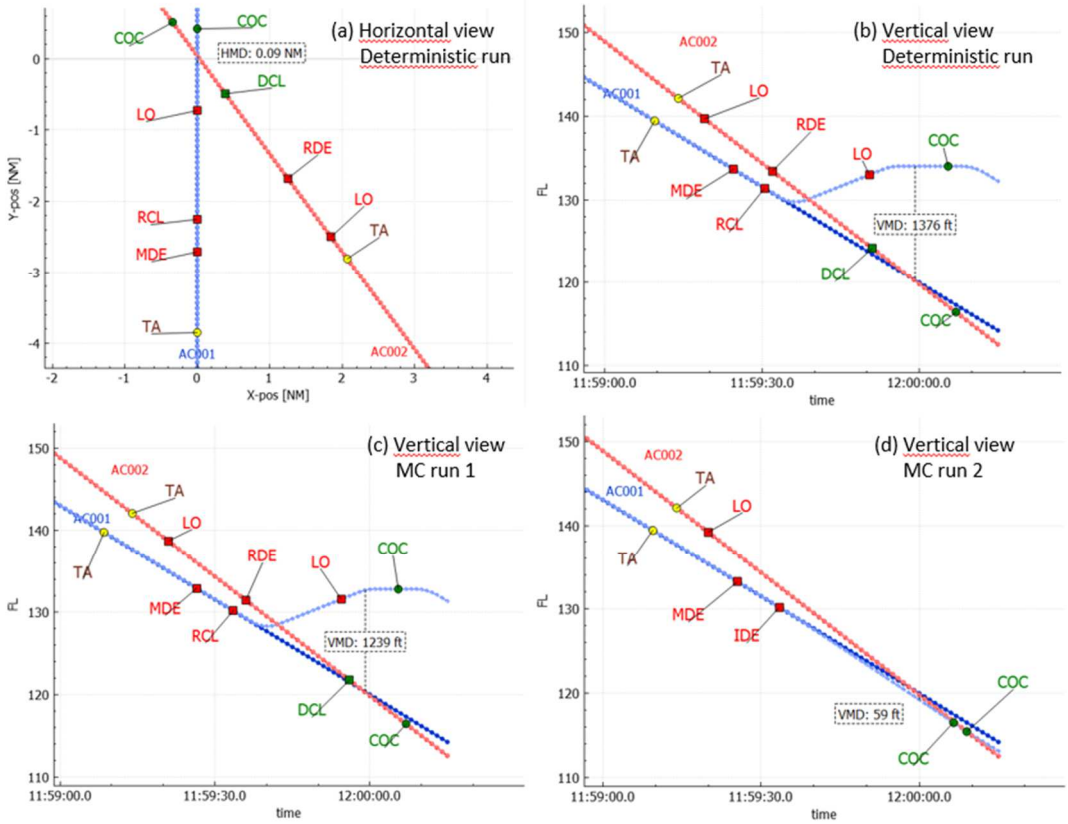

Fig. 5. Illustration of ACAS Xa [3] advisories and resulting trajectories for simulation of an encounter without sensor errors: (a) horizontal view, (b) vertical view; and for two MC simulation runs of a scenario with sensor errors: (c) vertical view of run 1, (d) vertical view of run 2.

In extension of the example of Fig. 5, distributions of VMDs for scenarios with straight encounters and sensor errors that have MC simulation runs with at least one NMAC are shown in Fig. 6 for TCAS II and ACAS Xa. It follows that the distributions have a peak for VMDs smaller than 200 ft and a peak for larger VMDs near 1000 ft for TCAS II and near 1300 ft for ACAS Xa. It means that the (limited) sensor errors lead to divergence: either a small or large VMD. This is an example of the butterfly effect in nonlinear system dynamics.

The highlighted nonlinear system dynamics implies that the customary altimetry error postprocessing, which stems from [25] and has still been used for validation of ACAS Xa [17, 18], is not appropriate. In particular this approach falsely assumes that NMAC probability can be estimated by adding an altimetry error from a probability distribution for the joint altimetry error of a pair of aircraft to the VMD achieved in a deterministic simulation of an encounter.

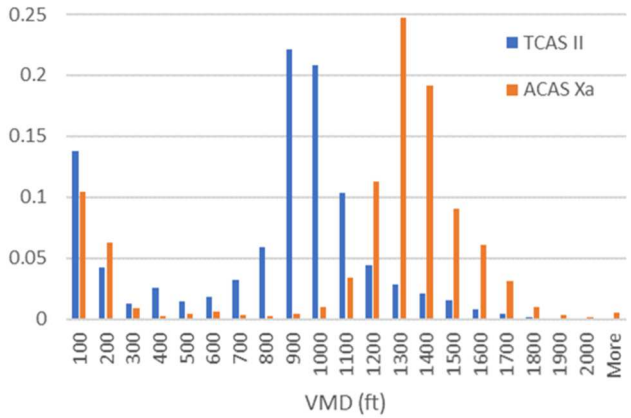

Fig. 6. Histograms of the relative frequencies of VMD for critical encounters with straight trajectories having at least one unresolved NMAC (source [9]).

