



Faculty Publications

2015-4

Detect and Avoid for Small Unmanned Aircraft Systems using ADS-B

Timothy McLain

Mechanical Engineering Department, Brigham Young University, mclain@byu.edu

Laith R. Sahawneh

Mechanical Engineering Department, Brigham Young University

Matthew O. Duffield

Mechanical Engineering Department, Brigham Young University

Randall W. Beard

Department of Electrical Engineering, Brigham Young University

Follow this and additional works at: <https://scholarsarchive.byu.edu/facpub>



Part of the [Mechanical Engineering Commons](#)

Original Publication Citation

Sahawneh, L., Duffield, M., Beard, R., and McLain, T. Detect and Avoid for Small Unmanned Aircraft Systems using ADS-B, *Air Traffic Control Quarterly*, vol. 23, no. 2-3, pp. 203-240, April 2015.

BYU ScholarsArchive Citation

McLain, Timothy; Sahawneh, Laith R.; Duffield, Matthew O.; and Beard, Randall W., "Detect and Avoid for Small Unmanned Aircraft Systems using ADS-B" (2015). *Faculty Publications*. 1895.
<https://scholarsarchive.byu.edu/facpub/1895>

This Peer-Reviewed Article is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Faculty Publications by an authorized administrator of BYU ScholarsArchive. For more information, please contact ellen_amatangelo@byu.edu.

Detect and Avoid for Small Unmanned Aircraft Systems using ADS-B

Laith R. Sahawneh, Matthew O. Duffield, Randal W. Beard, Timothy W. McLain
Brigham Young University

With the increasing demand to integrate unmanned aircraft systems (UAS) into the National Airspace System (NAS), new procedures and technologies are necessary to ensure safe airspace operations and minimize the impact of UAS on current airspace users. Currently, small UAS face limitations on their utilization in civil airspace because they do not have the ability to detect and avoid other aircraft. In this article, we will present a framework that consists of an Automatic Dependent Surveillance-Broadcast (ADS-B)-based sensor, track estimator, conflict/collision detection, and resolution that mitigates collision risk. ADS-B offers long range, omni-directional intruder detection with comparatively few size, weight, power, and cost demands. The proposed conflict/collision detection and planning algorithms for conflict/collision resolution are designed in the local level frame, which is unrolled, unpitched body frame where the ownship is stationary at the center of the map. The path planning method is designed to be multi-resolutional at increasing distance from the ownship to account for both self-separation and collision avoidance thresholds. We demonstrate and validate this approach using simulated ADS-B measurements.

INTRODUCTION

The number of applications of unmanned aircraft systems (UAS) is growing at a significant pace. Consequently the need for UAS in the National Airspace System is compounding at a similar rate. Governmental institutions are increasingly adopting UAS to perform tasks such as weather research, search and rescue, wildlife surveillance, law enforcement, wildfire monitoring, and military training. A report compiled by the US Department of Transportation on UAS service demands estimates that by the year 2035 there will be approximately 70,000 UAS operated by federal, state, and local departments and agencies (*Unmanned Aircraft System (UAS) Service Demand 2015-2035*, 2013). In the private sector, the ever growing number of UAS applications includes a wide variety of industries and tasks ranging from smoke stack inspection to cinematography to crop dusting to oil exploration to news and traffic reporting. The demand for UAS operations is manifest by the approximately six hundred petitions as of March 2015 to allow UAS operations under Section 333 of the FAA Modernization and Reform Act of 2012 (*FAA Modernization and Reform Act of 2012*, 2014).

While UAS operations have increased as a result of the Section 333 exemptions approved since September of 2014, the overall realized benefit of UAS operations is still a small fraction of the demand. Additionally Section 333 exemptions are not a long-term solution to supporting UAS in the National Airspace System. In laying the foundation for a long-term solution for UAS in the NAS, the Federal Aviation Administration (FAA) has mandated that UAS be capable of an equivalent level of safety (ELOS) to the see-and-avoid mandate for manned aircraft (Hottman, Hansen, & Berry, 2009; Federal Aviation Administration, 2015). As a result, similar to a pilot's ability to visually scan the surrounding airspace for possible intruding aircraft and take action to avoid a collision, a UAS must be capable of monitoring and avoiding other manned or unmanned aircraft with which it may collide. This detect-and-avoid (DAA) mandate is the capability of a UAS to remain well clear and avoid collisions with other air traffic (George, 2009). It is desirable that the DAA system should include both self-separation and collision avoidance functions. The self-separation function is responsible to maintain the well clear distance by maneuvering the UAS within a sufficient time to prevent activation of a collision avoidance maneuver. On the other hand, the collision avoidance function should act within a relatively short time frame to maneuver the UAS to prevent an intruder from penetrating the collision volume. The collision avoidance function is engaged when all other modes of separation fail to prevent an imminent collision. It is the last resort effort to steer the UAS onto a safe course. Ultimately DAA capability will provide UAS an equivalent level of safety to the current manned aircraft procedures.

In general, the DAA functionality can be broken into three sub-functions: detect and track, conflict/collision detection, and avoidance. The main role of the first sub-function is to detect any intruders and track the motion of the detected object. Not every aircraft that is observed by the sensing system, however, presents a conflict or collision threat. Therefore the conflict/collision detection system determines whether or not an approaching intruder aircraft is on a conflict/collision course. The term *conflict* is associated with self-separation and it usually implies an event where two aircraft come within 5 to 10 nautical miles over time horizons on the order of minutes (Paielli & Erzberger, 1997; Hu, Lygeros, Prandini, & Sastry, 1999). On the other hand, the term *collision* detection and avoidance is used for close proximity encounters over time horizons on the order of tens of seconds (Angelov, 2012).

For small UAS weighing less than 55 pounds, the algorithms and hardware necessary for DAA make up a notable portion of the available size, weight, and power (SWaP) resources. Scaling traditional sensors down to small UAS sizes often requires compromises in range, accuracy, field of view, or processing speed. Such compromises reduce the overall capability of the DAA system and, consequently, decrease the assurance of self-separation/collision avoidance. Radar is one sensor that is widely used for air-to-air detection in manned aircraft. One of the primary strengths of radar is the ability to detect all objects regardless of cooperative sensor equipment or functionality. In applying radar to small UAS, SWaP constraints impose restrictions on the hardware that result in significant trade offs between radar range, bearing accuracy, and field of view. At a set transmit power, improving the range requires a narrower beam, which also improves the bearing accuracy. Narrowing the beam, however, reduces the field of view and consequently requires additional antennas or a method to steer the beam. Demonstrated hardware that falls within the SWaP limitations of small UAS is not currently suited to support a feasible set of range, bearing

accuracy, and field of view requirements (Mackie, Spencer, & Warnick, 2014). Optical sensors such as cameras are also candidate sensors for DAA on small UAS. Similar to radar, vision-based intruder detection methods do not require cooperative communication from intruders. Flight testing of visual methods has achieved intruder detection at 0.54 nmi from a small UAS (J. Lai, Ford, Mejias, Shea, & Walker, 2012). Ground-based testing has resulted in detection up 4.3 nmi (Dey, Geyer, Singh, & Digioia, 2009). The flight tested range of 0.54 nmi is promising, but not sufficient to provide enough avoidance time for high-speed intruders. Even with sufficient range, visual methods inherently have low range accuracy. Adverse weather conditions such as fog, clouds, precipitation, and sun glare can reduce overall visibility and significantly limit visual intruder detection. While recent developments have improved visual intruder detection, such methods are not yet suitable for DAA implementation on small UAS.

Automatic Dependent Surveillance-Broadcast (ADS-B) is a cooperative sensor that is a promising option for DAA on small UAS. It has been demonstrated in small UAS flight testing to have an omni-directional range of 20 nmi (Moody & Strain, 2009), and due to the fact that the cooperative information is shared over radio waves it is relatively unaffected by adverse weather conditions. An omni-directional antenna and low-power requirements for both transponder and receiver hardware contribute to the promising characteristics of ADS-B. Two drawbacks of ADS-B are its dependence on global positioning system (GPS) information and its fundamentally cooperative nature. While GPS coverage of the national airspace is very good, there are areas where GPS information can become degraded such as narrow valleys or urban canyons. Furthermore, the cooperative aspect of ADS-B requires widespread adoption of ADS-B technology to ensure detect-and-avoid reliability. While the Federal Aviation Administration does not yet require all aircraft to be equipped with ADS-B transponders, the 2020 mandate requiring all aircraft in A, B, C, and some E class airspace to equip with ADS-B (Federal Aviation Administration, 2010b) is a significant step.

There is also a considerable body of work on conflict/collision detection and risk assessment methods. A survey of 68 conflict detection and resolution methods is presented in (Kuchar & Yang, 2000), and a recent survey is conducted by (Albaker & Rahim, 2010; Angelov, 2012). These different methods can be classified under four fundamental approaches: deterministic or straight line, worst case, probabilistic, and flight plan sharing. Many of these methods stress the deterministic approach, where a single trajectory of an intruder is predicted using straight-line extrapolation. This is a reasonable approach when there is a perfect knowledge of the states of the detected intruder. In practice, however, the uncertainty free model could lead to erroneous prediction of the collision threat. While, many of these techniques are applicable for either conflict or collision detection, an appropriate scaling in design parameters, assumptions, and thresholds is required.

Similarly, airborne conflict/collision avoidance has gained considerable attention, and various methods and approaches have been suggested in the literature. Among the many collision avoidance algorithms, the local or reactive motion planning approaches are considered to be the most suitable approach for UAS collision avoidance. This is because a collision event occurs over a relatively short time horizon, which requires a planning method that promptly reacts to plan an avoidance maneuver using limited computation power. Moreover, reactive planning methods do

not require a priori knowledge of the environment. Reactive path planning is also well suited for dynamic environments where sensor information is uncertain and incomplete. The most common reactive path planning approaches are geometric-based guidance methods (Hyunjin, 2013; Rajnikant, Saunders, & Beard W., 2012) and potential field methods (Lam, Mulder, Van Paassen, Mulder, & Van Der Helm, 2009; Sahawneh, Randal W. Beard, & Bai, 2013). The sampling-based methods, like Probability Road Maps (PRM) (Kavraki, Svestka, Latombe, & Overmars, 1996) and Rapidly-exploring Random Trees (RRTs) (LaValle, 1998) have shown considerable success for obstacle avoidance and path planning, especially for ground robots. They often require significant computation time for replanning paths making them unsuitable for reactive avoidance. Recent extensions to the basic RRT algorithm, however show promising results for uncertain environments and nontrivial dynamics (Luders, Karaman, & How, 2013; Kothari & Postlethwaite, 2013; Luders, Karaman, Frazzoli, & How, 2010). Cell decomposition is another widely used path planning approach that partitions the free area of the configuration space into cells, which then are connected to generate a graph (Mirolo & Pagello, 1995). Generally, cell decomposition techniques are considered to be global path planners that require some a priori knowledge of the environment. A feasible path is found from the start to goal configuration by searching the connectivity graph using search algorithms, like A^* or Dijkstra’s algorithm (Dijkstra, 1959).

This article presents a complete detect-and-avoid solution for small unmanned aircraft including reliable intruder sensing, multi-target tracking and estimation, conflict/collision detection, and self-separation/collision avoidance. As shown in Figure 1, the ADS-B Out transmissions are received by a dual-link ADS-B In receiver. This receiver decodes the raw signal and passes it to the intruder tracker/estimator. In the estimator the intruder state measurements are processed to have a coherent set of units and then passed through a Kalman filter. After Kalman filtering, the intruder position and velocity estimates are projected forward in time to identify possible conflicts or collisions. If either a conflict or collision threat is detected, the intruder position and velocity estimates and an activation flag are passed into the self-separation/collision avoidance algorithm. Once either the conflict or collision level of the avoidance logic has been activated a new, conflict and collision-free path is generated. In the case of long-range intruders that pose a conflict risk the ownship takes less aggressive behavior due to the longer allowable reaction time. For short-range collision risks the ownship plans a much more aggressive action to quickly reduce the possibility of a collision. The ultimate output of the DAA system is a revised set of ownship waypoints that is free from conflict and collision risks. The system shown in Figure 1 and presented in this article is a complete DAA system for small UAS. It is viable for both fixed wing and multirotor aircraft, and could reasonably be extended for larger UAS outside of the small UAS definition.

