

An Introduction to ACAS Xu and the Challenges Ahead

Guido Manfredi
ENGIE Ineo - Sagem UAS Chair
Ecole Nationale d'Aviation Civile
Toulouse, France
Email: guido.manfredi@enac.fr

Yannick Jestin
ENGIE Ineo - Sagem UAS Chair
Ecole Nationale d'Aviation Civile
Toulouse, France
Email: yannick.jestin@enac.fr

Abstract—According to a 2013 AUVSI report, delays in integrating Unmanned Aerial Systems (UAS) into the National Airspace System (NAS) could cost more than \$10 billions a year for the United States alone. Worldwide regulatory bodies are under pressure by the UAS industry to accelerate the regulation process, but safety remains their main objective. One condition for the safe introduction of UAS in the NAS is for them to be equipped with a collision avoidance system. Though the existing Airborne Collision Avoidance System II (ACAS II) could have been an option, the transformations of air traffic management engaged through NextGen (US) and SESAR (Europe) led to the definition of a new ACAS based on new logics, namely ACAS X. Its definition contains in particular two variations : ACAS Xa, for large aircraft, and ACAS Xu, for unmanned aircraft.

As noted in a 2014 RTCA annual report, divide in technological knowledge between those experienced in ACAS II and those involved in the development of ACAS X is a concern. To help preventing this divide we believe it is essential to keep the community updated with the latest evolutions of the ACAS X standards. As work on Minimum Operational Performance Standards (MOPS) for ACAS Xu just started, it is of interest to know which parts of the MOPS are already decided, which remain flexible for the industries to make the difference and which are open research problems.

Being a member of the ACAS X family, ACAS Xu lays on the same foundations as the well defined ACAS Xa standard. This work proposes an introduction to the ACAS Xa/Xu common basis, as it is unlikely to change, including the general architecture and Collision Avoidance (CA) logics. It is followed by a presentation of concepts specific to ACAS Xu such as the tailored threat logic, horizontal CA logic, CA coordination and automatic responses. For the flexible part, we believe it mainly concerns the surveillance sources. Instead of a precise standard, the regulation is likely to ask for requirements on the sensors capabilities. A state of the art of recent works allows proposing minimum sensor performances and focusing on an essential set of sensors. This work is concluded by presenting future challenges that need to be addressed to build a safe ACAS Xu baseline and to extend it to smaller and lower altitude UAS.

I. INTRODUCTION

A recent AUVSI report [1] stated that each day where the UAS are not integrated in the National Aviation System (NAS) represents a loss of \$27.6 million dollars in economic impact, for a total of \$10 billion a year. Though these figures have been subject to discussion and skepticism [2], more conservative reports still speak about a potential market worth €1100 millions for Europe and €10 billion worldwide by 2020

[3]. However, before reaching their full potential, UAS civilian applications require easy access to the NAS. But integration of UAS in the NAS raises two main concerns: safety and privacy. Though the EUROCAE group identified privacy as the main concern for the public opinion [4], aeronautic communities still consider safety as the main obstacle.

In their roadmap for civil UAS integration [5], the European RPAS Steering Group identified six technological locks preventing a safe integration: detect and avoid, data communication links, integration into air traffic management, contingency measures, verification and validation, cyber security. Among these challenges, the *Detect and Avoid* (DAA) problem is one of the most significant. The Detect and Avoid consists in providing UASs with two functions: Traffic Avoidance (TrA) and Collision Avoidance (CA). Traffic Avoidance allows keeping a safe distance from other aircraft. Collision Avoidance allows last moment maneuver to avoid Near Mid-Air Collisions (NMAC). Traffic Avoidance is not required in controlled airspace as aircraft separation is ensured by air traffic control. For Collision Avoidance, it is currently performed by pilots with the help of dedicated avionics, e.g. TCAS II. However, with the always increasing traffic, existing collision avoidance logics are reaching their limits [6]. Modernisation efforts of Air Traffic Management (NextGen and SESAR) have dealt with this issue through the development of new CA technologies, indeed updating ACAS II would have been too costly. Solutions based on deterministic planning and probability thresholding have been considered but rejected because the former cannot handle a probabilistic model of the world and the later cannot account for the future of the world. The group of experts tasked with the development of this new CA technology settled on a decision theoretic planning method called ACAS X [6]. This method comes in different versions built around a common basis. Each version tackles a specific area of operation, ACAS Xa for large aircraft, ACAS Xo for special operations, ACAS Xu for unmanned aircraft and ACAS Xp for general aviation.

The logic used in the ACAS X family differs from the ones used in ACAS II. Indeed, ACAS II relies on deterministic methods As expressed in a 2014 RTCA-SC147 report [7], the appearance of this new CA technology (ACAS X) based on principles very different from existing technology (ACAS

II) threatens to create a divide between those with years of experience in ACAS II and those implicated in the development of ACAS X. Besides, a disparity exists among the UAS communities with some of them lacking the aviation culture needed to integrate UASs in the NAS.

The present paper aims at mending these gaps by presenting to the interested parties (research, industry and regulatory bodies) the principles behind ACAS Xu and its expected evolutions. A reminder of the collision avoidance problem is provided in Section II. Then, Section III reviews the existing solution: ACAS II. In Section IV, the baseline of ACAS X is introduced and its differences with ACAS II are highlighted. Concepts specific to ACAS Xu are described in Section V. Then, the surveillance source requirements are discussed in Section VI. The last Section proposes lines of research to build a safe ACAS Xu and to extend it to small UASs. It is followed by concluding remarks.