TABLE I. RELATIVE DIFFERENCE IN P(NMAC) VALUES ESTIMATED BY MC SIMULATION VERSUS POSTPROCESSING OF ALTIMETRY BIAS [9]

Encounters	**P(NMAC) estimate increase**	
	TCAS II	***ACAS Xa***
Straight	0%	0%
Horizontal turns only	5.6%	63%
Vertical rate changes only	8.0%	20%
Horizontal turns & vertical rate changes	5.8%	10%

To achieve insight into the impact of this postprocessing step, Table I shows the relative difference in P(NMAC) estimates based on MC simulation of altimetry biases versus P(NMAC) estimates based on altimetry error postprocessing (see details in [9]). In both approaches it is assumed that the altimetry errors are normally distributed (as per the ACAS MOPS [1, 3]). It follows from this table that for encounters with straight trajectories the P(NMAC) estimates are not different (all estimates are about zero). However, for all other encounter sets with non-zero P(NMAC) values, the estimates as achieved by altimetry error MC simulation are larger than those achieved by altimetry error postprocessing. Remarkably, the differences are considerably larger for ACAS Xa than for TCAS II. A potential explanation is that the tuning of the reward function and TRM algorithms of ACAS Xa in an environment that applied altimetry error postprocessing has led to a bias in its logic that makes it less robust.

B. Impact of pilot performance in aircraft encounters

Key elements of pilot performance variability concern the probability of response and the delay and strength of response. In Fig. 5 an example was already provided for a scenario with one pilot responding to RAs. It illustrates that the impact of a non-responding pilot may critically depend on the sensor errors.

Fig. 7 shows another example for an encounter between two descending aircraft, where both aircraft make turns leading to RAs by ACAS Xa. Fig. 7a,b shows the result of a deterministic simulation with standard pilot response. AC1 receives a Maintain Descent (MDE) RA, while AC2 receives a Level Off RA, leading to a VMD of 659 ft. Fig. 7c shows a scenario where the pilot of AC2 does not respond. Following the before mentioned initial RAs, AC1 receives Reversal to Climb and Increase Climb RAs, leading to a VMD of 450 ft. Fig. 7d shows a MC simulation run of a scenario with the pilots of both aircraft responding, but with variability in the response delay and strength. In this realization especially the pilot of AC2 is responding slowly. As a result, AC2 receives a Reversal to Descent RA at the time that it just responded to the initial Level Off RA. This leads to an NMAC with AC1, which has started to climb following the Reversal to Climb RA. So in this example it turns out that the result of a prolonged response delay is more severe than the impact of a pilot who does not respond at all. It illustrates the complexity of the interactions between the ACASs and the performance of the pilots in attaining NMAC conditions.

Fig. 8 provides a comparison of various contributions to the probabilities of unresolved NMACs for TCAS II and ACAS Xa over the four types of encounters (straight, with horizontal turns, with vertical rate changes, with turns and vertical rate changes); see [9] for details. Here the base results (orange) are achieved for deterministic simulations without sensor errors and with standard pilot response. The sensor error results (yellow) are the additional risks due to sensor errors. The pilot performance results (green) are the additional risks due to three pilot models that all have stochastic delay and response strength, and furthermore: P1. response mode is chosen once (stubborn pilot), P2. response mode may switch for each RA (reconciliating pilot), and P3. pilot always responds. The results in Fig. 8 show that pilot performance is typically the most important factor for the effectiveness of ACAS, both for TCAS II and ACAS Xa. The extents of the pilot performance contributions largely depend on the type of pilot response model and their parameterization, but the types of encounters and the ACAS type also influence the P(NMAC) contributions. The base risks stemming from the types of trajectories in the encounters mostly provide the second-most important contribution. These base risks are especially prominent for encounters that include vertical rate changes with or without horizontal turns, as well as encounter scenarios with horizontal turns and TCAS II. ACAS Xa is considerably more effective in attaining low base risks than TCAS II for these types of encounters. For encounters with straight trajectories, both TCAS II and ACAS Xa are very effective, and the base risk is practically zero. The contributions of sensor errors to P(NMAC) are mostly the smallest. Especially in encounters with straight trajectories or involving only horizontal turns, these contributions are very small. In encounters with vertical rate changes, sensor errors play a more prominent role, and their contribution is larger for ACAS Xa than for TCAS II.