The purpose of this article is to explore ADS-B as a sensor for detect-and-avoid on small unmanned aircraft and to demonstrate conflict/collision detection and self-separation/collision avoidance methods that take advantage of ADS-B characteristics. For the methods and simulations presented, we assume that the intruder aircraft are equipped with ADS-B Out, in other words the ability to transmit their cooperative information. The small UAS ownship is assumed to have ADS-B Out and dual-link ADS-B In. Thus it is capable of both transmitting its cooperative information and receiving the cooperative information from all other aircraft. Consequently the responsibility of conflict detection, self-separation assurance, collision detection, and collision

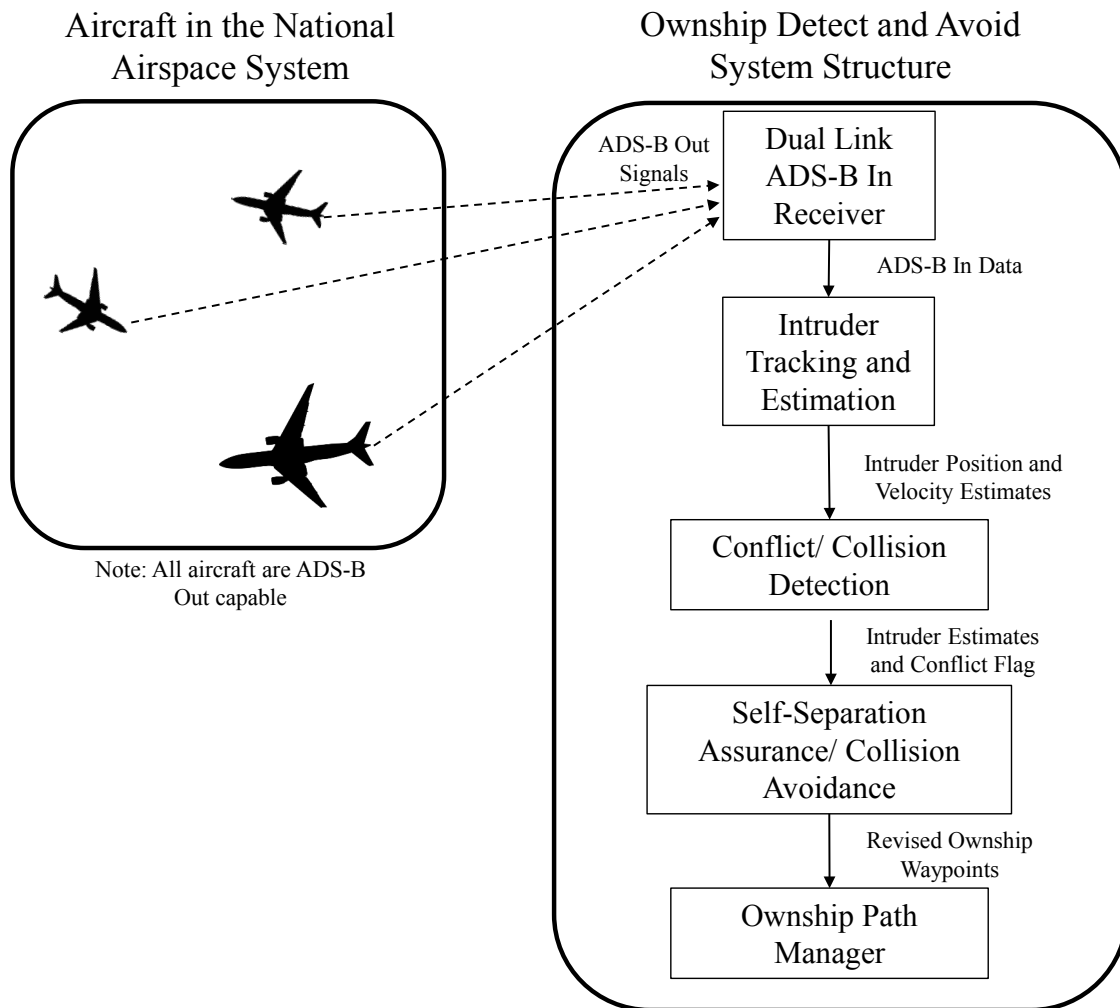


Figure 1: Proposed detect-and-avoid system structure diagram.

avoidance lies entirely on the small UAS ownship. Although these assumptions do not exactly match the requirements of the FAA 2020 mandate, they do represent a condition where full integration of UAS into the NAS would be possible. Thus in addition to presenting a DAA system for small UAS, we submit that complete ADS-B equipage requirements would meet the wide demand for significantly increased UAS operations in the NAS.

ADS-B ON SMALL UAS

ADS-B is rapidly becoming a major tool in the air traffic management system. In 2010 the FAA issued a final rule for the implementation of ADS-B on manned aircraft (Federal Aviation Administration, 2010b). This ruling mandated ADS-B Out in key parts of the NAS. The FAA Modernization and Reform Act of 2012 further directed the FAA to make plans for the adoption

of ADS-B In technology (*FAA Modernization and Reform Act of 2012*, 2014). As a result of the level of adoption and capability of ADS-B technology, ADS-B is an attractive sensor for detect and avoid efforts on UAS.

This section provides a description of ADS-B and the associated regulations as they relate to detect and avoid. A statistical characterization of ADS-B error and drop out is derived from the current FAA regulations. Further, we explore the capability of ADS-B as a DAA sensor by examining key characteristics and limitations of ADS-B.

Characteristics and Regulations of ADS-B

ADS-B is a cooperative sensor that supports the exchange of a wide variety of information over long ranges. Information that is typically exchanged includes aircraft state information, state error estimates, aircraft identifiers, and aircraft operating indicators. This exchange occurs approximately once per second (Cirillo, 2005). To exchange this information, two sets of hardware are necessary, ADS-B In and ADS-B Out. As the names suggest, ADS-B In allows for information to be received, and ADS-B Out supports the broadcasting of information. The hardware performing these two functions can be sold separately or as a single unit. In addition to the In or Out capability of ADS-B hardware, ADS-B transmissions can occur over two different frequencies, 1090 MHz and 978 MHz (Federal Aviation Administration, 2010b). The 1090 MHz Extended Squitter (ES) frequency is an internationally recognized ADS-B frequency. It is intended that this frequency be used for most commercial and high-performance aircraft. The 1090 MHz frequency is the same frequency used for current Mode S transmissions. The Extended Squitter designation indicates a message packet that is much longer than the standard Mode S packet. This allows for the transmission of much more information than what is exchanged via secondary surveillance radar (SSR). The 978 MHz Universal Access Transceiver (UAT) frequency is unique to United States airspace. It is primarily intended for private and low-altitude aircraft. ADS-B Out hardware is specific to one of these two frequencies. The airspace class in which an aircraft will operate dictates the required frequency. ADS-B In hardware also is specific to a particular frequency, but dual-link hardware that is capable of receiving transmissions on both frequencies is becoming increasingly available.

FAA regulations set forth in the 2010 Final Rule dictate most aspects of ADS-B operation. The message elements, airspace class, transmit power, latency, and error characteristics are all among the aspects of ADS-B that are regulated by the FAA. While these regulations do add complexity to the implementation and operation of an ADS-B system, they also provide a consistent basis upon which ADS-B can be evaluated for DAA on small UAS.

Message Element Requirements

The message elements exchanged by ADS-B transmissions provide a view of the transmitting aircraft's status. Table 1 shows a list of these elements that is arranged by functional category. The state elements transmitted are the latitude and longitude, barometric altitude, geometric altitude, and velocity. A certified position source must be used for latitude and longitude information. Typically a Satellite Based Augmentation System (SBAS) source is used. The barometric altitude

Table 1: Required set of message elements for ADS-B Out.

State Elements	Identification Elements	Error Elements	Other Elements
Latitude	Mode 3/A Transponder Code	NACp	Emitter Category
Longitude	Call Sign	NACv	Emergency Code
Barometric Altitude	IDENT	NIC	TCAS II equipped
Geometric Altitude	ICAO 24-bit address	SDA	TCAS II Advisory
Velocity	Length and Width	SIL	ADS-B In Equipped

is provided as the primary altitude as it is typically more accurate than the GPS-derived geometric altitude. The velocity transmitted is a ground reference velocity in knots and can be given as a combination of north and east velocity or speed and heading depending on whether the aircraft is on the ground or airborne (Federal Aviation Administration, 2015)(Radio Technical Commission for Aeronautics, 2009). If the aircraft is airborne, then the vertical velocity is given in feet per minute. On the other hand if the aircraft is on the ground, then the length and width of the aircraft is given instead of the vertical velocity.

The identification information provided by ADS-B permits simple identification of the transmitting aircraft. While a detailed explanation of each of the identification elements listed in Table 1 is beyond the scope of this paper, it is useful to note that each of these elements provides a unique identifier for the aircraft.

The message elements detailing the error in the state information are also shown in Table 1. Navigation Accuracy Category for Position (NACp) is a value that correlates to an Estimated Position Uncertainty (EPU) bound. The EPU bound used is defined as the “radius of a circle, centered on the reported position, such that the probability of the actual position being outside the circle is 0.05.” (Radio Technical Commission for Aeronautics, 2009) The FAA requires that the NACp must be greater than 8 which corresponds to an EPU < 303.8 ft (Radio Technical Commission for Aeronautics, 2009)(Federal Aviation Administration, 2015). The Navigation Accuracy Category for Velocity (NACv) is similar in that it is a value that corresponds to a error bound on the transmitted velocity. This bound is a 95% bound in that there is less than 0.05 probability that the error between the true velocity and the transmitted velocity exceeds the NACv bound. The FAA requires the NACv value to be greater than or equal to 1 which corresponds to the transmitted velocity error being less than 19.4 kn. Navigation Integrity Category (NIC) is a value that corresponds to an integrity containment radius, Rc. It signifies the maximum position error such that the probability that no integrity alert is indicated is less than the Source Integrity Level (SIL). In other words this radius is the value where there is an SIL probability that the measurement has been identified as a low integrity (possibly erroneous) measurement. This value must be greater than 7 which corresponds to Rc < 1215.2 ft. The SIL probability assumes no avionics faults, and the FAA mandates that SIL=3 which corresponds to probability $\leq 1 \times 10^{-7}$ per sample or per hour. The distinction between a per sample or per hour probability is made in an ADS-B message field known as SILsupp. To account for errors due to avionics faults, the System Design Assurance (SDA) is a value that corresponds to the “...probability of an ADS-B system fault causing false or

misleading information to be transmitted.” (Radio Technical Commission for Aeronautics, 2009) “The ADS-B system includes the ADS-B transmission equipment, ADS-B processing equipment, position source, and any other equipment that processes the position data transmitted by the ADS-B system.” (Radio Technical Commission for Aeronautics, 2009) This information includes latitude, longitude, velocity, accuracy metrics, or integrity metrics. The FAA mandates that the SDA value be 2 which corresponds to a probability $\leq 1 \times 10^{-5}$ per flight hour (Radio Technical Commission for Aeronautics, 2009)(Federal Aviation Administration, 2015). While both the SDA and the SIL report a probability of exceeding the NIC, it is important to note that the SIL assumes no avionics fault, but the SDA is the probability that an avionics fault is the cause of the reported error. Elements in the fourth column, labeled as Other Elements, provide information concerning the operational status of the aircraft. The first field specifies the emitter category of the transmitting aircraft. The emitter category indicates the type of aircraft and gives some indication of aircraft weight, size, and maneuverability. The emergency code is the second item in the fourth column. This code indicates if there is an emergency on-board the transmitting aircraft such as a medical emergency, minimum fuel, unlawful interference, or a downed aircraft. Such information is useful both to identify aircraft that need special attention from air traffic control services and for search and rescue efforts in a downed aircraft situation. The last three fields listed in column four of Table 1 indicate the equipage and activity of cooperative sensors. The third field indicates whether Traffic Collision Avoidance System (TCAS) is operable on the transmitting aircraft. Field four extends this and reports whether a traffic advisory or resolution advisory is in effect. The fifth field indicates whether the transmitting aircraft has ADS-B In capability.