II. THE COLLISION AVOIDANCE PROBLEM

To compare CA solutions, it is essential to understand the underlying problem as well as typical hypothesis and metrics. For simplicity, let's consider encounters between just two aircraft. We take the perspective of one of them and call it the ownship, the second aircraft is called the intruder. Note that the intruder can belong to one of the two following families: cooperative traffic, which transponders facilitate sensing with cooperative sensors; non-cooperative traffic, which needs to be sensed with ownship non-cooperative sensors. In the rest of this paper, the term sensor is used to designate cooperative and non-cooperative sensors dedicated to the Detect and Avoid task, not payload sensors.

A. General Problem and Hypothesis

When flying, an aircraft is surrounded by a virtual collision volume shaped like a 200ft high and 500ft radius cylinder (Figure 1). Two aircraft are on a collision course if, with no change in trajectory, at some point one of them will enter the collision volume of the other. An collision volume violation is called a Near Mid-Air Collision (NMAC), at this point only providence can prevent an actual Mid-Air Collision (MAC).

Collision Avoidance systems exist to prevent NMAC by monitoring a collision avoidance threshold larger than the collision volume (Figure 1). If an intruder aircraft enters the collision avoidance threshold, it becomes a threat and needs to be avoided. Thus, the CA problem can be divided into two subproblems: threat detection (is this aircraft in the collision threshold ?) and threat resolution (how to avoid it ?).

When considering a collision problem, two hypothesis are usually made. First, as it is not possible to predict the intruder trajectory, it is assumed that the aircraft involved in the encounter have constant velocity vectors. This assumption allows predicting future positions by linear interpolation. Second, to simplify calculations, we consider that the conflict takes place in the horizontal plane. This effectively transforms a 3-D problem into a more tractable 2-D problem.

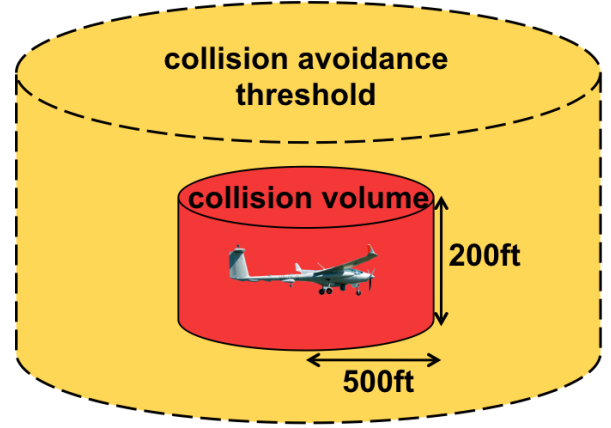


Fig. 1: Aircraft are surrounded by a collision volume and a collision avoidance threshold. Collision avoidance systems monitor the collision avoidance threshold and maneuver to prevent intruders from reaching the collision volume [8].

B. Distance and Time Metrics

Different distances are used to characterize an encounter. One of them is the slant range, i.e. the euclidean distance between the center of both aircraft. Two other distances will be useful in the following: relative vertical/horizontal distances. There are also dynamical variables, the closure rate between aircraft and the aircraft vertical/horizontal speeds.

Time as well is important and here resides one of the main differences between ACAS II and ACAS X. In the ACAS II logic, the central time variable is t_{cpa} , the time to Closest Point of Approach (CPA). The CPA is the point in space where the slant range between both aircraft is minimum (Figure 2a). Computing t_{cpa} accurately may not be possible because the trajectory of the threat is unknown and the constant velocity vector hypothesis may not hold [9]. Instead, approximations of t_{cpa} can be computed. A first approximation, called τ_{cpa} , is equal to t_{cpa} for head on encounters (worst case situation):

$$\tau = -\frac{r}{\dot{r}},$$

where r is the slant range between both aircraft and \dot{r} is the closure rate. However, the τ metric has two issues: if the threat has a very low closure rate, then τ will be high even if the range between aircraft is small; if the closure rate is very large then τ will be small, even if the threat is far and not on collision course. To overcome these limitations, a modified version of τ , called τ_{mod} , has been defined as follows:

$$\tau_{mod} = \begin{cases} \frac{DMOD^2 - r^2}{r\dot{r}} & , \text{ if } r > DMOD \\ 0 & , \text{ if } r \leq DMOD. \end{cases}$$

where DMOD is a horizontal distance threshold under which to alert regardless of the range and closure rate. The τ_{mod} metric also has limitations. If aircraft are converging with high closure rate but large miss distance, τ_{mod} will be small [10]. A

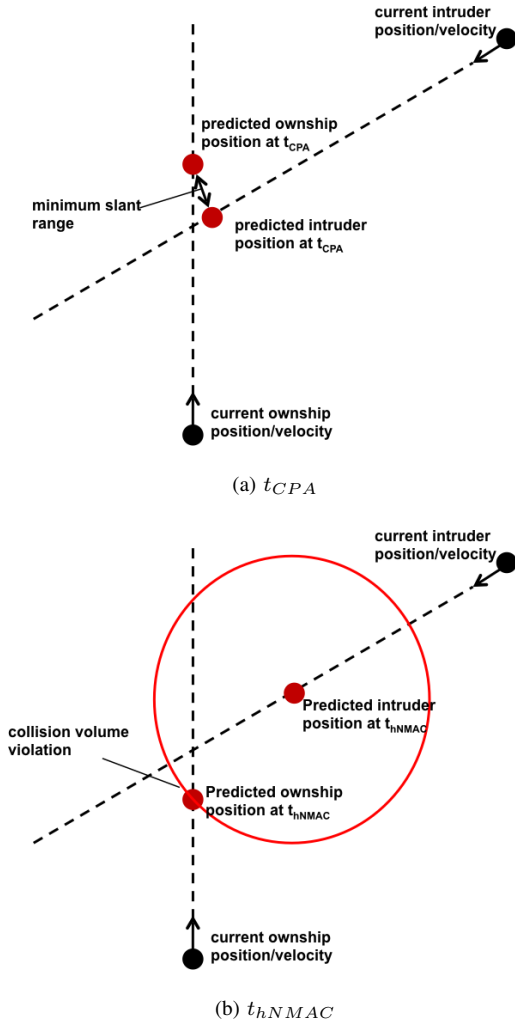


Fig. 2: The τ variables used in ACAS II and ACAS X are different. The first one estimates t_{CPA} , the time to the closest point of approach (minimum distance). The second one estimates t_{hNMAC} , the time before entering the collision volume.

whole family of time metrics, including τ and τ_{mod} have been defined in [9], but they are of limited interest in the context of this work as they are not used in the ACAS family.