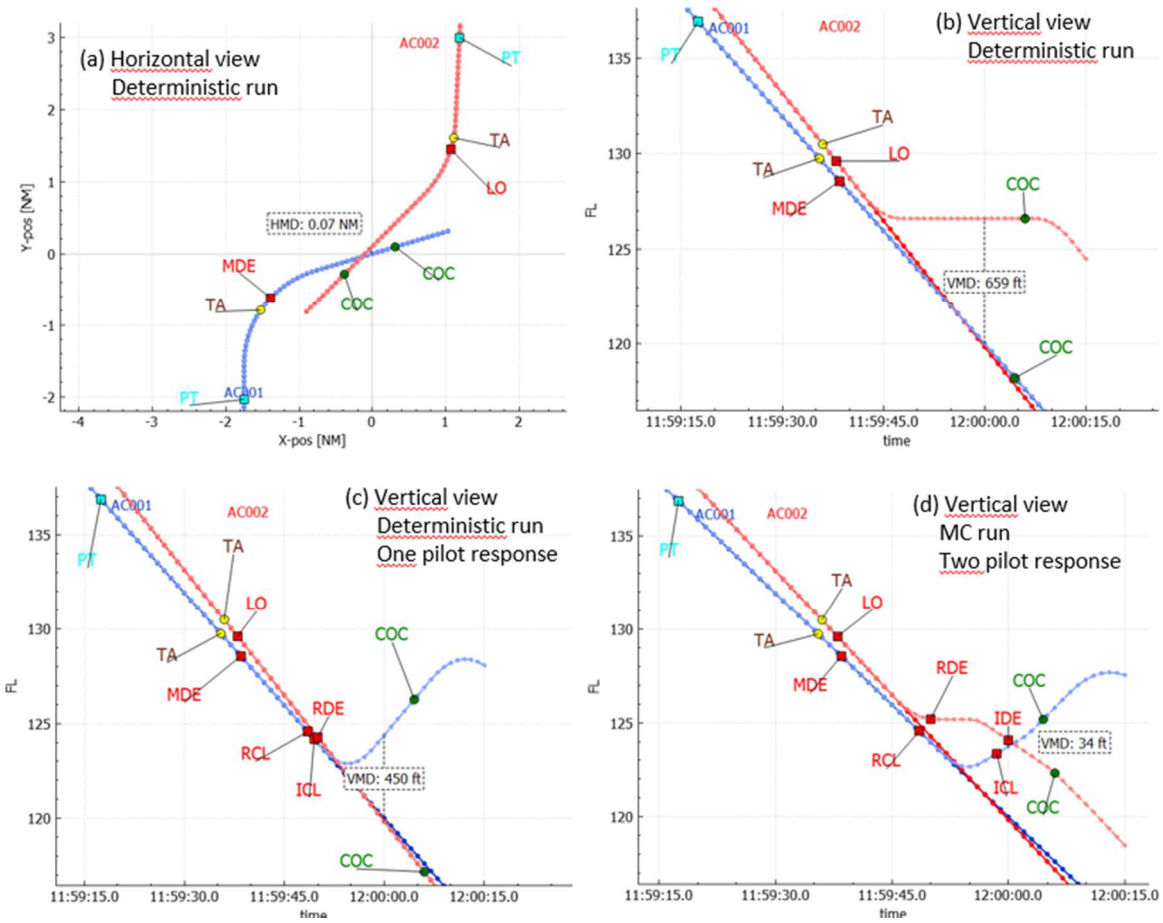

Fig. 7. Illustration of ACAS Xa [3] advisories and resulting trajectories for simulation of an encounter with standard pilot response (a) horizontal view, (b) vertical view; (c) vertical view if only pilot of AC1 responds, (d) vertical view of a MC simulation run with variability in pilot delay and strength (no sensor error).