Airspace and Power Requirements

The 2010 Final Rule on ADS-B mandated that by the year 2020 all aircraft in A, B, C, and some E class airspace be equipped with ADS-B Out. There is no mandate for ADS-B In. The FAA further mandated airspace where each of the two frequencies of ADS-B Out, 1090 MHz and 978 MHz, can be used. Class A airspace requires 1090 MHz. Where ADS-B is required below 18,000 ft, either 1090 MHz or 978 MHz is acceptable. Both B and C class airspace require ADS-B. ADS-B is also required within 30 nmi of a Class B airport reaching from the surface up to 10,000 ft mean sea level (MSL). Above B and C class airspace extending up to 10,000 ft MSL, ADS-B is required. E class airspace requires ADS-B from 10,000 ft MSL and above with the exception of the surface to 2,500 ft above ground level (AGL). In other words, if 0 ft AGL is above 10,000 ft MSL then there is a 2,500 ft region above ground level where ADS-B is not required. Finally ADS-B is required above 3,000 ft MSL over the Gulf of Mexico within 12 nmi of the coast of the United States (Federal Aviation Administration, 2015). Figure 2 summarizes the airspace requirements for ADS-B Out (Federal Aviation Administration, 2012).

The range of ADS-B transmissions is largely dependent on the transmit power of the ADS-B transponder. FAA regulations mandate different levels of transmit power for 1090 MHz and 978 MHz. For the 978 MHz frequency, there are three transmit power levels. Each level corresponds to a minimum transmit power and consequently a transmission range. The 1090 MHz frequency also has three levels which correspond to a minimum transmit power and range.

While transmit ranges vary as a result of frequency congestion, antenna differences, and other external factors, estimated, air-to-air ranges for the 978 MHz frequency extend from 10 nmi to 90 nmi and for the 1090 MHz estimated ranges extend from 10 nmi to 140 nmi (Radio Technical Commission for Aeronautics, 2009)(Radio Technical Commission for Aeronautics, 2011). Air-to-ground or ground-to-air transmissions have a much longer anticipated range as a result of more sensitive receivers and more powerful transponders that are available for ground-based equipment.

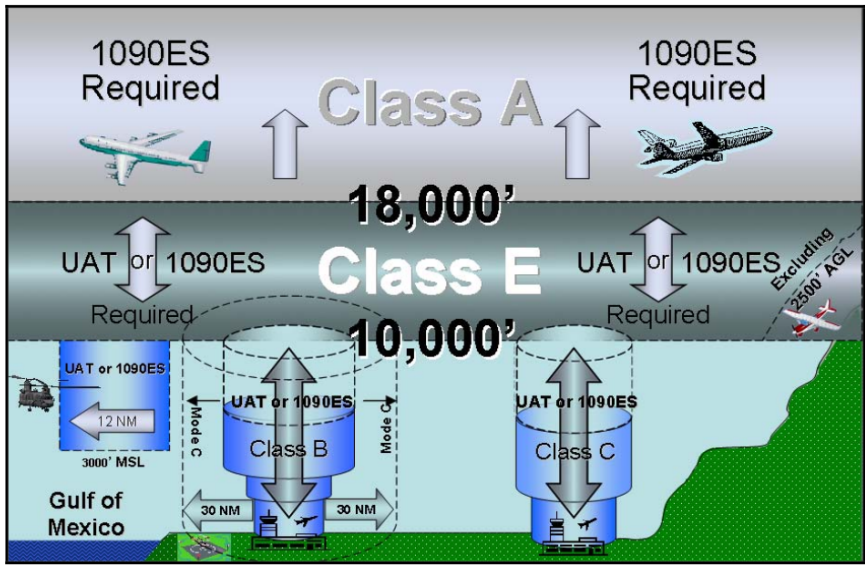


Figure 2: Diagram of airspace where ADS-B Out is required (Federal Aviation Administration, 2012).

Error Characterization

In addition to the error metrics outlined in Table 1, ADS-B is subject to several additional sources of error namely latency error, resolution error, and message success rate (MSR) error. These additional sources of error, along with those previously defined in Table 1, play a role in defining an error characterization of ADS-B.

Due to processing needs, data latency is inherent in the ADS-B system. This latency falls into two categories. Total latency is the time from measurement to transmission and must be less than 2.0 s. Of those 2.0 s, all but 0.6 s must be compensated for by the ownship. In compensating for latency the transmitting aircraft must “[extrapolate] the geometric position to the time of message transmission.” (Federal Aviation Administration, 2015) The uncompensated 0.6 s of the total latency is referred to as uncompensated latency (UL) (Federal Aviation Administration, 2015). It is the uncompensated latency that is the primary source of latency error.

Resolution error results from encoding state information into an ADS-B message where the information is represented by discrete bits. Table 2 shows the resolution limits for an ADS-B message (Radio Technical Commission for Aeronautics, 2009).

ADS-B regulations require that receivers are capable of supporting a given message success rate. For messages on the 978 MHz frequency this is 10%, and for messages on 1090 MHz, this is approximately 15%. These success rates imply that one out of every 10 or 3 out of every 20 messages is not received, thus resulting in message success rate error.

The NACp, NACv, NIC, SIL, SDA, latency error, resolution error, and MSR error provide a basis from which to derive an error characterization to model ADS-B. The error characterization

presented here will focus on state information and will use statistical methods to model the error of the actual measurements rather than the accuracy of individual bits. Given the NACp and NACv, the horizontal position and velocity can be modeled as a Rayleigh random process. From the Rayleigh process, the 95% bound on both the position and velocity error can be used to derive the variance for a Gauss-Markov process with zero-mean Gaussian noise for the north and east position and velocity (Mohleji & Wang, 2010)(Papoulis & Pillai, 2002). For derivation of the variation of a Gauss-Markov process in accordance with FAA requirements we use, NACp=303.8 ft and NACv=19.4 kn. Let X and Y each represent a Gauss-Markov process with zero-mean Gaussian noise such that $X \sim N(0, \sigma^2)$ and $Y \sim N(0, \sigma^2)$. R is a Rayleigh distributed variable such that $R \sim Rayleigh(\sigma)$ where σ is derived from the 95% NAC bound. Thus it can be shown that the variance is given by $\sigma^2 = \frac{-NAC^2}{2 \ln(0.05)}$. The NAC variance is considered generally for both NACp and NACv. Substituting values for NACp and NACv respectively results in $\sigma_x = \sigma_y = 124$ ft and $\sigma_{vx} = \sigma_{vy} = 8$ kn. From this analysis, it is determined that the horizontal north and east position error can be modeled as a zero-mean Gaussian distribution with a standard deviation of 124 ft and the north and east velocity can be modeled as a zero-mean Gaussian distribution with a standard deviation of 8 kn.

Correlation of errors in the position are accounted for by a Gauss-Markov model. Since the error correlation is a result of the correlation of GPS errors, the time constant used to simulate GPS errors is used to simulate ADS-B error correlation also. In the following equation, $T_s=1$ s and $k_{GPS}=1/1100$ s (Beard & McLain, 2012). Using position north, X , as an example, $X[n + 1] = e^{-k_{GPS}T_s} X[n] + N(0, \sigma_u^2)$. It is necessary to calculate σ_u^2 from the variance of X . Mohleji and Wang put forth a method to do this (Mohleji & Wang, 2010). Given that T_c is the time of correlation, $\sigma_u^2 = (1 - e^{-2/T_c})\sigma_x^2$. In the particular case of ADS-B where $\sigma_x = \sigma_y = 124$ ft and $T_c = 1100$ s, $\sigma_u = \sqrt{(1 - e^{-2/1100})\sigma_x^2} = 5.28$ ft. This is the variance of the Gaussian noise necessary for the zero-mean Gaussian random variable in the Gauss-Markov process with standard deviation $\sigma = 124$ ft.

FAA regulations require that ADS-B pressure altitude reporting equipment must report an altitude that is within 125 ft of the true altitude with 95% confidence (Federal Aviation Administration, 2015)(Federal Aviation Administration, 2010a). Let the pressure altitude error, A_{pres} , be a zero-mean Gaussian random variable such that $A_{pres} \sim N(0, \sigma_{A_{pres}}^2)$. It can then be shown that $\sigma_{A_{pres}} = 75.9$ ft. For geometric altitude reports the error is typically less than 147.6 ft with 95% certainty (Radio Technical Commission for Aeronautics, 2009)(Radio Technical Commission for Aeronautics, 2011). Assuming that the geometric altitude error, A_{geo} , is a zero-mean Gaussian random variable such that $A_{geo} \sim N(0, \sigma_{A_{geo}}^2)$, it can be shown that $\sigma_{A_{geo}} = 89.8$ ft. In addition to the noise of the pressure reporting sensors, the encoding of barometric altitude information has a resolution of 25 ft and geometric altitude information has a resolution of 45 ft. This resolution introduces some additional error.

Table 2: Resolution limits for ADS-B message information.

Message Element	Resolution
Latitude	0.5 deg
Longitude	0.5 deg
Altitude	25 ft
Horizontal Velocity	1 knot
Vertical Velocity	64 feet/min

The error in the ADS-B reported vertical velocity varies with increasing vertical rate. For vertical rates between ± 500 ft/min the vertical rate tolerance is ± 46 ft/min. For rates outside that range, the tolerance is 5% of the vertical rate (SAE International, 1996)(Radio Technical Commission for Aeronautics, 2003). Given the assumption that these tolerances are 95% bounds, it can be shown that the standard deviation of the climb rate is 27.96 ft/min for vertical rates of ± 500 ft/min. Additionally the vertical rate error is effected by the resolution of the ADS-B message encoding which is 64 ft/min.

The loss of valid ADS-B signal can be modeled using SIL, SDA, and MSR error. FAA regulations stipulate that position measurements outside the reported NIC can only be transmitted once per 10^7 transmissions. The SDA requirements permit values outside the NIC with a probability of 10^{-5} . MSR error requirements allow for a 10% or 15% message loss rate. These probabilities of erroneous or lost messages provide a method with which to model ADS-B signal dropout.

The error characteristics detailed above make it possible to model the error in ADS-B reported horizontal position, altitude, horizontal velocity, and vertical velocity. This results in a method capable of simulating ADS-B messages. It also provides a basis for estimating ADS-B messages and developing conflict detection, collision detection, separation assurance and collision avoidance methods.

ADS-B as a DAA Sensor

The characteristics and requirements of ADS-B make it a capable sensor for DAA on small UAS in the National Airspace System. One key aspect of ADS-B that makes it feasible for use on small UAS is the availability of ADS-B receivers that meet the (SWaP) constraints of a small UAS. The Clarity ADS-B receiver provides a dual-link ADS-B receiver that is 2.5 in by 2.5 in by 1.5 in, weighs 0.344 lbs, and consumes 2.4 Watts of power. Freeflight Systems has also recently introduced the RANGR RXD which is a dual-link ADS-B receiver. While slightly larger at 5 in by 5.75 in by 1.7 in, it still weighs less than one pound and consumes approximately 2.4 Watts of power. These hardware options both provide a suitable ADS-B In solution for small UAS.

Another key advantage of ADS-B is the long range at which information is available. While there is a significant amount of variation in the range of ADS-B signals, the shortest expected range is 10 nmi. Flight tests of ADS-B units suitable for small UAS have demonstrated reliable ranges of up to 80 nmi (Moody & Strain, 2009). Additionally the long range of ADS-B is advantageous in that the quality of information transmitted over ADS-B does not degrade with range. Thus the accuracy of ADS-B is not dependent on the size, power, or range of the transmitter and receiver units. This is a significant advantage over radar and optical sensors, and makes conflict detection and separation assurance path planning possible at long ranges.

A compelling result of the long range availability of ADS-B messages is the time to loss of separation (TLOS) and time to collision (TC). Figure 3 shows the TLOS and TC for head-on and over-taking scenarios given different intruder aircraft and various small UAS ownships. The detection range is set to the FAA required minimum of 10 nmi, and the separation distance is 0.66 nmi (Cook, Brooks, Cole, Hackenberg, & Raska, 2015). For this table a collision is defined as a violation of a 500 ft collision radius. The speeds listed are the maximum speeds for each aircraft.