In the ACAS X logics, the considered time is t_{hNMAC} , the time to horizontal NMAC. It is different from t_{CPA} as it measures the remaining time before the intruder is within 500ft horizontally (Figure 2b). A distribution over τ_{hNMAC} is computed using Dynamic Programming (DP) from the relative horizontal range and range rate, difference between own horizontal speed vector and bearing of the intruder. The slow horizontal closure rate case is handled by using vertical distance and speed. More details about τ_{hNMAC} computation are provided in Section IV.

With two initial hypothesis, an encounter can be described thanks to a limited set of distance, speed and time metrics. In the following we will present the metrics used in ACAS II

and ACAS X, and explain how they are used to compute the collision avoidance threshold depending on the method.

III. AN OVERVIEW OF ACAS II 7.1

Before diving into the subject of ACAS X, and to ease comparison between ACAS II and ACAS X, this section gives an overview of the logics behind ACAS II. This description is based on ACAS II version 7.1 [11].

A. The ACAS sensor

ACAS II only supports a single dedicated cooperative sensor system. It is composed of two antennas for active transponder interrogation. Though two directional antennas is the preferred setup, one directional and one omnidirectional antennas are sufficient for the system to work. By actively interrogating surrounding transponders, the system is able to calculate the surrounding aircraft range, closure rate and bearing. The altitude is obtained from the transponder answer. This system allows interrogation with a range varying between 22NM and 86NM [12], [13], [14], a field of regard of $\pm 180^\circ$ horizontally and $\pm 22^\circ$ vertically [12]. Precision in altitude depends on the intruder transponder type (mode S or C). Bearing error is about 5° and can go up to 30° [15]. The measurements are done with a hybrid surveillance scheme so the update rate varies from 0.1Hz to 1Hz depending on the distance to the threat.

B. Threat Detection

ACAS II is a threshold based approach using deterministic encounter models. It classifies an aircraft based on a range and an altitude test. At any given time, the encounter state is described by three parameters:

- τ_{mod} , an estimation of the time before minimum separation, in seconds.
- h , the altitude of the threat relative to the ownship, in feet.
- s_{RA} , state of the resolution advisory, i.e. what is the current resolution.

Bearing information is not necessary, but it may be used to suppress nuisance alerts when the horizontal miss distance is large.

Based on the current state, if an intruder aircraft is projected to be under a range threshold and under a certain horizontal distance within time τ_{mod} , the range test is passed. Then an altitude test checks if the intruder is under an altitude threshold. The thresholds used in both range and altitude tests depend on a parameter called the sensitivity level, which in turns depends on the aircraft altitude. If both tests pass, the intruder is declared a threat and action needs to be taken. Which action to take is decided during the threat resolution step.

C. Threat Resolution

Once a threat has been detected and declared, the logic computes a resolution in the vertical plane (maintain, climb, descend or level off) with an associated strength (from 500 up to 2500 ft/min). This is a two steps process, first the

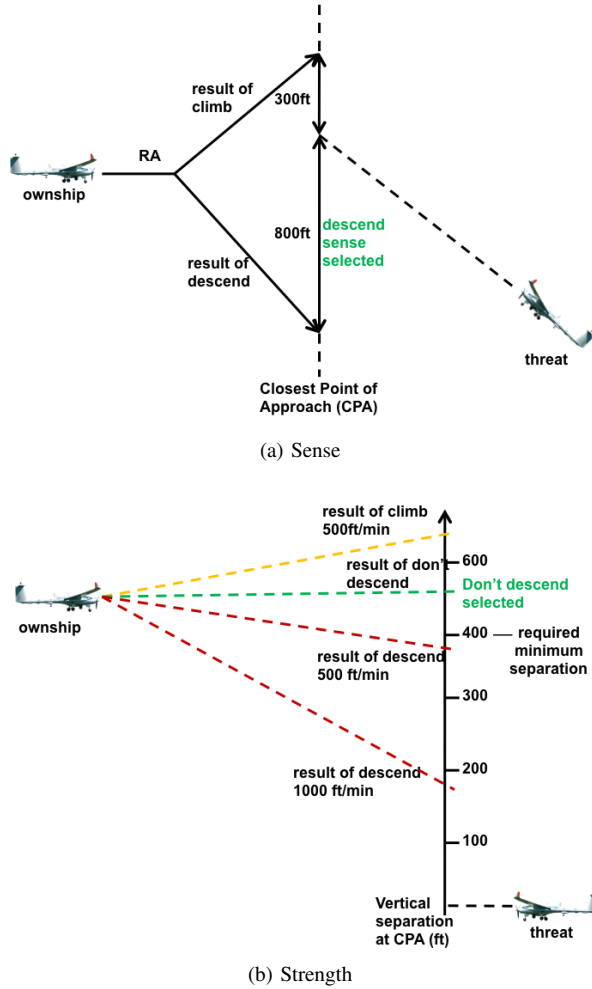


Fig. 3: Sense and strength selection process for ACAS II.

logic selects the sense which provides the greatest vertical distance at CPA (Figure 3a); then strength is selected to provide the least disruptive path while respecting a minimum separation altitude with the threat (Figure 3b). A maneuver with sense and strength is called a resolution advisory (RA) and is communicated to the pilot.