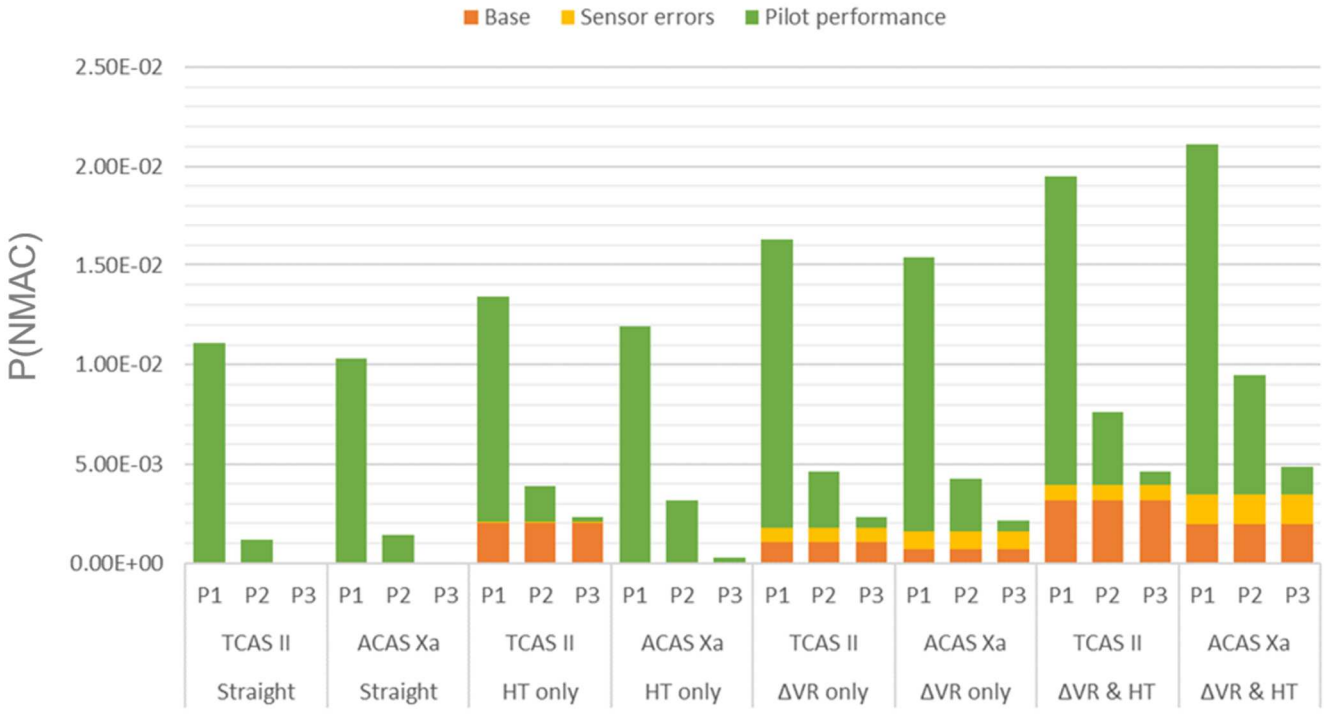

Fig. 8. Contributions of sources of variability to P(NMAC) for encounters leading to an NMAC without resolution (source [9]).

C. *Simulation of RPAS encounters*

Although the same type of simulation approach is used for evaluation of DAA-supported RPAS and for ACAS-supported manned aircraft, the additional dimensions and RP settings lead to considerably more diversity in results for DAA. For clarity this paper shows illustrative examples of encounters and manoeuvres that are in the horizontal plane only; see [26] for examples and statistics of RPAS manoeuvres in both the horizontal and vertical plane.

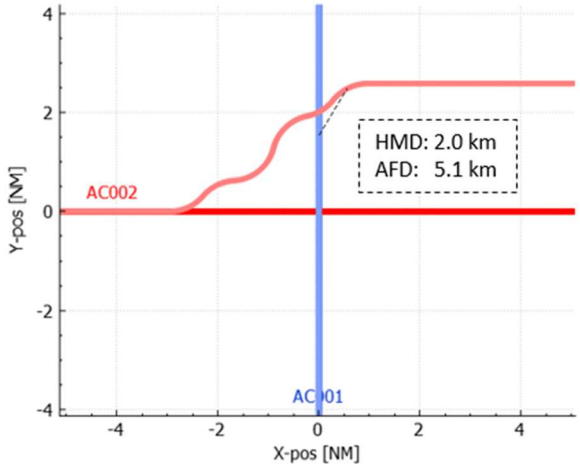

Fig. 9. Deterministic simulation of encounter with angle of 90 deg between two drones with same speed where de RP of AC2 responds to ACAS Xu RWC guidance with a closed loop response delay of 12 s.

Fig. 9 shows the simulated response of an ACAS Xu equipped aircraft AC2 in an encounter with an unequipped aircraft that comes from the right under an angle of 90 degrees, where both aircraft have the same speed of 110 kt. In this example the RP of AC2 is responding with a closed loop delay of 12 s. It can be seen that while following the RWC guidance AC2 turns left and right in three sequences so as to pass in front of AC1 at a distance of 2.0 km with an additional flight distance (ADF) of 5.1 km.

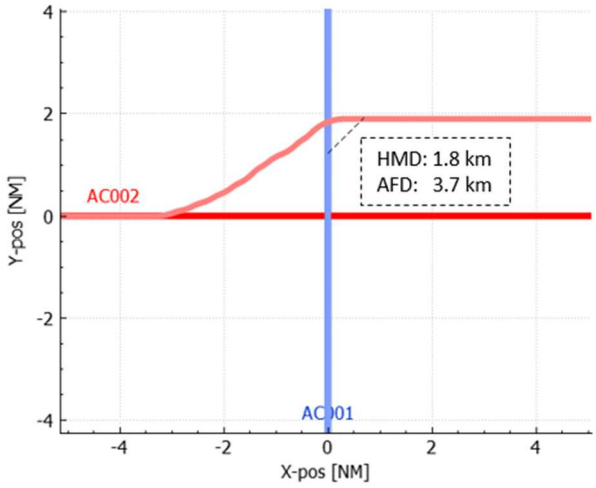

Fig. 10. Deterministic simulation of encounter with angle of 90 deg between two drones with same speed where de RP of AC2 responds to ACAS Xu RWC guidance with a closed loop response delay of 1 s.

The dynamics of the fluctuations in the trajectory depend on the closed loop delay as is illustrated in Fig. 10. Here a delay of 1 s (emulating an automatic response) induces a number of shallow fluctuations before AC2 passes in front of AC1 at 1.8 km, leading to an AFD of 3.7 km.

Whereas in the above two examples AC2 manages to pass AC1 after some fluctuations, the example in Fig. 11 shows that under a more acute encounter angle of 45 degrees AC2 does not manage to pass AC1. The aircraft pair ends up in a livelock condition and comes at a HMD of 1.4 km in the simulation. The term livelock stems from computer science and describes a situation where interacting processes are repeatedly changing their state without finishing their tasks. For this encounter a livelock condition also exists for smaller delays.