			Ownship					
			RQ-11B Raven (43 kn)		ScanEagle (80 kn)		DJI Phantom 1 (19 kn)	
			Head-on Scenario	Over-Taking Scenario	Head-on Scenario	Over-Taking Scenario	Head-on Scenario	Over-Taking Scenario
Intruders	Model	Max. Speed (kn)						
	F-35	1042	31.0/32.3	33.2/34.6	30.0/32.2	34.3/35.7	31.7/33.0	32.8/34.2
	Boeing 747	533	58.4/60.9	66.8/69.6	54.9/57.2	71.2/74.2	60.9/63.5	65.3/68.0
	Cessna TTX	240	119.3/124.3	160.2/166.9	105.5/109.9	188.0/195.9	130.1/135.5	151.8/158.1
	Cessna SkyHawk	126	198.9/207.1	346.0/360.4	163.2/170.0	508.8/530.0	230.7/240.3	308.9/321.8
	Scan -Eagle	80	274.6/286.0	665.4/693.1	211.0/219.8	1730.1/1802.0	339.2/353.3	540.7/563.1
	RQ-11B Raven	43	393.2/409.5	2471.5/2574.3	274.6/286.0	-	540.7/563.1	1330.8/1386.2
Time to Loss of Separation (s) / Time to Collision (s)								

Figure 3: Table outlining the time to loss of separation and time to collision for ADS-B given 10 nmi detection range.

To identify a true worst-case scenario in the over-taking intruder configuration, the speed used for the ownship is a cruising speed rather than a maximum speed. The figure demonstrates the value of the long range detection available through ADS-B. Even for a worst-case scenario where an F-35 type aircraft is flying directly at a small UAS, the minimum TLOS is 30.0 s. This provides a sufficient amount of time for the UAS to perform an avoidance maneuver.

ADS-B is a very capable sensor for DAA on small UAS, but it is not without limitations. One notable limitation of ADS-B is that it is a cooperative technology. This means that to have visibility of other aircraft they also must be equipped with ADS-B. Given the FAA mandate that only some aircraft need to be ADS-B compliant, there certainly will be aircraft in lower altitudes that are not ADS-B equipped. While these lower altitudes are prime locations for small UAS operations, the capability of ADS-B presented in this paper provides motivation to implement an ADS-B equipage requirement for all aircraft. An additional technology that could be used to account for uncooperative aircraft, birds, and ground based obstacles is ADS-B radar. This technology is essentially a phase modulated ADS-B signal that is used as a radar and traditional ADS-B transmission simultaneously (C. Lai, Ren, & Lin, 2009). This would allow for visibility of uncooperative intruders. The method does require additional processing of the ADS-B signal and

some additional hardware, but it could be practical for UAS. While an in-depth discussion of this technology is outside the scope of this paper, it is promising.

Another limitation of ADS-B is that it is heavily dependent on line-of-sight availability of GPS and ADS-B transmissions. Without GPS information, ADS-B transponders are unable to transmit usable position information. Air-to-air ADS-B transmissions also require line-of-sight visibility for reliable exchange of information. One demonstrated solution to the line-of-sight limitation is the use of satellite-based ADS-B repeaters. This system uses ADS-B transceivers on satellites to gather and re-transmit ADS-B signals. This system allows for over the horizon visibility of other aircraft and could be particularly valuable in mountainous or heavily contoured terrain. Again the validation of this technology is beyond the scope the research presented in this paper.

The cost of ADS-B equipage may pose a limitation. Certified ADS-B Out hardware costs typically range from \$1,500 to \$25,000 USD. ADS-B In hardware costs range from \$400 to \$3,000. While these costs are not necessarily prohibitive, they are significant especially for many of the small-to-medium-sized companies that plan to use UAS for commercial purposes. For ADS-B to be a fully viable, accessible technology, hardware costs need to decrease. As the FAA 2020 mandate approaches an increasing number of companies are producing ADS-B hardware, and the cost of hardware is trending downward.

Ultimately the message elements, airspace and range requirements, hardware availability, and error characteristics of ADS-B make it a viable sensor for detect and avoid on small UAS in the NAS. While there are limitations to ADS-B sensors, development of promising solutions is reducing the impact of those limitations. As a DAA sensor, ADS-B offers all the information necessary to detect conflicts, maintain separation, and detect and prevent collisions.

CONFLICT/COLLISION DETECTION

The goal of conflict/collision detection is to identify intruder aircraft and determine the collision risk that they pose to the ownship. To do this, it is necessary to track and estimate the intruder states and extrapolate those states forward in time to identify possible future conflicts/collisions. In this section, we address the key components of a conflict/collision detection algorithm.

ADS-B Signal Processing

Estimation of the ADS-B messages is capable of mitigating some of the error in the transmitted measurements. The primary goal of estimation is to account for missed measurements that result from signal drop out or frequency congestion. Additionally, by filtering and estimating ADS-B measurements, it is possible to account for grossly erroneous measurements such as would be occasionally permitted through the SIL and SDA probabilities, smooth measurement noise that is typical of any real sensor, and estimate the transmitting aircraft state at a rate greater than the 1 Hz measurement rate (Krozel, Andrisani, Ayoubi, Hoshizaki, & Schwalm, 2004). Due to the fact that ADS-B messages contain an aircraft identifier such as the call sign or International Civil Aviation Organization (ICAO) address, there is no need for data association methods. This greatly simplifies the tracking task.

We use a Kalman filter to process ADS-B In tracks. The Kalman filter offers a linear estimator that is computationally efficient. The prediction model in our implementation is a constant-jerk model capable of accounting for high maneuverability of the intruders (Mehrotra & Mahapatra, 1997). While it is not expected that fixed-wing aircraft will maneuver aggressively, more aggressive maneuvers such as would be characteristic of a rotor-craft or small UAS must also be accounted for in the model. The states of the filter are position north, position east, altitude, velocity north, velocity east, climb rate, acceleration north, acceleration east, vertical acceleration, jerk north, jerk east, and vertical jerk. The measurements used to update the estimator states are the position north, position east, altitude, and climb rate. In updating the states, the transmitted horizontal velocities are ignored as a result of transmission errors. Recorded ADS-B data sets from the NAS have revealed that on rare occasions the north and east velocities are transmitted in reverse order resulting in an apparent velocity that is perpendicular to the actual direction of travel of the transmitting aircraft. Updating the Kalman filter with only a subset of measurements mitigates this problem and results in equally accurate estimation after a brief transient estimation period of several measurements.

Each transmitting aircraft broadcasts an ADS-B message approximately once per second; however, the broadcasts can occur at any point within a given second. Thus the Kalman filter must run at a higher rate than 1 Hz to account for the different times at which a transmission may be received. Our Kalman filter implementation runs at 10 Hz, and each received ADS-B message is assigned to the nearest discrete time-step.

A set of measurement gates is necessary to account for message dropout and grossly erroneous measurements. If at a given time step there is no measurement, only the Kalman filter prediction occurs. The update step occurs only when there is a valid measurement. The validity of the horizontal position and altitude measurements is determined separately due to the fact that in ADS-B messages the horizontal position and altitude can be updated at slightly different times. A horizontal position is determined to be valid if it is confirmed to be a new position and if the innovation falls within a 5 Mahalanobis distance bound. An altitude/climb rate measurement is valid only if it falls within a 5 Mahalanobis distance bound.

Each track is initialized using the first measurement from a given transmitting aircraft. The initial track covariance is initialized using the error levels given by the reported NAC_p and NAC_v and the error characterization described earlier. At each time step, the track covariance is monitored to ensure that the track is still valid. If the covariance of the track grows such that the position uncertainty in the track is greater than the NIC bound, then the track is determined to be invalid. Should another measurement from that aircraft be received, the track would be re-initialized.

The Kalman filter is capable of overcoming ADS-B message drop out and rejecting grossly erroneous measurements. Additionally it smooths the ADS-B signal and provides estimates of transmitting aircraft at a much faster rate than the 1 Hz measurement update rate. This ultimately allows for more accurate and more timely conflict and collision detection and resolution.

Conflict/Collision Risk Assessment

The main concern of the air traffic management system for manned aviation is safety, which is typically measured by number of incidents that happen when distance between aircraft becomes closer than a predefined safe distance to one another. This safety distance is quantified by means of a minimum allowed horizontal and vertical spacing (Prandini, Hu, Lygeros, & Sastry, 2000). As depicted in Figure 4, the collision volume or the protection zone is a virtual fixed-volume-based boundary. The general choice of this volume is a truncated cylinder of radius d_c and height h_c centered at the UAS current location. Current FAA regulations (14 CFR §91.113) have no explicit values for the collision volume (Federal Aviation Administration, 2015). Yet generally, 500 ft in radius and ± 100 ft in height is cited in the literature (Boskovic, Jackson, & Mehra, 2013; George, 2009). A near-midair collision (NMAC) is defined as an incident that occurs when two aircraft pass less than 500 ft horizontally and 100 ft vertically from each other. On the other hand, the collision volume threshold is a variable boundary that is dependent on the encounter geometry, time, distance to intruder and maneuverability (Consiglio, Chamberlain, Munoz, & Hoffer, 2012; George, 2009). As shown in Figure 4, a self-separation volume is added to the airspace volumes to provide a minimum practical separation distance between the UAS and any intruder, and to compensate for unexpected maneuvers by the intruders (Consiglio, Carreno, & Williams, 2005). In the context of DAA, the self-separation boundary is often called well clear to coincide with the FAA regulations (Cole et al., 2013). The self-separation volume is typically much larger than the collision volume but it may vary in size with operational area and airspace class. The self-separation threshold is then defined as the threshold boundary at which the UAS performs a maneuver to prevent the intruder from penetrating the self-separation volume. Hence, the addition of the self-separation volume provides a performance goal that is analogous to the collision volume.

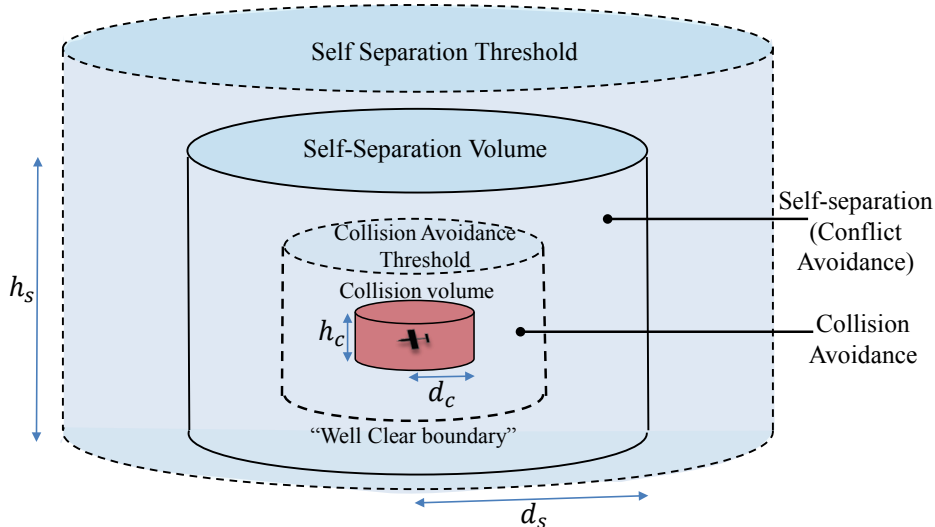


Figure 4: Definition of the DAA airspace volumes and thresholds.

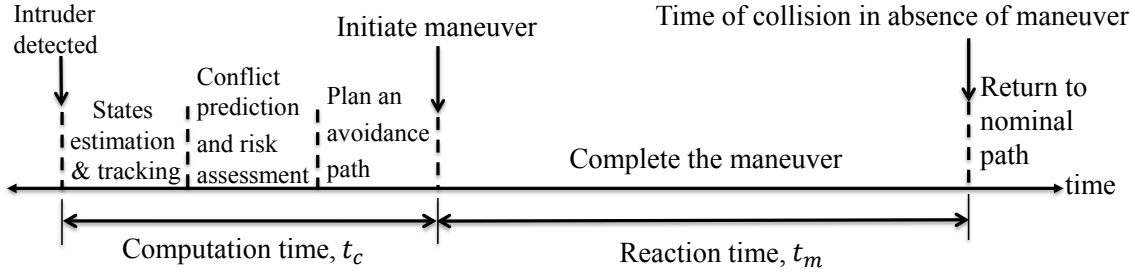


Figure 5: Proposed detect and avoid time line.