If the threat aircraft is equipped with an ACAS II, coordinated avoidance is possible. Aircraft decide on a master and slave, depending on their 24-bit ICAO address. The one with the highest address is the master and decides on a sense, the slave aircraft declares a RA in the opposite sense. For the sake of brevity, reversal and strengthening logics are not considered in this paper.

D. Limitations

From an aeronautical point of view, ACAS II has some limitations: unnecessary RA, specific encounter geometries, costly and long update cycles, etc. From a UAS point of view, the problems are different: non-cooperative traffic, resolution strength, assumptions about pilot, non-probabilistic world:

- ACAS II is not able to use the input from other sensors than the ACAS cooperative sensor. This is particularly limiting as UAS need to sense non-cooperative traffic.
- resolution's strength are tailored for large airliner performances. A UAS may not be able to follow the strongest resolutions and how much the UAS can do changes depending on the type of UAS.
- ACAS II specifications make assumptions about the pilot response time, e.g. pilot answers after 5s, and acceptable acceleration, e.g. 0.25g, which may be different for a UAS.
- modifying ACAS II to fit UAS performances would be time consuming and expensive.
- ACAS II uses a deterministic representation of the world so it cannot cope with unexpected pilot reactions, surveillance source errors or unexpected aircraft dynamics.

These limitations make ACAS II at best a temporary choice for UASs' collision avoidance. Various replacements have been proposed like JADEM (NASA) [16], MuSICA (AFRL) [17], MIDCAS (SESAR) [18] and ACAS Xu (FAA) [19]. Though no official successor has been chosen at the international level yet, ACAS Xu appears as a likely candidate.

IV. AN INTRODUCTION TO ACAS X BASELINE

The definition of ACAS Xu by expert committees, WG-75 (EU) and RTCA SC-147 (US), is just beginning but the foundations for the ACAS X family have already been laid during ACAS Xa development [6]. This section introduces these foundations. Unlike ACAS II, ACAS X is an optimization based approach relying on probabilistic models.

A. Sensors

A major benefit of ACAS X is its capacity to use information from different surveillance sources. The idea is to differentiate the sensing part from the avoidance logics and to have a plug-and-play architecture capable of handling different sensors. A model of the sensor is needed to take into account measurement uncertainty. For example, ACAS X is able to combine ADS-B data with ACAS sensor measurements, which is of particular interest as ADS-B is projected to be mandatory for large aircraft by 2020.

The existence of different surveillance sources calls for a fusion process where a track created by a given source is associated with those from other sources when they correspond to the same intruder. This process has been left unspecified for industrials to differentiate themselves.

The main requirement for the suite of sensors is to provide the slant range, bearing and altitude of intruders as these are the parameters needed for state estimation. Possible requirements in terms of performances and sensor types are discussed in Section VI.

B. State and Action Spaces

The ACAS X logic models the conflicts and their evolution through a combination of states and actions. ACAS II

describes the state of the system with three parameters, in ACAS X the state is described with five variables:

- τ_{hNMAC} , time to potential NMAC horizontally.
- h , altitude of the intruder relative to the ownship.
- s_{RA} , state of the resolution advisory, i.e. what is the current resolution followed by the ownship and how much time remaining before pilot respond to the resolution.
- h_0 , vertical rate of the ownship.
- h_1 , vertical rate of the intruder.

Unlike its predecessor, ACAS X handles uncertainty and represents the current state variables as probability distributions. However, the dynamic programming method used in ACAS X requires the state space to be discretized. For each variable a minimum, a maximum and a step need to be chosen keeping in mind that the resulting number of states needs to be computationally acceptable. As an example, the discretization used in [20] yields 15 millions different states.

Regarding actions, they correspond to the actual resolutions that ACAS X can issue, e.g. for ACAS Xa available actions include *Climb*, *Do Not Climb*, *Maintain Climb*, *Strengthen Climb*, *Reverse to Climb*. Each action has a minimum and maximum speed associated, an acceleration and a list of state resolutions from which they can be accessed, e.g. a *Strengthen Climb* can only be issued if the current state of resolution is a *Climb*, *Reverse to Climb* or *Maintain Climb*. Example state and action space tables for ACAS Xa can be found in [20].

C. State Estimation

At a given time, the state is determined by combining offline data (sensor models and ownship/intruder dynamic model) with online data (sensor measurements and the previous state variables), see Figure 4. The sensor model allows estimating the sensor error. It is combined with sensor measurements to obtain a probabilistic distribution of the measured variables. The aircraft dynamic model represents sequences of acceleration which influence the aircraft speed and consequently positions over time.

The state estimation is done in a three steps process. First, intruder altitude and vertical rate are estimated from sensor measurements, intruder dynamic model and previously observed altitude and vertical rate. Ownship altitude and vertical rate are assumed to be errorless. This allows computing h , h_0 and h_1 . Second, ownship discretizes (own altitude, heading) and sensor measurements (intruder slant range, bearing, altitude) are used to estimate a representation of the encounter with three parameters: r , the horizontal range, r_v , the horizontal speed, and θ_v , the difference in direction between own horizontal speed and intruder bearing. These parameters are used to determine τ_{hNMAC} by looking up an optimized table created offline. For the low r_v case (slow closure), a different table based on vertical values (h , h_0 and h_1) is looked up. In both cases multilinear interpolation allows retrieving the value of τ_{hNMAC} from the table's discrete values. Third, s_{RA} is estimated by combining the ownship dynamic model, the previously observed ownship vertical rate and resolution state.