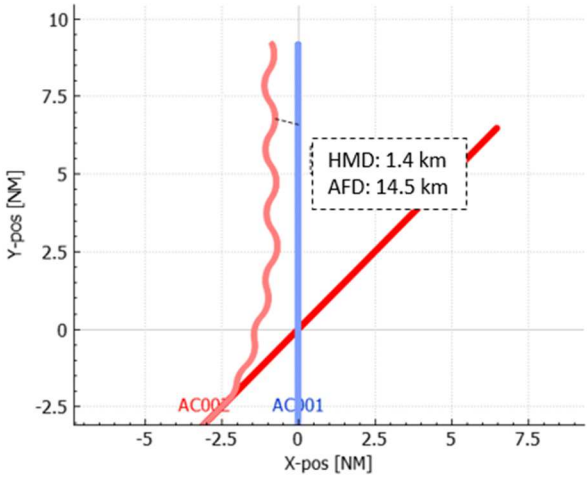

Fig. 11. Deterministic simulation of encounter with angle of 45 deg between two drones with same speed where de RP of AC2 responds to ACAS Xu RWC guidance with a closed loop response delay of 12 s.

The existence of a livelock condition depends on the aircraft speeds. Fig. 12 shows an example of an encounter-scenario that is the same as Fig. 11, except for a larger speed of AC2. As a result, AC2 manages to pass in front of AC1 after a number of fluctuations, inducing an AFD of 10.6 km.

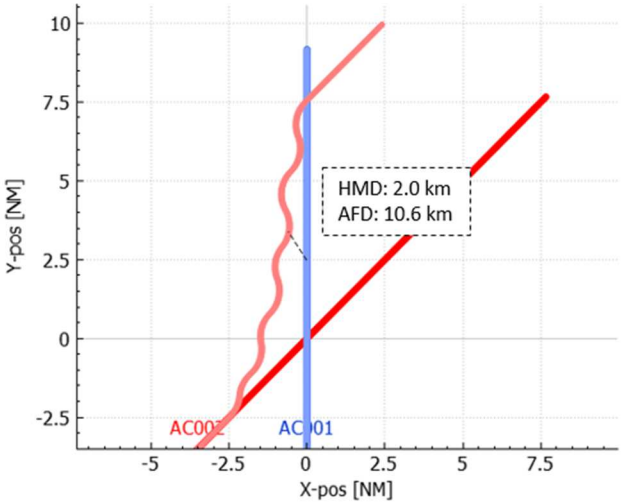

Fig. 12. Deterministic simulation of encounter with angle of 45 deg between two drones with different speeds where de RP of AC2 responds to ACAS Xu RWC guidance with a closed loop response delay of 12 s.

In stochastic encounter-scenarios, which include sensor errors and delay variability, several patterns can arise. Also here the aircraft can end up in a livelock condition similar to Fig. 11. A different realization that can be attained is shown in Fig. 13. Here AC2 turns right towards AC1 and it passes behind AC1 with a HMD of 1.8 km. Although there is an LDWC in this example, the additional flight distance is limited to 3.3 km.

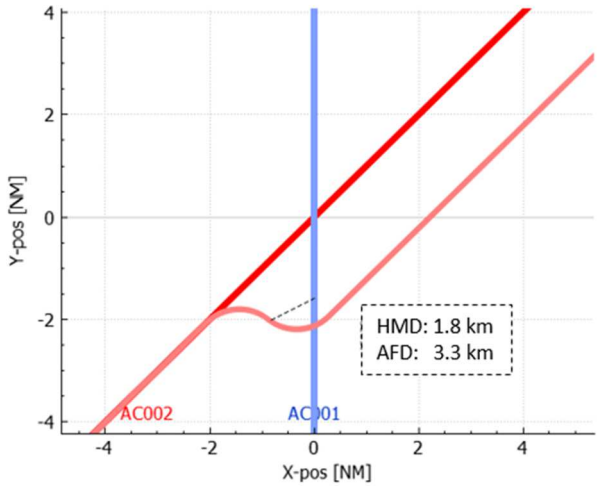

Fig. 13. MC simulation run of encounter with angle of 45 deg between two drones with same speed where de RP of AC2 responds to ACAS Xu RWC guidance (with stochastic delay and sensor errors).

In the above examples only AC2 manoeuvres in line with RWC guidance. This can be seen as consistent with the Right-of-Way (ROW) rules [27], specifying that if two aircraft are converging at approximately the same level then the aircraft having the other on the right shall give way. If both aircraft manoeuvre in line with the RWC guidance even more complicated trajectory patterns may arise, which depend on the encounter geometry and the dynamics of the agent-based system. Fig. 14 illustrates that also in such cases livelock conditions can exist. In this case both aircraft come into a pattern where they are basically pushing each other away without reaching their destination.