A time sequence of events for a DAA system, similar to the proposed sequence in Geyer, Singh, & Chamberlain, 2008, is shown in Figure 5. The minimum time required to perform an evasive maneuver and avoid the intruder by a safe distance determines the distance at which the UAS must detect the intruder. In other words, the detection of a collision threat must be done at a minimum range allowing the ownship to execute the maneuver with sufficient time that results in the minimum required safe distance from the intruder. Accordingly, the required sensing distance can be given as

$$d_r = v_c t_{daa}, \quad (1)$$

where v_c is the closing speed, and t_{daa} is the detection time required by the DAA system to be able to track the intruder, detect a collision, plan an avoidance maneuver and actually fly it. According to the time sequence shown in Figure 5, the t_{daa} is the sum of the computation time t_c and the reaction time t_r . The estimate of the time required for a manned aircraft to consistently avoid midair collisions range from 5 s to 12.5 s (*Collision Avoidance Functional Requirements for Step 1*, 2006). This time duration does not include the time required to perform an actual maneuver initiated by the collision avoidance system, and the estimate was for two jet aircraft with a closing speed of about 956 kn (*Collision Avoidance Functional Requirements for Step 1*, 2006). The minimum detection range can be derived based on collision geometry (Geyer, Singh, & Chamberlain, 2008; Hyunjin, 2013) or combining worst-case scenario analysis with extensive Monte Carlo simulations (Boskovic et al., 2013).

14 CFR does not provide any quantitative visual detection requirements for manned aviation other than pilots responsibility to be vigilant so as to see and avoid other aircraft according to the right-of-way rules (14 CFR §91.113). The FAA Aeronautical Information Manual (AIM) suggests that proper scanning of the sky is a key factor in collision avoidance. It should be used continuously by the pilot to cover all areas of the sky visible from the cockpit. On-board collision detection and avoidance instruments like the TCAS-II and the Air Traffic Control (ATC) support efficiently improve the pilots virtual visibility and awareness of surrounding air traffic and effectively resolve conflicts to a large extent.

The purpose of computing the collision risk is to have an alert threshold value above which the collision avoidance system is triggered to initiate an evasive maneuver to avoid an imminent collision with the detected intruding aircraft. There are a number of approaches to evaluate the future collision risk of an encounter situation. Most of these approaches can either be classified as geometric or probabilistic, where each approach has different techniques to deal with errors. In

the geometric approach, the collision risk is described based on the geometric relationship between aircraft. Aircraft trajectory predictions are based on linear projections of current aircraft states such that the uncertainty of the predicted trajectory is translated into areas around the predicted trajectory referred as to safe zones. Linear projections can be computationally efficient. Prediction errors are negligible over short time horizons (Geser & Muoz, 2002) or assumed known when flight plans are communicated (Munoz, Narkawicz, & Chamberlain, 2013). On the other hand, the probabilistic methods estimate the probability of collision based on a probabilistic model of future intruder dynamics. This event probability is then compared to a certain threshold above which the aircraft is deemed to be on a collision path. These probabilities can be estimated using an approximate analytical solution (Paielli & Erzberger, 1997), numerical approximation (van Daalen & Jones, 2009), or Monte Carlo methods (Jackson & Boskovic, 2012). In general, the probabilistic approaches are computationally intensive but suffer less from false alarms than geometric approaches.

Our approach is based on evaluating both near-term threats that need immediate action and long-term conflicts that can be smoothly resolved so that they will not become a close proximity threat. Similar to the resolution advisory concept and collision detection logic implemented by the current operationally mandated version of TCAS (TCAS II, Version 7.0) (Munoz et al., 2013), the proposed conflict/collision detection approach constructs a virtual volume surrounding the ownship UAS, that, when penetrated by the intruder, triggers the conflict/collision detection algorithm. These virtual boundaries are the collision volume threshold and the self-separation threshold shown in Figure 4. In the TCAS framework, this virtual volume is called the threat boundary, and its size is based on the τ criterion, which is a metric that is used to estimate the time to closest point of approach. The threshold value of τ used to construct the alert boundary varies from 15 s to 35 s, depending on the altitude layer at which the collision may occur (Munoz et al., 2013). The encounter geometry is evaluated in the relative coordinate frame where the relative motion of the aircraft is analyzed by investigating the dynamics of the intruder aircraft with respect to the ownship. In this relative frame of reference, the ownship is stationary and the intruder location and motion is determined by the relative position and velocity states. The relative position and velocity vectors are then used to derive the the time remaining to the horizontal closest point of approach (CPA) denoted as t_{cpa} , and the distance at the closest point of approach is denoted as d_{cpa} . In the vertical plane, the ownship is at the center of the relative coordinate system with a vertical protected zone from

Algorithm 1 Conflict/collision detection algorithm

```

1: if  $\exists t \in (0, T]$  such that
    •  $\mathbf{p}_r < d_{th}$  or (Intruder is converging horizontally and  $t_{cpa} \leq \tau_{th}$ ) and
    •  $|h_r| < h_{th}/2$  or (Intruder is converging vertically and  $t_{hc} \leq \tau_{th}$ )
    then
2:   if  $d_{cpa} < d_c$  and  $|h_{r_{cpa}}| < h_c/2$  then
3:     conflict/collision detected at time  $t$ 
4:   else
5:     no conflict/collision detected at time  $t$ 
6:   end if
7: end if

```

$-h_c/2$ to $h_c/2$. The time remaining to enter the vertical zone is denoted as t_{h_c} , and the relative altitude at the closest point of approach is denoted as $h_{r_{cpa}}$. Given the relative altitude between the ownship and aircraft h_r , and position and velocity states in the horizontal and vertical plane \mathbf{p}_r , \mathbf{v}_r , p_{r_z} , and v_{r_z} , respectively, then the pseudo code shown in Algorithm 1 predicts whether a conflict/collision will occur within a lookahead time window T .

Given a look ahead time window T , a collision occurs if there exists a future time $t \in [t_0, T]$ such that the relative position of the intruder is inside the collision volume. At any given time t , Algorithm 1 checks whether time to closest point of approach and time to enter the vertical protected zone fall below a threshold value τ_{th} anytime within the next T seconds. Algorithm 1 declares a collision if the relative horizontal distance, and the relative altitude at the closest point of approach is less than a minimum safe distance (i.e., $d_{cpa} < d_c$ and $|h_{r_{cpa}}| < h_c/2$). The parameters τ_{th} and T are positive real numbers and design parameters for the collision detection algorithm. Algorithm 1 can be easily modified to detect conflicts by changing the horizontal and vertical safe distances to be the radius and the height of the self-separation volume and choosing τ_{th} to satisfy the well clear boundary. In this work, we adopt the well clear candidate selected by the Sense and Avoid Research Panel (SARP) for recommendation to the sense and avoid stakeholders upon extensive analysis (Cook et al., 2015). The candidate well clear boundary is defined by a truncated cylinder that consists of a modified τ value of 35 s with distance of 4000 ft in the horizontal plane, and a vertical distance from the ownship of 700 ft. Hence, to detect conflicts over a long-range, in Algorithm 1 we modify the horizontal and vertical safe distances to be $d_s = 4000$ ft, and $h_s/2 = 700$ ft, respectively and $\tau_{th} = 35$ s.

Since there will always be uncertainty associated with these estimates, an additional provision should be made to account for it. Sampling-based methods such as Monte Carlo simulations and importance sampling are existing uncertainty propagation approaches. Instead of using extensive simulations, however, an analytic expression can also be used to propagate the error variance using a Taylor series approximation. This approach of variance estimation mostly used in statistical applications has several names in the literature, including the linearization method, the delta method, and propagation of variance (Lee & Forthofer, 2006). If we assume, for convenience, that aircraft in conflict use the same type of ADS-B sensor and that they are observing mostly the same GPS satellites, then both aircraft will experience the same position and velocity error accuracy. In addition, if the x - y components of the horizontal position and velocity are assumed to

Algorithm 2 Modified conflict/collision detection algorithm

```

1: if  $\exists t \in (0, T]$  such that
    •  $\mathbf{p}_r < d_{th}$  or (Intruder is converging horizontally and  $t_{cpa} \leq \tau_{th}$ ) and
    •  $|h_r| < h_{th}/2$  or (Intruder is converging vertically and  $t_{h_c} \leq \tau_{th}$ )
    then
2:   if  $d_{cpa} - m_1\sigma_{dcpa} < d_c$  and  $|h_{r_{cpa}}| - m_2\sigma_{h_{cpa}} < h_c/2$  then
3:     conflict/collision detected at time  $t$ 
4:   else
5:     no conflict/collision detected at time  $t$ 
6:   end if
7: end if

```

be independent random variables, then the error variances of the horizontal position and velocity measurements denoted as σ_p and σ_v , respectively are the same for both aircraft. It can then be shown that the variance of the distance at CPA is $\sigma_{dcpa}^2 = 2\sigma_p^2 + 2t_{cpa}\sigma_v^2$ (Krozel, 1997). Similarly, if the error variances of the vertical velocity and altitude measurements denoted as σ_h and σ_{vz} are independent and the same for both aircraft, then the variance of relative vertical altitude at CPA is $\sigma_{hcpa}^2 = 2\sigma_{hr}^2 + 2t_{cpa}\sigma_{vz}^2$. Therefore, Algorithm 1 is modified to become Algorithm 2 to account for uncertainties in the ownship and intruder states. In Algorithm 2, m_1, m_2 are design parameters for the conflict/collision detection algorithm and are constrained to be positive integers.

SELF-SEPARATION AND COLLISION AVOIDANCE

Since the collision-detection metrics are derived from the relative position and velocity vectors between conflicting aircraft, it is convenient to develop a collision avoidance logic using a translated coordinate system. In this relative coordinate system, the ownship is fixed at the center of the coordinate system, and the intruder is located at a relative position \mathbf{p}_r and moves with a relative velocity \mathbf{v}_r with respect to the ownship. An advantage of this approach is that collision avoidance is inherently a local phenomenon and can be more naturally represented in the local frame than the global frame.

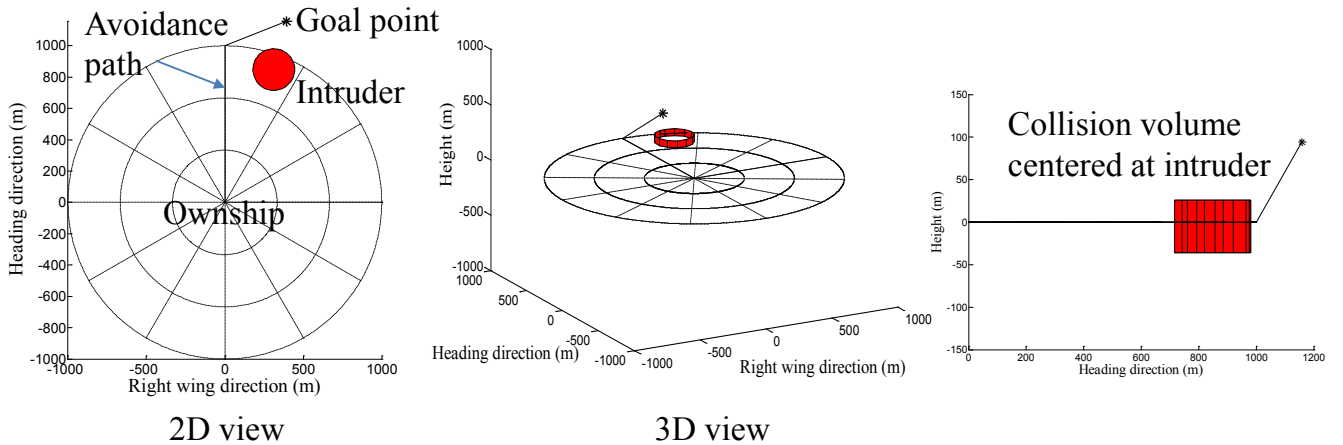


Figure 6: Local-level frame map.