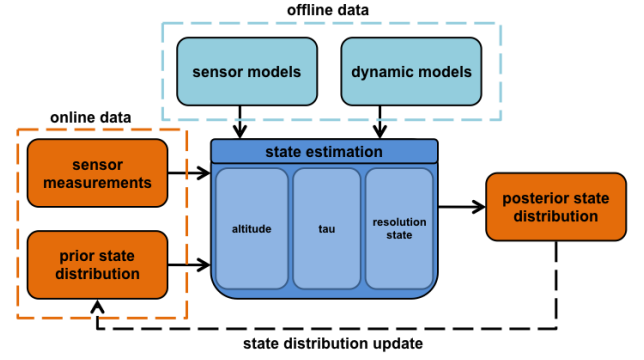


Fig. 4: The state estimation is a recursive three steps process based on offline and online data.

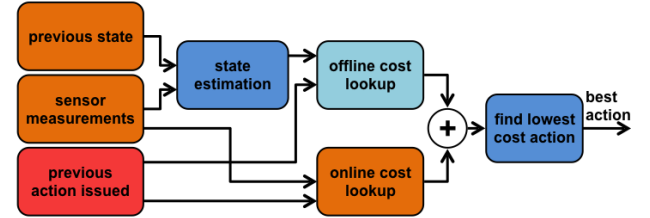


Fig. 5: The cost of actions is computed from online and offline costs. The minimum cost action is the best action.

The distributions provided by each of these sources are combined by a cartesian product to provide a joint distribution over the state variables [21]. This approach takes into account past states when estimating the current state. The estimated state can then be used to find the best action to take based on offline and online costs.

D. Threat Detection and Resolution

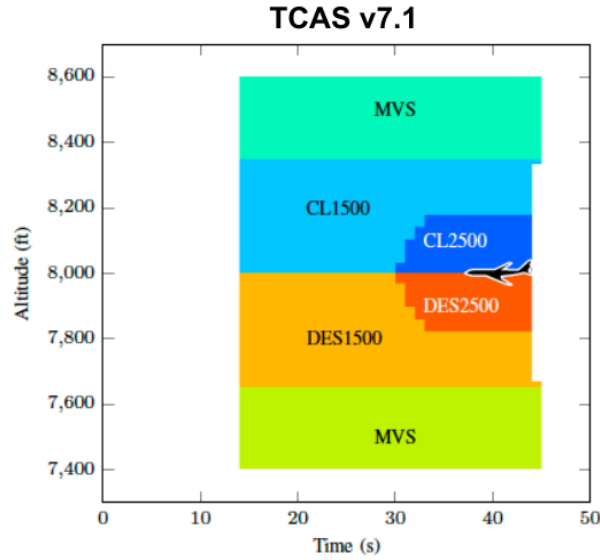
For any action a and state s , executing a from s is assigned a cost. The cost can depend on the action only (Issue CoC), on the state only (NMAC) or on both (corrective advisories when relative altitude greater than 500ft) [20]. Because these costs depend on a finite number of states and actions, they can be optimized offline and the cost for actions from each state can be stored in a table.

However, some operational situations cannot be described by the state variables. In these cases, costs are attributed to actions that lead to these situations. Because they don't depend on a finite number of states, the corresponding values cannot be computed offline, so they are called online costs.

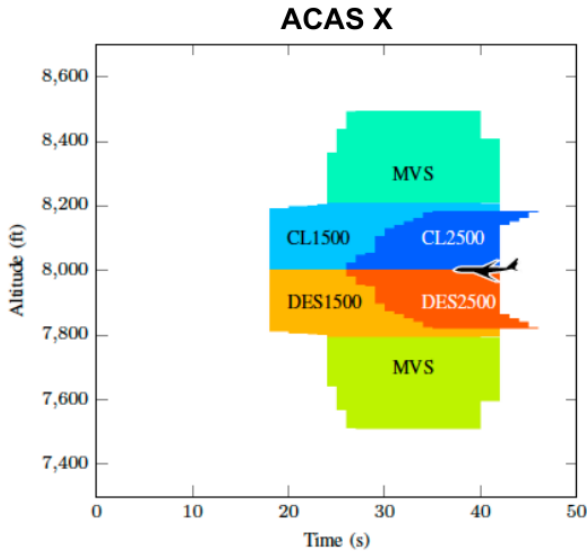
As illustrated in Figure 5, from a given state, offline and online costs are added and the lowest cost action is selected.

E. Optimized tables computation

The cost tables used through ACAS X are created by iterative optimization, simulation and adjustment. Quality criterion are decided beforehand, costs are optimized with respect with these criterion and the resulting ACAS X is tested in simulation. Simulation results are analyzed and costs are



(a) ACAS II



(b) ACAS X

Fig. 6: ACAS II vs ACAS X alerting threshold shapes (images from [22]).

changed to improve the results, more information can be found in [22]. At the time of writing, more than fifteen optimization iterations have been performed and work is still in progress. The optimization process is of limited interest to users as the optimization effort is going to be supported by the regulatory bodies which will provide certified optimized tables.

Figure 6 provides a practical illustration of the difference between ACAS II and ACAS X. Being a threshold based method, ACAS II action zone follows a simple geometrical shape (Figure 6a). In the case of ACAS X, the existence of multiple cost influencing the decision creates a complex, but smaller, action zone shape (Figure 6b).

This section drew a parallel between the logics of ACAS

II and ACAS X baseline. The main differences reside in the modeling of the encounter and the probabilistic approach adopted in ACAS X. With the baseline clear in mind, let's move on to the features required to give an ACAS to the UASs.

V. THE SPECIFICITIES OF ACAS Xu

ACAS X baseline is unlikely to change, however the development of ACAS Xu is just beginning, so part of its description may evolve in the coming years. This section aims at providing the current state of the ACAS Xu standard description [19]. Note that ACAS Xu is currently being developed for large UAS only (MALE, HALE), extension to small UAS is mentioned in Section VII.