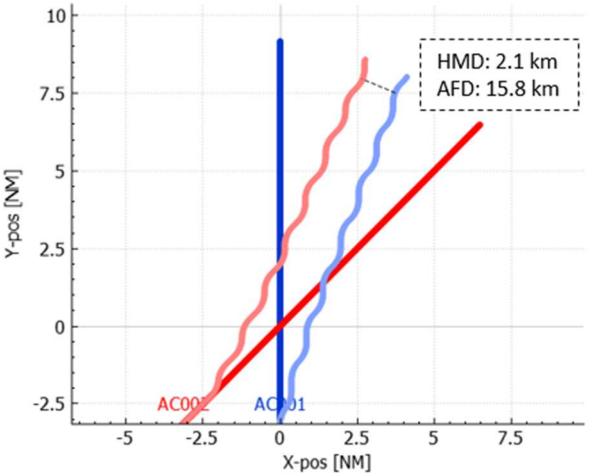

Fig. 14. Deterministic simulation of encounter with angle of 45 deg between two drones with same speed where de RPs of both aircraft responds to ACAS Xu RWC guidance with a closed loop response delay of 12 s.

In summary, the agent-based simulations of the DAA sociotechnical system show that strict adherence to the ACAS Xu RWC guidance can lead to livelock conditions and that its nonlinear dynamics are sensitive for sensor errors and RP performance variability. As shown in [26] livelock conditions can also exist if the RPAS manoeuvre both horizontally and vertically in response to ACAS Xu RWC guidance.

V. Discussion

Since ACAS and DAA systems provide last-instance safety-critical functions for remaining well-clear and collision avoidance, it has been common practice to simulate the dynamics of the aircraft in encounters to support the validation and feedback to design for these systems. In such simulations considerable emphasis has been placed on encounter models, describing the characteristics and probabilities of ways that pairs of aircraft may come at small distances. However, the variability of other aspects of the sociotechnical ACAS and DAA systems have typically been evaluated to a limited extent only by postprocessing of results of deterministic simulations, like accounting for altimetry errors or pilot non-response post-simulation.

There are several types of implications of such limited representation of variability in the sociotechnical ACAS and DAA systems. The most obvious implication is that the effects of variability are not evaluated. For instance, by considering only static biases in pressure altimetry, the impact of variable jitter components remains unknown. By not representing errors in GNSS state estimation, bearing and range measurements, their effects stay obscure. By leaving out possible variability in (remote) pilot performance, like variability in response delay, strength and probability, and various decision rules, no insight is attained in such human operator performance. As a way forward, in the agent-based models of this paper many types of variability were included and their impact was evaluated in MC simulation of encounter-scenarios. These simulation results show that human performance has the largest impact on the effectiveness of ACAS and DAA systems. In particular, as illustrated by Fig. 8, differences in pilot response are the most prominent factor driving ACAS impact. In the DAA evaluation, the behaviour of the RP model to steer back towards the planned course after the ACAS Xu RWC bands allowed so, contributed to the existence of livelock conditions shown in Section IV.C.

Another key implication of the limited representation of variability is that interactions between uncertainties are not well accounted for. As explained in Fig. 2, the ACAS/DAA sociotechnical system has nonlinear stochastic dynamics, where the impact of sensor errors, pilot performance, and aircraft trajectories are interrelated and cannot be described by linear superpositions. In Section IV several examples and statistics were given of limited sensor errors, which lead to differences in RA sequences and thereby to considerable differences in the VMD or the occurrence of an NMAC. Such divergence in VMD is an example of the butterfly effect in nonlinear system dynamics. Due to such nonlinear relations, the customary approach of postprocessing of altimetry errors gives wrong results. It follows from a systematic evaluation of encounter sets shown in Table I that such postprocessing can lead to considerable underestimation of NMAC probabilities.

Furthermore, the limited representation of variability in the ACAS/DAA sociotechnical system has repercussions on the feedback to design as illustrated in Fig. 1. If the performance of the sociotechnical system is not well accounted for, then important aspects may not be found in the validation phase, and designers may not consider them and rather focus on other (possibly less relevant) aspects. This feedback to design is especially relevant for optimization-based design approaches as used in ACAS X. In particular, the reward function that is used in the DP optimization and the

online cost parameters may be tuned for achieving particular desired performance metrics, which may be biased if the performance variability has been represented insufficiently. While the settings of a rule-based system (like TCAS II) may also be tuned and become biased, the expert-based interpretation of such rules is a support to avoid excessive biases. The larger increase in NMAC probabilities for ACAS Xa versus TCAS II due to altimetry bias postprocessing as shown in Table I may have been caused by excessive finetuning in the ACAS Xa optimization.

Finally, the limited representation of variability in the ACAS/DAA sociotechnical system is problematic for the approval of new ACAS/DAA systems by certification authorities. Obviously, the validation exercises that are used as a basis for approval should be different from those used in the feedback to design. This is needed to assure an independent evaluation that can find potential biases that may have been developed during design. For instance, this means that the robustness for different encounter models should be assured (as is common practice). However, also a sufficient range of other types of variability should be independently evaluated in validation exercises for approval. If particular types of variability (in pilot performance, in CNS performance) are lacking, then the ACAS/DAA sociotechnical system may not be robust for such sources. Also if the same models describing the variability would be used in the validation for feedback to design as in the validation for approval, the latter would not be able to find potential biases that have been incorporated due to feedback to design cycles.