We call this translated coordinate system the local-level frame because the environment is mapped to the unrolled, unpitched local frame where the ownship is stationary at the center. The local-level frame map is depicted in Figure 6, where the origin of the local-level frame is the center of mass of the ownship. In this configuration the x -axis points out the nose of the airframe, the y -axis points out the right wing direction, and the z -axis points down forming a right handed coordinate system. The intruder is encircled with the collision volume cylinder. The local-level frame also provides a convenient method for defining a collision between two aircraft. If the origin of the local-level frame penetrates the collision volume around the intruder, a collision is said to have occurred.

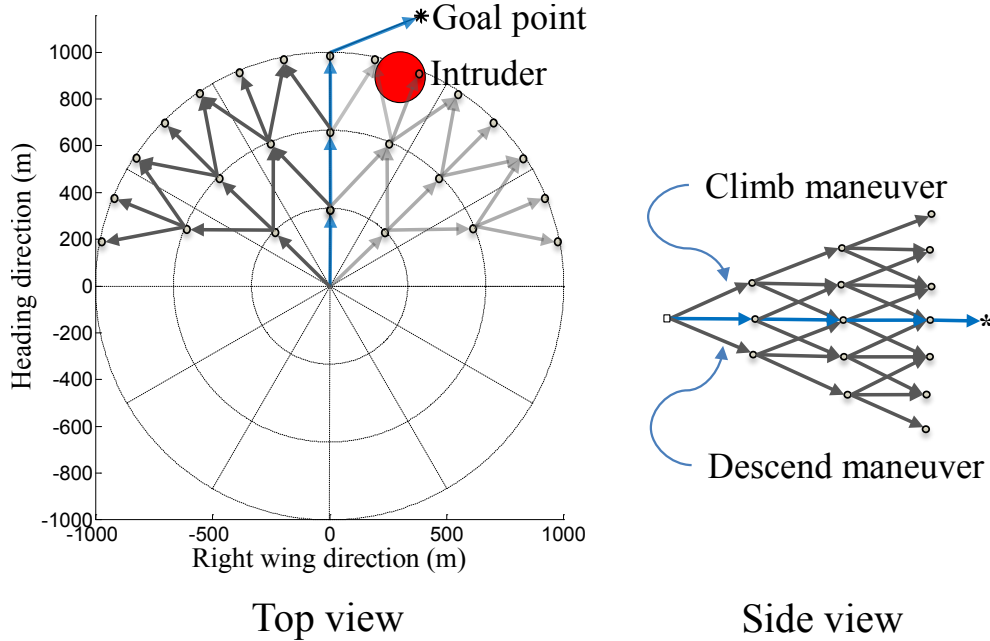


Figure 7: Discretized local-level map. The three concentric circles represent three maneuvers points.

As shown in Figure 7, the detection region is divided into concentric circles that represent three maneuvers points at increasing range from the ownship. The radius of the outmost circle represents the sensor detection range. We discretize the map and construct a weighted graph where the edges represent potential maneuvers, and the weights represent the collision risk and maneuver cost. The map is discretized by using a cylindrical grid, and the ownship is commanded to move along the edges of the grid. The result is a directed weighted graph that can be described by the tuple $\mathcal{G}(\mathcal{N}, \mathcal{E}, \mathcal{C})$, where \mathcal{N} is a finite nonempty set of nodes, and \mathcal{E} is a collection of ordered pairs of distinct nodes from \mathcal{N} such that each pair of nodes in \mathcal{E} is called a directed edge or link, and \mathcal{C} is the cost associated with traversing each edge.

In assigning a cost to each edge in the resulting graph, it is important to note that the main priority of an ownship under distress is to maneuver to avoid predicted conflicts/collisions. The cost associated with traveling along an edge is a function of the edge length and the collision risk. We set the cost associated with the length of the i^{th} edge to be equal for all edges except for the edges that connect the graph's outer nodes to the goal point. The cost of the conflict/collision risk for each edge can be defined using a binary cost function given as

$$C_{col,i} = \begin{cases} \infty, & \text{if } e_i \in \mathcal{B}, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

where \mathcal{B} is the collision volume cylinder centered at the detected intruder. The cost assigned to any edge that leads to a collision is ∞ , thus basically eliminating that edge and the path passing through it.

To provide an increased level of safety, an additional cost is used to penalize edges close to the collision volume even if they are not within the collision volume. Assuming a constant-velocity model, a linear extrapolation of the current position and velocity of the detected intruders is computed at evenly spaced time instants over the look-ahead time window. The look-ahead time interval is then discretized into several discrete time instants. At each discrete time instant the distances from the propagated locations of the intruders to all candidate locations of the ownship at that time instant. The cost of each edge is then the sum of the reciprocal of the associated distances to each intruder. Dijkstra’s algorithm is then employed to find the path with minimal cost from the start point to the goal point. Dijkstra’s algorithm solves the problem of finding the shortest path in a directed graph in polynomial time given that there are no negative weights assigned to the edges. The output of the local-level collision avoidance algorithm generates a waypoint avoidance path that consists of an ordered sequence of waypoints $\mathcal{W} = w_1, w_2, \dots, w_i$. These waypoints are basically nodes in the discretized local-level graph selected by the Dijkstra’s search.

A key feature of the proposed approach is that the future motion of the ownship is constrained to follow nodes on the map that are spaced by a constant time. Since the path is represented using waypoints that are at fixed time instants, it is easy to determine roughly where the ownship will be at any given time. This timing information can be used when assigning cost to edges to better plan paths and prevent collisions. To handle conflicts at long-range, the resolution algorithm should plan smooth maneuvers. This can be achieved by increasing the resolution of discretized map or by using smooth path parameterizations like Dubins paths.

SIMULATION RESULTS

To validate the performance of the presented ADS-B sensor model, estimation scheme, conflict/collision detection, and self-separation/avoidance approaches, we conducted two separate sets of Monte Carlo simulations to address encounter scenarios over short and long ranges. We developed a simulation environment with a five-degree-of-freedom aircraft model for both the ownship and the intruders. The state estimates of the intruders are provided by the ADS-B sensor, while we assume a perfect knowledge of the ownship states. To avoid simulating encounters that are unlikely to result in a collision or loss of self-separation, we focus on encounters that occur in an encounter circle centered on the ownship. The encounter geometry is constructed using an approach similar to that suggested by Kochenderfer, et al. (Kochenderfer, Kuchar, Espindle, & Griffith, 2008). The encounter circles used for simulating short and long range encounter scenarios have radii of 1.62 nmi and 10 nmi respectively. For both sets of Monte Carlo simulations, the ownship is initialized at the center of the encounter circle and follows a straight-line waypoint path. The ownship initial heading is zero, and the initial airspeed is set to 80 kn, which is similar to the maximum airspeed of the ScanEagle UAS. At the beginning of each simulation, the first intruder is initialized at one of 20 evenly spaced points on the perimeter of the encounter circle, while other intruders are initialized by sampling a uniform distribution over the remaining points. In addition, the intruders are initialized with random headings that are required to penetrate the encounter region at the initial time. The speed of the intruders is randomly drawn from a uniform

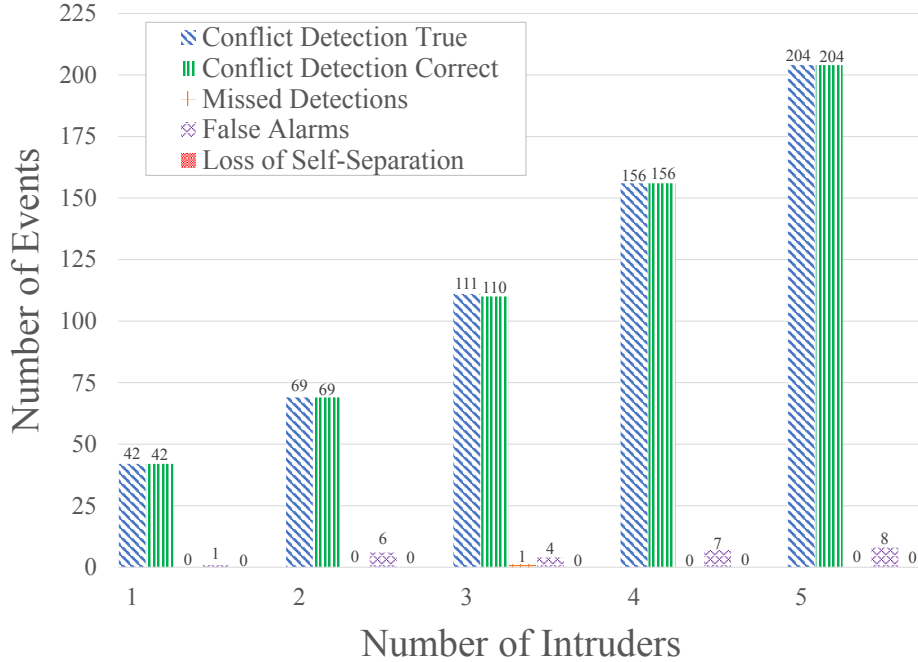


Figure 8: Self-separation evaluation metrics of ADS-B-based conflict detection and self-separation algorithms using 2,000 encounter scenarios for each intruder set (a total of 10,000 encounters).

distribution over [39, 250] kn. For the short-range encounter scenarios experiment, 1000 Monte Carlo simulation runs are conducted at each discrete point on the encounter circle. In the long-range encounter scenarios experiment, 100 Monte Carlo simulation runs are conducted for each discrete point on the encounter circle. In both sets of experiments, the Monte Carlo simulations are repeated to include multiple intruder scenarios.

We have implemented the conflict/collision detection method given by Algorithm 2. In the long-range experiment, the threshold boundary value that defines a conflict $\tau_{th} = 45$ s with distance threshold values of $d_{th} = 5$ nmi in the horizontal plane and $h_{th} = 1000$ ft in the vertical plane. To define loss of separation, we consider a well clear boundary defined by a τ value of 35 s with distance thresholds of 4000 ft in the horizontal plane and 700 ft in the vertical plane (Cook et al., 2015). In the short-range experiment, the well clear boundary serves as the threshold boundary so that the collision detection algorithm activates the collision avoidance planner once the well clear boundary is penetrated. This ensures that collision avoidance is triggered when the self-separation mode fails to maintain the well clear distance from the intruding aircraft. Our choice of collision volume is a truncated cylinder of radius 500 ft and height 100 ft. The design parameters used in Algorithm 2 for both experiments are

$$\begin{aligned}
 m_1 &= m_2 = 1, \\
 \sigma_{dcpa}^2 &= \sigma_p^2 + t_{cpa}^2 \sigma_v^2, \text{ where } \sigma_p = 124 \text{ ft}, \text{ and } \sigma_v = 8 \text{ kn} \\
 \sigma_{hcpa}^2 &= \sigma_{hr}^2 + t_{cpa}^2 \sigma_{vrz}^2, \text{ where } \sigma_{hr} = 75.9 \text{ ft}, \text{ and } \sigma_{vrz} = 27.96 \text{ ft/min}.
 \end{aligned}$$

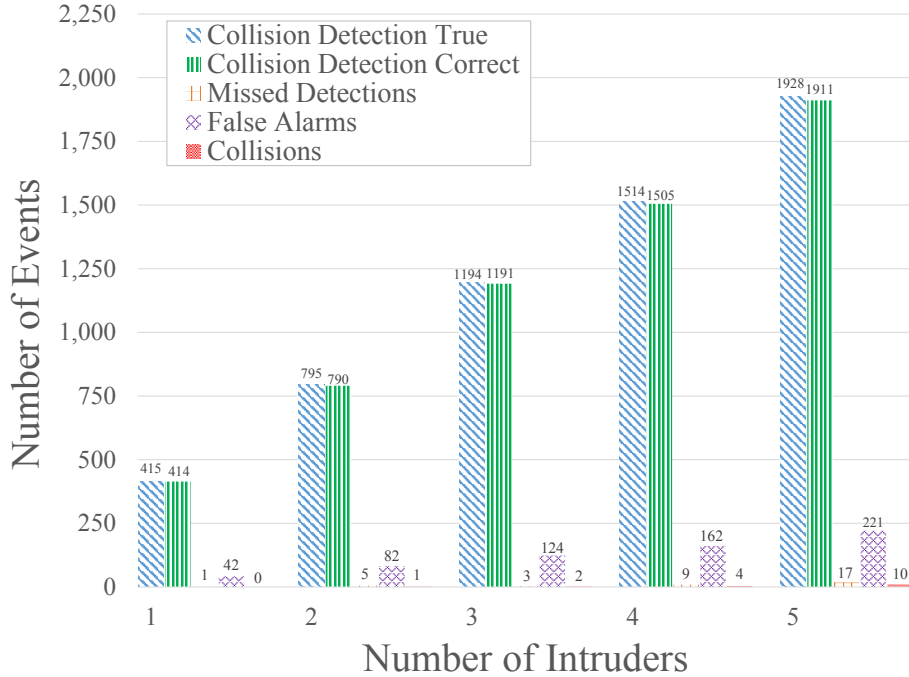


Figure 9: Collision avoidance evaluation metrics of ADS-B-based collision detection and avoidance algorithms using 20,000 collision encounter scenarios for each intruder set (a total of 100,000 encounters).