A. Non-cooperative Surveillance Sources

The ACAS X baseline relies exclusively on cooperative surveillance sources (ACAS sensor, ADS-B in, etc.). However, UASs will need to sense non-cooperative traffic, so they will require non-cooperative sensors. Tracks from cooperative and non-cooperative sensors will be merged independently. It is important to note that when merging non-cooperative sensor data, different weights should be applied to different pieces of information. For example, since a radar has precise range but limited bearing accuracy [12], range should have a higher weight than bearing. Then the two merged tracks will be correlated to determine if they belong to the same intruder. If cooperative and non-cooperative tracks are available for a given intruder, only the cooperative track will be passed on to the threat resolution module. The reason for this is that cooperative sensors are deemed more precise and bring more information than non-cooperative ones [19].

B. Tailored Threat Logic

Aircraft flying in controlled airspace have similar performances, so a small set of optimized tables is sufficient for ACAS Xa. However, as noted by Dalamagkidis in [23], UASs come in a large variety with a broad spectrum of performances. In order to have suitable collision avoidance resolutions they will need optimized tables tailored to their capabilities. To address this problem, it will be possible to tune the ACAS Xu algorithms to support a large range of performances.

C. Vertical and Horizontal Resolutions

For manned aviation, ACAS II/Xa systems provide resolutions in the vertical plan only. However, for different reasons, detailed hereafter, UAS will need both vertical and horizontal resolutions. Historically, vertical avoidance has been preferred over horizontal avoidance. Indeed, bearing measurements from the ACAS sensors are inaccurate and it can be hard to judge if the threat is on the right or on the left. On the contrary altitude provided by the transponder is precise. Plus, the collision volume around an aircraft is 100ft vertically and 500ft horizontally, so the avoidance distance is smaller in the vertical plan. Finally, for an aircraft, climbing is faster than banking. For UASs most of these reasons are still valid, that

is why vertical avoidance is preferred. However, the speed ratio between a large aircraft and a UAS makes the horizontal avoidance safer in some situation. In particular, for a non cooperative intruder, if the sensors are not able to provide an accurate altitude estimation the horizontal resolutions is the safest option. As a general rule, ACAS Xu will try to solve conflicts with cooperative traffic using vertical motions and with non-cooperative traffic using horizontal motions.

In practice, integrating variables specific to horizontal threat detection and resolution in the optimization process would increase the state space beyond computational limits. Tackling this problem requires a second optimized table dedicated to horizontal maneuvers. In order to choose between vertical, horizontal or blended maneuver, ACAS Xu will rely on a to-be-defined module named Nucleus.

D. Coordination

Coordination is an important part of all the ACAS logics. At the time ACAS Xu enters service, there will be three types of collision avoidance systems in operation: ACAS II, ACAS Xa and ACAS Xu. Coordination should be different depending on the encounter:

- ACAS Xu/ACAS Xu, the coordination will be similar to the coordination protocole used when two ACAS II meet (active coordination).
- ACAS Xu/ACAS Xa, the unmanned system will present itself as a ACAS Xu so the manned aircraft knows it is dealing with a UAS. The ACAS Xu will be in the slave position (modified coordination).
- ACAS Xu/ACAS II, a resolution can be generated but no maneuver will be made until the manned aircraft generates an RA. The idea is to not alert manned aircraft unless they already are. So the UAS will wait for the manned aircraft to issue a RA before trying to coordinate a maneuver. The ACAS Xu will be in the slave position for the encounter (responsive coordination).

Though the current idea is to always have the UAS take the slave position, recent work from Londner [24] showed that giving the slave role to the UAS is not always the safest option. Based on risk ratios he showed that in many instances, collision risk is lower when the UAS is the master. This is due to the fact that manned aircraft don't always answer to RA resolution and can even maneuver contrary to the RA.

It is interesting to note that, according to Asmar et al. [25], ACAS X cannot perform implicit coordination with itself. Indeed, the fact that two altitude rate and different dynamic models are used breaks the symmetry of the offline table. In clear, one aircraft alerting does not mean that the other will alert so implicit coordination cannot be done as it is not sure if the second aircraft will maneuver.

E. Automatic RA

The CA function is a critical system which needs to be functional at all time. For this, it needs to be independent from the communication with the ground station so that even if the command and control link is lost, CA is still active. The

procedure would be to transmit any RA to the remote pilot, give some time to the pilot to possibly override the maneuver and if no override is issued proceed to the avoidance. If for some reason the link with the remote pilot is lost, the UAS will follow the RA. After avoidance, the UAS will follow a post RA procedure to go back to its mission safely unless the remote pilot takes control. Definition of such procedures is still a research topic with works, like the ODREA project [26], already providing some insight.

VI. SURVEILLANCE SOURCES REQUIREMENTS

It is likely that regulation will intervene in the threat detection and resolution by providing standardised and certified cost tables. For the sensing part, it is still unclear what will be the requirements. It is likely that the software parts will remain unregulated to allow room for competition between industrials. However, the sensors are likely to be subject to restrictions. This may concern their performances and the overall performances of the UAS's sensor suite. It is important to note that UASs will need to ensure CA as well as TrA (i.e. self-separation), so sensor's capabilities should cover both needs. The rest of this section proposes likely performances for sensors as well as a likely sensor suite required for UAS to fly in the NAS.

A. Performances

According to Zeitlin [27], performances of a sensor can be described by five variables: *field of regard* (FOR), *range*, *update rate*, *accuracy*, *integrity*; we add a sixth important variable: *safety*. In a study on sensor's range and field of regard, Park et al. [28] showed that with a horizontal FOR of $\pm 90^\circ$ horizontally, $\pm 20^\circ$ vertically and a range of 10NM, more than 90% of the aircraft could be detected 60 seconds before loss of separation. They also mention that a sensor with high vertical FOR, about 65° , and short range, about 2NM, could detect all intruders at the latest when loss of separation occurs. Concerning update rate, the hybrid surveillance scheme used in the ACAS sensor provides satisfactory results, so similar update rates seem to be acceptable. For integrity and accuracy, there are no known requirements and few existing system to compare to. Finally, for safety, some sensors are easier to disable than other. For example, a camera can be blinded by lasers just as a human pilot. However, a air-to-air radar is hard to foul unless the intruder wears particular shields.