The focus in this paper has been on evaluation of ACAS and DAA systems, providing key results for TCAS II, ACAS Xa and ACAS Xu. As explained, these are all extensive systems of which the functioning is determined by many interacting algorithms. In addition to such extensive rule basis, the functioning of ACAS Xa and Xu is driven by the optimization-based cores of their threat resolution modules. Furthermore, the ACAS/DAA safety-critical functions are achieved in direct connection with human operators. As such, ACAS and DAA systems are prime cases of human-centric AI-based systems, and their validation and approval should be aligned with developing requirements and guidelines for AI trustworthiness [8, 28]. While that work has mostly focused on supervised learning approaches in AI, the validation of ACAS and DAA systems would require the inclusion of more diverse methods for evaluation of their rule-based and optimization-based systems and their time-critical interaction with human operators. It was shown in this paper that stochastic dynamic agent-based modelling and simulation of the sociotechnical system encompassing AI-based systems provides a broad repertoire of results that can support such evaluation.

In the learning assurance scheme for supervised learning of [28] considerable emphasis is placed on ensuring that the resulting trained models possess sufficient generalisation and robustness capabilities. Part of this is data management to ensure that different data is used for model training and for validation and testing, such that the achieved models are not overly tuned to the specific training data. Similarly, for the design of rule-based or optimization-based AI systems it must be assured that they are not overly tuned to the specific models that have been used in fast-time or HITL simulations for the feedback to design. The finding that in some cases the optimization-based ACAS Xa is less robust than the rule-based TCAS II is an indication that the design of ACAS Xa may have been more tuned to the models and methods used in the feedback to design.

Certification by extensive testing of AI-based systems in large numbers of simulated hours, including faults and contingencies, has been called 'licensing of AI systems' [29]. The agent-based modelling and simulation results of ACAS/DAA shown in this paper support such testing for sensor errors and human performance variability (including the possibility of non-response). These kinds of conditions are normal conditions in ACAS/DAA operations. Other types of conditions that need to be considered in a complete safety assessment [30] are abnormal conditions, e.g. extreme weather, and faulted conditions, where the integrity of system components is affected. Agent-based modelling and specific types of MC simulation and risk decomposition can be used to address such broader scope of conditions [31], such as e.g. shown for DAA supported operations [32].

VI. Conclusions

In conclusion, the nonlinear dynamics and stochastic influences in ACAS/DAA equipped aircraft encounter-scenarios can critically affect their manoeuvres, and the safety and efficiency of the operations. Herein, the pilot performance has the largest impact on the overall simulated performance of the ACAS/DAA-supported operations. As such it is important to employ sufficiently sophisticated human performance models for validation of ACAS and DAA systems. While human performance models will never describe all nuances of situation awareness and performance by human operators, they are crucial for studying the emerging dynamics of sociotechnical systems in large numbers of varying conditions and for understanding the critical factors and uncertainties that should be addressed in extended studies (e.g. focused human-in-the-loop simulations) and in design. This is especially important in the design of optimization-driven AI-based systems like ACAS Xa/Xu, where there exists a close connection between evaluation of the performance of the overall system and tuning of meta-parameters of the optimization process. Stochastic dynamic models describing all elements of the sociotechnical systems encompassing the ACAS/DAA system and transparent documentation of their combined behaviour are essential to this end.

References

[1] EUROCAE, "Minimum operational performance standards for Traffic Alert and Collision Avoidance System II (TCAS II), Volume I," Malakoff, France, 2008.

[2] M. J. Kochenderfer, J. E. Holland, and J. P. Chryssanthacopoulos, "Next-generation airborne collision avoidance system," *Lincoln Laboratory Journal,* vol. 19, no. 1, pp. 17-33, 2012.

[3] EUROCAE, "Minimum Operational Performance Standards for Airborne Collision Avoidance System X (ACAS X) (ACAS Xa AND ACAS Xo)," Saint-Denis, France, June 2023.

[4] RTCA, "Minimum Operational Performance Standards (MOPS) for Detect and Avoid (DAA) Systems," 18 March 2021.

[5] C. Munoz *et al.*, "DAIDALUS: Detect and Avoid Alerting Logic for Unmanned Systems," presented at the Proceedings of the 34th Digital Avionics Systems Conference (DASC 2015), Prague, Czech Republic, 2015.

[6] EUROCAE, "Minimum operational performance standards for Airborne Collision Avoidance System Xu (ACAS Xu): Volume I," December 2020.

[7] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*, Second edition ed. Cambridge (MA), USA: MIT Press, 2020.

[8] EASA, "Artifical Intelligence Roadmap 2.0: Human-centric approach to AI in aviation," European Union Aviation Safety Agency, May 2023.