In each Monte Carlo simulation run we have also implemented the conflict/collision detection algorithm in Algorithm 1 to produce the intruder truth tracks needed to calculate the true number of conflict and collision events. The number of true conflict/collision detections is compared to the number of the conflict/collision detections that are predicted using Algorithm (2), which is based on the track estimates of the intruders. We have also recorded the number of collisions and loss of separation incidents for each encounter scenario given that the self-separation/collision avoidance algorithm will be activated once an intruder crosses the self-separation/collision avoidance threshold boundary. We have implemented the local-level frame path planner approach presented previously in the Self-separation and Collision Avoidance section. The local-level frame map is discretized into three levels with 38 nodes for the collision avoidance algorithm and five levels with 150 nodes for self-separation algorithm. The correct detections, missed detections, and false alarms for both the conflict and collision scenarios are shown in Figures 8 and 9.

Figure 8 shows results from the long-range self-separation assurance simulations, and Figure 9 contains results from the short-range tests for collision avoidance. In each figure five metrics are presented for the different numbers of intruders considered. Conflict Detection True refers to the number of collisions detected using truth information for each of the intruders. Missed Detections refers to the number of collisions detected using intruder truth information that were not detected using estimated intruder information. Conflict Detection Correct describes the number of conflicts detected using estimated information that were also detected using truth information. The

relationship between Conflict Detection True, Missed Detections, and Conflict Detection Correct can be expressed as $CD_{True} = Missed\ Detections + CD_{Correct}$. False Alarms is the number of collisions detected using estimated information that were not detected using truth information. Finally, Loss of Self-Separation and Collisions are the number of instances that the well clear or collision volumes were penetrated by an intruder.

As shown in Figures 8 and 9, both conflict and collision detection algorithms are able to correctly detect conflict and collision events with a small number of missed detections and few false alarms. The increasing number of conflict and collision incidents with the increasing number of intruders is expected since greater numbers of intruders in the encounter circle result in a higher probability of a conflict or collision with the ownship.

Table 3: Conflict Detection Algorithm: Probability of correct detection, probability of false alarm, and safety ratio.

Scenario	P_{cd}	P_{fa}	Safety Ratio
1 intruder	0.998	0.00214	0.002
2 intruders	0.994	0.00209	0.006
3 intruders	0.997	0.00211	0.003
4 intruders	0.994	0.00206	0.006
5 intruders	0.991	0.00225	0.009

Table 4: Collision Detection Algorithm: Probability of correct detection, probability of false alarm, and safety ratio.

Scenario	P_{cd}	P_{fa}	Safety Ratio
1 intruder	1.000	0.00051	0.000
2 intruders	1.000	0.00153	0.000
3 intruders	0.990	0.00068	0.010
4 intruders	1.000	0.00089	0.000
5 intruders	1.000	0.00082	0.000

The performance of the proposed conflict/collision detection approach is quantified using the probability of correct detection P_{cd} and the probability of false alarm P_{fa} (Sahawneh, Mackie, Spencer, Beard, & Warnick, 2015; Kuchar, 1996). If N is the number of performed simulations, among which there are E true conflict/collision events, and the proposed conflict/collision detection algorithm detects M conflicts/collisions, among which \tilde{E} detections are the correct conflict/collision detections, then the correct detection rate P_{cd} and false alarm rate P_{fa} are given by

$$P_{cd} = \frac{\tilde{E}}{E}, \quad P_{fa} = \frac{M - \tilde{E}}{N - E}.$$

We further quantify the system performance by computing the safety ratio (Kuchar, 1996)

$$\text{Safety ratio} = \frac{1 - P_{cd}}{1 - P_{fa}}$$

The P_{cd} and P_{fa} results for both collision and conflict detection are shown in Tables 3 and 4. An ideal conflict/collision detection algorithm would result in $P_{fa} = 0$ and $P_{cd} = 1$, with all conflicts/collisions threats correctly detected and no false alarms. This corresponds to the theoretical ideal point $(P_{fa}, P_{cd}) = (0,1)$ in signal detection theory. Unfortunately, due to uncertainty in the intruder state estimates, this ideal is not achievable. The closer the value of (P_{fa}, P_{cd}) to the point $(0,1)$, however, the better the detection performance. In addition, a safety ratio of 0 indicates that the detection system provides perfect protection from loss of separation/collision incidents. A safety ratio of 1, however, indicates that the detection system provides no additional protection from loss of separation/collision incidents. The results shown in Tables 3 and 4 demonstrate that our approach produces results near the ideal operation point $(P_{fa}, P_{cd}) = (0,1)$ with a safety ratio that is near zero.

Table 5: Conflict Detection Algorithm: Event detection time delay.

Scenario	Average max. time delay (s)	95th percentile	Max. time delay (s)
1 intruder	1.389	11.280	12.9
2 intruders	1.308	8.900	13.7
3 intruders	1.370	8.160	24.9
4 intruders	1.444	6.255	36.9
5 intruders	1.148	5.200	18.9

Table 6: Collision Detection Algorithm: Event detection time delay.

Scenario	Average max. time delay (s)	95th percentile	Max. time delay (s)
1 intruder	2.010	9.990	18.2
2 intruders	2.348	9.900	31.9
3 intruders	2.219	9.900	35.9
4 intruders	2.354	10.245	31.9
5 intruders	2.190	9.900	28.5

The conflict/collision detection algorithm can be also evaluated by measuring the time delay between a conflict/collision event that has been detected using the intruder truth information, and the time instant at which the proposed conflict/collision detection algorithm is able to detect the same event using estimated intruder information. Ideally, the time delay should be zero. Due

Table 7: Conflict Detection: Execution time for one cycle of estimator, and conflict detection, and self-separation algorithms.

Scenario	Average max. computation time (s)	95th percentile	Max. computation time (s)
1 intruder	0.218	0.292	0.363
2 intruders	0.387	0.523	0.649
3 intruders	0.519	0.627	0.645
4 intruders	0.521	1.022	2.131
5 intruders	0.578	0.930	2.403

Table 8: Collision Detection: Execution time for one cycle of estimator, and collision detection, and avoidance algorithms.

Scenario	Average max. computation time (s)	95th percentile	Max. computation time (s)
1 intruder	0.108	0.198	0.386
2 intruders	0.108	0.329	0.392
3 intruders	0.072	0.079	0.140
4 intruders	0.014	0.026	0.734
5 intruders	0.019	0.042	0.878

to the errors in the state estimates of the intruders, however, this cannot be always achieved. The average of the maximum, the 95th percentile, and the maximum of simulation time delays is shown in Tables 5 and 6. Figure 3 shows that a small UAS similar to a ScanEagle has 105.5 s to a loss of separation and 109.9 s to a collision when encountering an intruder approaching head-on at speed of 240 kn. Table 6 shows that the average of the maximum time delay to detect a collision for all of the five different scenarios is 2.224 s which leaves the ownship with about 103 s to plan and fly an avoidance maneuver. Even with the maximum time delay recorded (35.9 s), the ownship still has about 74 s to plan and execute an avoidance maneuver.

Another important aspect of evaluating the performance of the whole system is its ability to run in real time. The DAA system must be able to process the measurement information provided by the sensor, provide estimates of the current states of the intruders, assess the encounter risk, and plan an avoidance path, while leaving ample time to avoid the intruder and ensure that the constraint on the minimum required safe distance from the intruder is not violated. To demonstrate this capability, we have recorded the time required to execute these processes from the point the measurements are provided to the estimator until a new waypoint command is generated by the path planner and is ready to be delivered to the ownship autopilot. The average, 95th percentile, and the maximum time required to execute one cycle of the estimation, the conflict/collision detection, and the self-separation/collision avoidance algorithm are shown in Tables 7 and 8. The algorithms were run using Matlab on an Intel i7 processor. The ADS-B estimator provides

estimates at a sample rate of 0.1 s. To process all of the estimator data, the other algorithms should operate at the same sample rate. Table 7 shows that the average of the maximum run time to execute the estimator, conflict detection, and the self-separation algorithms is 0.445 s, and the maximum recorded run time is 2.403 s. Similarly, in Table 8 the average of the maximum run time of the estimator, collision detection, and avoidance algorithms for all of the five different scenarios is 0.064 s, while the maximum recorded run time is 0.878 s. Although the recorded run times are larger than the 0.1 s, these algorithms are running near real time in Matlab. We expect that implementing these algorithms in a compiled language, such as C or C++, will show that real-time execution is feasible using low-cost computational hardware.

In these experiments, the collision volume is large (500 ft radius, 100 ft height) compared to the size of the ownship aircraft. If the collision volume were reduced to more closely reflect the aircraft size, the number of collisions detected drops even further. For example, Figure 9 shows a total number of 10 collisions out of the 1928 potential collisions detected using intruder truth information for the five intruder scenario case. If the collision volume is reduced to a third of the original size, then the number of collisions decreases to only one. If the collision volume is further reduced to a cylinder of radius 10 ft, approximating the wingspan of a small UAS, and height of 3 ft, we record zero collisions. From this we can conclude that none of these collision incidents were actual collisions, but only a violation of the surrounding collision volume.

CONCLUSIONS

We have introduced in this paper an ADS-B-based collision detection and avoidance approach that is computationally feasible for small UAS. Taking advantage of the long-range detection offered by the ADS-B system, the proposed approach is used to detect conflict events over the range of 5 to 10 nmi, and consequently provide self-separation of the host ownship.

The results from Monte Carlo simulations show that the proposed system supports a high level of safety. In the 10,000 executed self-separation simulations, well clear was fully maintained. For the 100,000 collision scenarios, the number of collision-volume violations was only 17, with no physical collisions occurring. The run-time results also indicate that the estimation scheme, detection method, and avoidance logic are capable of running in near real time in Matlab. Considering the increase in computational efficiency that would result from a compiled-code implementation of the algorithm, real-time execution can be easily achieved.

A key result of this paper is that it demonstrates that mandating ADS-B Out equipage for all aircraft could play a significant role in the safe integration of UAS into the NAS. The theoretical basis and positive simulation results presented here demonstrate the feasibility of the proposed system. While the FAA 2020 mandate for ADS-B includes only some aircraft, requiring all aircraft to have ADS-B Out would create a suitable environment for implementation of ADS-B-based DAA. Ultimately, the combination of complete ADS-B equipage and DAA system like the one presented here would be a major step toward allowing UAS to safely operate in the NAS with manned aircraft.

ACKNOWLEDGMENTS

This research was conducted in the NSF Center for Unmanned Aircraft Systems (C-UAS) with support from National Science Foundation I/UCRC program grant number IIP-1171036 and C-UAS industry members.