No single sensor is perfect in every of these six aspects. For this reason the solution to the sensing problem is a suite of sensors.

B. Sensors Suite

As noted by Euteneur et al. [29], regulation will make some sensors mandatory by 2020. They cite eight high level requirements that will make ADS-B in/out, ACAS and air-to-air radar required for UAS by 2020. They also note that the radar poor bearing accuracy needs to be compensated by an additional sensor. Because they are low cost and easy to setup, the current solution is to use electro-optics (EO) and infrared

(IR) sensors for this role. So a UAS suite of sensors for detect and avoid would be composed of ADS-B in/out, ACAS, air-to-air radar and EO/IR. We believe this composition could change if LIDAR technologies reach a higher level of maturity.

Based on further research work, regulatory bodies will be able to provide more detailed requirements for equipment and hopefully allow the industry to advance in the right direction.

VII. THE CHALLENGES AHEAD

The definition of ACAS Xu progresses steadily. However, challenges still need to be solved before ACAS Xu can be finalized. The current effort concerns large UAS, more work needs to be done for small UAS to use this DAA solution.

A. For large UAS

1) *Sensors Miniaturization*: One of the general conclusion of the RPAS integration project MIDCAS [18] is that sensors miniaturization is needed. Currently, sensors fitted on UASs are general purpose sensors or aviation sensors. Equipement specific for UAS is needed to increase DAA systems performances. Some sensors already available at small size need maturation to provide useful results.

2) *Bearingless Active Surveillance*: The use of directional antennas in the ACAS sensor significantly increases its overall SWaP. For small UASs to benefit of ACAS Xu, it could be decided to allow a bearingless mode. Indeed, bearing is not essential for ACAS Xu or ACAS II/X. Even without bearing information, the information retrieved through the sensor may be sufficient to validate ADS-B data. Though more research is needed in this area, it appears as the best option to allow ACAS Xu for the lighter UAS among those considered.

3) *Prediction Integration*: Though ACAS Xu is a probabilistic CA method that takes into account the different possible trajectories of an intruder, it is limited to a model based on accelerations. It is not possible to include explicit trajectories. Though, this could prove useful in situations where the procedure allows knowing the futur trajectory of an intruder or if they have explicit intents. The idea of explicit intent is to broadcast ownship's trajectory to surrounding aircraft [30] to allow precise computation of t_{cpa} by giving trajectory information up to some time horizon.

B. For small UAS

1) *Table Discretization*: As seen earlier, for the optimization table, states values need to be discretized. However, the discretization depends on the UAS capabilities. In the particular case of small drones flying in Very Low Levels (<500ft), they are at risk of flying in dense flight zone with numerous other UAS. In order to have more avoidance options, a fine discretization scheme is required, but this imply a large number of states. On the contrary, the capabilities of small UAS are limited, so the maximum and minimum for each state variable are likely to be smaller. So there is an appropriate number of states to find to provide numerous options to the UAS, within the limits of its capabilities.

2) *Threat Dynamic Model*: The dynamic model of aircraft is central to ACAS Xu optimized logic [21]. For an airliner flying among airliners, there is no problem. For a UAS flying among different performances UAS, it requires having different dynamic models and being able to associate a dynamic model to the intruder before state can be estimated. The problem may appear when integrating UAS in controlled airspace and will be even stronger for very low level UAS, with the differences between rotary wings and fixed wings. A dynamic model could be associated by visually categorizing the intruder as a known type of aircraft or by estimating the dynamic model from surveillance measures if it has been under surveillance for a long enough time. According to [21], prior expert information could help to complete these measurements.

VIII. CONCLUSION

The logics at the heart of ACAS II and ACAS X are very different, from a deterministic method to a decision theoretic one. Understanding these differences is complicated by the fact that some concepts that look similar in both methods, like the τ variable, are in fact distinct. This work aimed at providing a clear explanations of the ACAS X baseline while highlighting the differences with ACAS II. The main one being that ACAS X is a probabilistic approach which can accept uncertainty in various aspects like sensors measurements or dynamic responses. Still, there are some concepts that were not mentioned, like the multi-threat management, reversal and strengthen logics, pilot reaction, which could fit in a complement to this introduction. For ACAS Xu, the main differences with ACAS X concern the handling of non-cooperative sensors and the existence of horizontal resolutions. These concepts have been described generally and more detail will be available has the ACAS Xu standardisation process goes forward.

The advent of ACAS Xu may provide the very much needed Detect And Avoid function for UASs but plenty of work is still needed. Especially since few tests have been performed in the European sky, and early results surprisingly showed no improvement, or worse, over ACAS II [31]. As the development of ACAS Xu progresses, the need for more explanatory works like this one will arise to keep the gap between future ACAS Xu users and ACAS Xu developers limited.

ACKNOWLEDGMENT

This work was supported by the ENGIE Ineo - Sagem RPAS Chair. The authors would like to thank Christian Aveneau for sharing his extended expertise on the matter, as well as Catherine Ronflé-Nadaud and Julien Farjon for the enlightening discussions.