[9] S. Stroeve, "What Matters in the Effectiveness of Airborne Collision Avoidance Systems? Monte Carlo Simulation of Uncertainties for TCAS II and ACAS Xa," *Aerospace,* vol. 10, no. 11, p. 952, 2023. https://www.mdpi.com/2226-4310/10/11/952.

[10] EUROCAE, "TCAS II Collision Avoidance System (CAS) Requirements Specification, Volume II ", Malakoff, France, September 2008.

[11] M. J. Kochenderfer, *Decision making under uncertainty*. Cambridge (MA), USA: The MIT Press, 2015.

[12] M. P. Owen, A. Panken, R. Moss, L. Alvarez, and C. Leeper, "ACAS Xu: Integrated collision avoidance and detect and avoid capability for UAS," presented at the 2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC), 2019.

[13] D. P. Bertsekas, "Rollout algorithms for discrete optimization: A survey," in *Handbook of combinatorial optimization*, vol. 5: Springer New York, 2013, pp. 2989-3013.

[14] EUROCAE, "Minimum operational performance standards for Airborne Collision Avoidance System Xu (ACAS Xu): Volume II Algorithm Design Description," December 2020.

[15] M. J. Kochenderfer, M. W. M. Edwards, L. P. Espindle, J. K. Kuchar, and J. D. Griffith, "Airspace Encounter Models for Estimating Collision Risk," *Journal of Guidance, Control, and Dynamics,* vol. 33, no. 2, pp. 487-499, 2010, doi: 10.2514/1.44867.

[16] G. Dean *et al.*, "Collision Avoidance Fast-time Evaluator (CAFE) Revised Encounter Model for Europe (CREME)," SESAR Joint Undertaking, Brussels, Belgium, 8 March 2022.

[17] FAA, "Post-FRAC Operational Validation Report," Federal Aviation Administration, Washington DC, USA, 10 August 2018.

[18] EUROCONTROL, "European Airborne Collision Avoidance System (ACAS) Xa Change Proposal (CP)1 validation report," Brussels, Belgium, 16 June 2022.

[19] ICAO, "Annex 10 Aeronautical telecommunications, Volume IV Surveillance and collision avoidance systems," International Civil Aviation Organization, Montreal, Canada, July 2014.

[20] J. E. Holland, M. J. Kochenderfer, and W. A. Olson, "Optimizing the next generation collision avoidance system for safe, suitable, and acceptable operational performance," presented at the Tenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2013), Chicago, USA, 2013.

[21] R. C. Rorie, C. Smith, G. Sadler, K. J. Monk, T. L. Tyson, and J. Keeler, "A Human-in-the-Loop Evaluation of ACAS Xu," presented at the 2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC), 11-15 Oct. 2020.

[22] S. H. Stroeve, H. A. P. Blom, C. H. Medel, C. G. Daroca, A. A. Cebeira, and S. Drozdowski, "Modeling and simulation of intrinsic uncertainties in validation of collision avoidance systems," *Journal of Air Transportation,* vol. 28, no. 4, pp. 173-183, 2020, doi: 10.2514/1.d0187.

[23] S. H. Stroeve and C. J. Villanueva Cañizares, "CAVEAT TAM - Models and Algorithms: Development of a Collision Avoidance Evaluation and Analysis Tool," Royal Netherlands Aerospace Centre NLR, Amsterdam, The Netherlands, August 2023.

[24] EUROCAE, "Minimum Operational Performance Standards for Airborne Collision Avoidance System X (ACAS X) (ACAS Xa AND ACAS Xo)," Saint-Denis, France, October 2018.

[25] M. P. McLaughlin and A. D. Zeitlin, "Safety study of TCAS II for logic version 6.04," Federal Aviation Administration, Washington D.C., USA, July 1992.

[26] S. Stroeve, C. J. Villanueva Cañizares, and G. Dean, "The critical impact of remote pilot modelling in evaluation of detect-and-avoid systems explained for ACAS Xu," *European Journal of Transport and Infrastructure Research,* vol. 24, no. 4, pp. 1-17, 2024, doi: https://doi.org/10.59490/ejtir.2024.24.4.7380.

[27] ICAO, "Annex 2 Rules of the air," International Civil Aviation Organization, Montreal, Canada, July 2005.

[28] EASA, "EASA Concept Paper: Guidance for Level 1&2 machine learning applications," European Union Aviation Safety Agency, Cologne, Germany, March 2024.

[29] S. Bhattacharyya, D. Cofer, D. J. Musliner, J. Mueller, and E. Engstrom, "Certification Considerations for Adaptive Systems," NASA, 2015.

[30] B. Rabiller and N. Fota, "Expanded Safety Reference Material (E-SRM) core document," EUROCONTROL, 17 Nov 2023.

[31] M. H. C. Everdij, H. A. P. Blom, S. H. Stroeve, and B. Kirwan, "Agent-based dynamic risk modelling for ATM: A white paper," EUROCONTROL, January 2014.

[32] S. Stroeve, B. Bakker, C. J. Villanueva Cañizares, and N. Fota, "Close proximity and collision risk assessment of drones and urban air mobility," presented at the 11th SESAR Innovation Days, Virtual conference, 2021.