REFERENCES

- Albaker, B. M., & Rahim, N. A. (2010). Unmanned Aircraft Collision Detection and Resolution: Concept and Survey. *2010 5th IEEE Conference on Industrial Electronics and Applications*, 248–253.
- Angelov, P. (2012). *Sense and Avoid in UAS: Research and Applications*. John Wiley & Sons, Ltd.
- Beard, R., & McLain, T. (2012). *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press.
- Boskovic, J. D., Jackson, J. A., & Mehra, R. k. (2013, Sep). Sensor and Tracker Requirements Development for Sense and Avoid Systems for Unmanned Aerial Vehicles. In *AIAA Modeling and Simulation Technologies (MST) Conference*. Boston, MA.
- Cirillo, M. (2005). *Air Traffic Bulletin: New Technology - ADS-B, TIS-B, and FIS-B* (No. 2005-3).
- Cole, R., Kochenderfer, M. J., Weibel, R., Edwards, M. W. M., Griffith, J. D., & Olson, W. (2013). Fielding a Sense and Avoid Capability for Unmanned Aircraft Systems: Policy, Standards, Technology, and Safety Modeling. *Air Traffic Control Quarterly, special Issue on Unmanned Aircraft Systems*, 21(1), 5-27.
- Collision Avoidance Functional Requirements for Step 1* (Tech. Rep.). (2006, Feb.). Edwards, CA: NASA.
- Consiglio, M., Carreno, V., & Williams, D. (2005). Conflict Prevention and Separation Assurance Method in Small Aircraft Transportation System. In *AIAA 5th Aviation, Technology, Integration, and Operations Conference (ATIO)*. Arlington, Virginia.
- Consiglio, M., Chamberlain, J., Munoz, C., & Hoffer, K. (2012). Concept of Integration For UAS Operations in The NAS. In *28th International Congress of the Aeronautical Sciences (ICAS)*. Brisbane, Australia.
- Cook, S. P., Brooks, D., Cole, R., Hackenberg, D., & Raska, V. (2015, January). Defining Well Clear for Unmanned Aircraft Systems. In *AIAA Infotech @ Aerospace*. Kissimmee, Florida: AIAA.
- Dey, D., Geyer, C., Singh, S., & Digioia, M. (2009). Passive, Long-range Detection of Aircraft : Towards a Field Deployable Sense and Avoid System. In *Field & Service Robotics*.
- Dijkstra, E. W. (1959). A Note On Two Problems In Connection With Graphs. *Numerische Mathematik*, 1, 269-271.
- FAA Modernization and Reform Act of 2012*. (2014).
- Federal Aviation Administration. (2010a). *Advisory Circular 20-165: Airworthiness Approval of Automatic Dependent Surveillance - Broadcast (ADS-B) Out Systems*.

- Federal Aviation Administration. (2010b). *Automatic Dependent Surveillance Broadcast (ADSB) Out Performance Requirements To Support Air Traffic Control (ATC) Service*.
- Federal Aviation Administration. (2012). *Advisory Circular 90-114: Automatic Dependent Surveillance-Broadcast (ADS-B) Operations*.
- Federal Aviation Administration. (2015). *Subchapter F - Air Traffic and General Operating Rules*.
- George, S. (2009). *FAA Workshop on Sense and Avoid (SAA) for Unmanned Aircraft Systems (UAS)*.
- Geser, A., & Muoz, C. (2002). A Geometric Approach to Strategic Conflict Detection And Resolution. In *Proceedings of the 21st IEEE Digital Avionics Systems Conference* (Vol. 1).
- Geyer, C., Singh, S., & Chamberlain, L. (2008). *Avoiding Collisions Between Aircraft: State of the Art and Requirements for UAVs operating in Civilian Airspace* (Tech. Rep.).
- Hottman, S., Hansen, K., & Berry, M. (2009). *Literature Review on Detect, Sense, and Avoid Technology for Unmanned Aircraft Systems* (Tech. Rep.).
- Hu, J., Lygeros, J., Prandini, M., & Sastry, S. (1999). Aircraft Conflict Prediction and Resolution Using Brownian Motion. In *Proceedings of the 38th Conference on Decision & Control*. Phoenix.
- Hyunjin, Y. K., Choi. (2013). Reactive Collision Avoidance of Unmanned Aerial Vehicles Using a Single Vision Sensor. *AIAA Guidance, Control, and Dynamics*, 36(4), 1234-1240.
- Jackson, J., & Boskovic, J. (2012, August). Application of Airspace Encounter Model for Prediction of Intruder Dynamics. In *AIAA Modeling and Simulation Technologies Conference*. AIAA.
- Kavraki, L. E., Svestka, P., Latombe, J. C., & Overmars, M. H. (1996). Probabilistic Roadmaps for Path Planning in High-dimensional Configuration Spaces. *IEEE Transactions on Robotics and Automation*, 12(4), 566-580.
- Kochenderfer, M. J., Kuchar, J. K., Espindle, L. P., & Griffith, J. D. (2008). *Uncorrelated Encounter Model of the National Airspace System Version 1.0* (Tech. Rep.). Lincoln Laboratory-Massachusetts Institute of Technology.
- Kothari, M., & Postlethwaite, I. (2013). A Probabilistically Robust Path Planning Algorithm for UAVs Using Rapidly-Exploring Random Trees. *International Robotic Systems*, 71, 231-253.
- Krozel, J. (1997). Strategic Conflict Detection and Resolution for Free Flight II. , 1822-1828.
- Krozel, J., Andrisani, D., Ayoubi, M. A., Hoshizaki, T., & Schwalm, C. (2004). Aircraft ADS-B Data Integrity Check. , 1-11.
- Kuchar, J. K. (1996). Methodology for Alerting-System Performance Evaluation. *Guidance, Control, and Dynamics*, 19(2), 438-444.
- Kuchar, J. K., & Yang, L. C. (2000, December). A Review of Conflict Detection and Resolution Modeling Methods. *IEEE Transactions on Intelligent Transportation Systems*, 1(4), 179-189.
- Lai, C., Ren, Y.-j., & Lin, C. (2009). ADS-B Based Collision Avoidance Radar for Unmanned Aerial Vehicles. In *Ieee 2009 microwave symposium digest* (pp. 85-88).
- Lai, J., Ford, J. J., Mejias, L., Shea, P. O., & Walker, R. (2012). See and Avoid Using Onboard Computer Vision. In *Sense and Avoid in UAS: Research and Applications* (pp. 265-294).
- Lam, T. M., Mulder, M., Van Paassen, M. M., Mulder, J. A., & Van Der Helm, F. C. (2009). Force-

- Stiffness Feedback in Uninhabited Aerial Vehicle Teleoperation with Time Delay. *AIAA Guidance, Control, and Dynamics*, 32(3), 821-835.
- LaValle, S. M. (1998). Rapidly-Exploring Random Trees: A New Tool for Path Planning. Retrieved from <http://msl.cs.uiuc.edu/~lavalle/rrtpubs.html>
- Lee, E. S., & Forthofer, R. N. (2006). *Analyzing Complex Survey Data* (Second ed.). Thousand Oaks, CA.: Sage Publications, Inc.
- Luders, B., Karaman, S., Frazzoli, E., & How, J. P. (2010). Bounds on Tracking Error using Closed-Loop Rapidly-Exploring Random Trees. In *American Control Conference*. Baltimore, MD.
- Luders, B., Karaman, S., & How, J. P. (2013). Robust Sampling-based Motion Planning with Asymptotic Optimality Guarantees. In *Guidance, Navigation, and Control (GNC) Conference*. Boston, MA.
- Mackie, J., Spencer, J., & Warnick, K. (2014). Compact FMCW Radar for GPS-Denied Navigation and Sense and Avoid. In *IEEE Antennas and Propagation Society, AP-S International Symposium* (pp. 989–990).
- Mehrotra, K., & Mahapatra, P. R. (1997). A Jerk Model for Tracking Highly Maneuvering Targets. *IEEE Transactions on Aerospace and Electronic Systems*, 33(4), 1094–1105. doi: 10.1109/7.624345
- Miroló, C., & Pagello, E. (1995). A Cell Decomposition Approach to Motion Planning Based on Collision Detection. In *Proceedings of the 1995 International Conference on Advanced Robotics* (p. 481-488).
- Mohleji, S. C., & Wang, G. (2010). *Modeling ADS-B Position and Velocity Errors for Airborne Merging and Spacing in Interval Management Application* (Tech. Rep.). McLean, VA: Center for Advanced Aviation System Development.
- Moody, C., & Strain, R. (2009, April). Implementation Consideration for Automatic Dependent Surveillance - Broadcast on Unmanned Aircraft Systems. In *AIAA Infotech@Aerospace Conference* (pp. 1–8). Reston, Virginia. doi: 10.2514/6.2009-1865
- Munoz, C., Narkawicz, A., & Chamberlain, J. (2013). A TCAS-II Resolution Advisory Algorithm. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*. Boston, MA.
- Paielli, R. A., & Erzberger, H. (1997). Conflict Probability Estimation for Free Flight. *AIAA Guidance, Control, and Dynamics*, 20(3), 588-596.
- Papoulis, A., & Pillai, S. U. (2002). *Probability, Random Variables, and Stochastic Processes* (Fourth ed.). McGraw-Hill New York.
- Prandini, M., Hu, J., Lygeros, J., & Sastry, S. (2000). A Probabilistic Approach to Aircraft Conflict Detection. *IEEE Transactions on Intelligent Transportation Systems*, 1(4), 199-220.
- Radio Technical Commission for Aeronautics. (2003). *Minimum Aviation System Performance Standards for Aircraft Surveillance Applications*.
- Radio Technical Commission for Aeronautics. (2009). *DO-282B Minimum Operational Performance Standards for Universal Access Transceiver (UAT) Automatic Dependent Surveillance - Broadcast (ADS-B) DRAFT Final Review and Comment* (No. 190).
- Radio Technical Commission for Aeronautics. (2011). *DO-260B Minimum Operational Performance Standards for 1090 MHz Extended Squitter Automatic Dependent Surveillance Broadcast (*

ADS-B) and Traffic Information Services Broadcast (TIS-B).

- Rajnikant, S., Saunders, J. B., & Beard W., R. (2012). Reactive Path Planning for Micro Air Vehicles Using Bearing-only Measurements. *International Robotic Systems, Volume 65*(1-4), 409-416.
- SAE International. (1996). *AS8002A Air Data Computer - Minimum Performance Standard*.
- Sahawneh, L. R., Mackie, J., Spencer, J., Beard, R. W., & Warnick, K. F. (2015). Airborne Radar-Based Collision Detection and Risk Estimation for Small Unmanned Aircraft Systems. *Aerospace Information Systems, 0(0)*, Ahead of Print: 1-11.
- Sahawneh, L. R., Randal W. Beard, S. A., & Bai, H. (2013). Chain-based Collision Avoidance for UAS Sense and Avoid Systems. In *Proceedings of the AIAA Guidance, Navigation and Control Conference*.
- Unmanned Aircraft System (UAS) Service Demand 2015-2035* (Tech. Rep.). (2013). U.S. Department of Transportation. Retrieved from <http://fas.org/irp/program/collect/service.pdf>
- van Daalen, C., & Jones, T. (2009). Fast Conflict Detection Using Probability Flow. *Automatica, 45*, 1903-1909.

BIOGRAPHIES

Laith R. Sahawneh is currently a Ph.D. candidate and a research assistant in the Electrical and Computer Engineering Department at Brigham Young University. His current research focuses on collision prediction, path planning, and detect and avoid for small UAS. He received his M.S. degree in mechatronics engineering in 2009 from the American University of Sharjah, UAE.

Matthew O. Duffield is graduate student in the Mechanical Engineering Department at Brigham Young University in Provo, Utah. He is pursuing a joint Master of Science in Mechanical Engineering and a Master of Business Administration. His research is focused on the opportunities and applications of Automatic Dependent Surveillance-Broadcast for detect and avoid on small unmanned aircraft. In 2014 he was recognized as a Utah NASA Space Grant Consortium Fellow. Matthew completed his Bachelor of Science degree in Mechanical Engineering at Brigham Young University in 2014.

Randal W. Beard received a Ph.D. degree in electrical engineering in 1995 from Rensselaer Polytechnic Institute. He is currently a professor in the Electrical and Computer Engineering Department at Brigham Young University. His primary research focus is in autonomous systems, unmanned air vehicles, and multiple vehicle coordination and control. He is a fellow of the IEEE and an associate fellow of AIAA. He is a former associate editor for the IEEE Transactions on Automatic Control, the Journal of Intelligent and Robotics Systems, and the IEEE Control Systems Magazine.

Timothy W. McLain is a professor of mechanical engineering at Brigham Young University. During 1999 and 2000, he was a visiting scientist at the Air Force Research Laboratory where

he initiated research in the guidance and control of unmanned aircraft systems. Since then, his UAS research has attracted the support of the Air Force, the Army, DARPA, NASA, NSF, and ONR. He is the author of over 120 peer-reviewed articles. With Randy Beard, he is the author of the textbook *Small Unmanned Aircraft*. He is currently the director of the Center for Unmanned Aircraft Systems sponsored by the National Science Foundation.