REFERENCES

- [1] Association for Unmanned Vehicle Systems International (AUVSI). (2013) The economic impact of unmanned aircraft systems integration in the united states. [Online]. Available: <http://www.auvsi.org/auvsiresources/economicreport>

- [2] C. Snow. (2014) Five reasons the auvsi got its drone market forecast wrong. [Online]. Available: <http://www.suasnews.com/2014/06/five-reasons-the-auvsi-got-its-drone-market-forecast-wrong/>
- [3] La Socit Savante de l'Aeronautique et de l'Espace (A3F) and Academie de l'Air et de l'Espace (AAE), *Present and Futur of Civilian Drones*. AAE - A3F, 2015.
- [4] EUROCAE. (2016) Eurocae symposium & general assembly. [Online]. Available: <https://www.eurocae.net/events/symposium/>
- [5] ESRG. (2013) Roadmap for the integration of civil remotely-piloted aircraft systems into the european aviation system.
- [6] TCAS Program Office, "Concept of operation for the airborne collision avoidance system x," FAA, CONOPS, April 2013.
- [7] RTCA SC-147. (2014) Minimum operational performance standards for traffic alert and collision avoidance systems airborne equipment.
- [8] ICAO, *Manual On Remotely Piloted Aircraft Systems (RPAS)*. ICAO, 2015.
- [9] C. A. Munoz, A. Narkawicz, J. Chamberlain, M. Consiglio, and J. Upchurch, "A family of well-clear boundary models for the integration of uas in the nas," in *Proceedings of 14th AIAA Aviation Technology, Integration, and Operations Conference*, AIAA, 2014.
- [10] M. Johnson, E. R. Mueller, and C. Santiago, "Characteristics of a well clear definition and alerting criteria for encounters between uas and manned aircraft in class e airspace," in *Eleventh USA/Europe Air Traffic Management Research and Development Seminar (ATM2015)*, 2015.
- [11] Eurocontrol, *Airborne Collision Avoidance Systems (incorporating TCAS II version 7.0 & 7.1 and introduction to ACAS X)*. Eurocontrol, 2015.
- [12] M. Contarino. (2009) All weather sense and avoid for uass. [Online]. Available: <http://rcubedengineering.com/wp-content/uploads/2013/04/AWSAS-Report-to-Office-of-Naval-Research-for-R3-Engineering.pdf>
- [13] S. Hottman, K. Hansen, and M. Berry. (2009) Literature review on detect, sense, and avoid technology for unmanned aircraft systems. [Online]. Available: <https://trid.trb.org/view.aspx?id=912934>
- [14] X. Yu and Y. Zhang, "Sense and avoid technologies with applications to unmanned aircraft systems: Review and prospects," *Progress in Aerospace Sciences*, vol. 74, pp. 152 – 166, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0376042115000020>
- [15] EUROCONTROL. (EUROCONTROL) Acas bulletin n6: Incorrect use of the tcas traffic display. [Online]. Available: <https://www.eurocontrol.int/sites/default/files/publication/files/acas-bulletin-6-disclaimer.pdf>
- [16] R. C. Rorie and L. Fern, "The impact of integrated maneuver guidance information on uas pilots performing the detect and avoid task," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 59, no. 1. SAGE Publications, 2015, pp. 55–59.
- [17] R. H. Chen, A. Gevorkian, A. Fung, W.-Z. Chen, and V. Raska, "Multi-sensor data integration for autonomous sense and avoid," in *AIAA Infotech at Aerospace Technical Conference*, 2011.
- [18] MIDCAS, "Mid-air collision avoidance system (midcas)," 2015.
- [19] TCAS Program Office, "Concept of use for the airborne collision avoidance system (acas) xu," FAA, CONUSE, September 2015.
- [20] D. M. Asmar and M. J. Kochenderfer, "Optimized airborne collision avoidance in mixed equipage environments," Lincoln Laboratory, MIT Report, May 2013.
- [21] M. J. Kochenderfer and J. P. Chryssanthacopoulos, "Robust airborne collision avoidance through dynamic programming," Lincoln Laboratory, MIT Report, January 2011.
- [22] J. E. Holland, M. J. Kochenderfer, and W. A. Olson. (2013) Optimizing the next generation collision avoidance system for safe, suitable, and acceptable operational performance. [Online]. Available: http://www.atmseminar.org/seminarContent/seminar10/presentations/247-Holland_0126130538-PresentationPDF-5-24-13.pdf
- [23] K. P. Valavanis and G. J. Vachtsevanos, *Handbook of Unmanned Aerial Vehicles*. Springer Publishing Company, Incorporated, 2014.
- [24] E. H. Londner, "Collision avoidance system effectiveness on low performance unmanned aircraft," in *Proceedings of 16th AIAA Aviation Technology, Integration, and Operations Conference*, AIAA, 2016.
- [25] D. M. Asmar, "Airborne collision avoidance in mixed equipage environments," Ph.D. dissertation, Massachusetts Institute of Technology, 2013.
- [26] ODREA, "Operational demonstration of rpas in european airspace (odrea)," 2015.
- [27] P. Angelov, *Sense and Avoid in UAS, Research and Applications*. Wiley, 2012.
- [28] C. Park, S. M. Lee, and E. R. Mueller, "Investigating detect-and-avoid surveillance performance for unmanned aircraft systems," in *Proceedings of 14th AIAA Aviation Technology, Integration, and Operations Conference*, AIAA, vol. 2413, 2014.
- [29] E. Euteneur, G. Loegering, S. Krishnan, and J. Jewell, "Required surveillance sensors for daa," in *Digital Avionics Systems Conference (DASC), 2014 IEEE/AIAA 33rd*. IEEE, 2014, pp. 6D6–1.
- [30] E. Thomas and O. Bleeker, "Options for insertion of rpas into the air traffic system," in *2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC)*, Sept 2015, pp. 5B4–1–5B4–14.
- [31] EUROCONTROL. (2015) Hindsight: Safety nets. [Online]. Available: <https://www.eurocontrol.int/publications/hindsight-22-winter-2015>