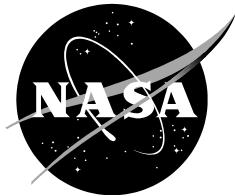


NASA/TM—2016-219441



# **UAS Integration in the NAS Project: Flight Test 3 Data Analysis of JADEM-Autoresolver Detect and Avoid System**

*Chester Gong*

*Ames Research Center, Moffett Field, California*

*Minghong G. Wu*

*Ames Research Center, Moffett Field, California*

*Confesor Santiago*

*Ames Research Center, Moffett Field, California*

---

**December 2016**

## NASA STI Program ... in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

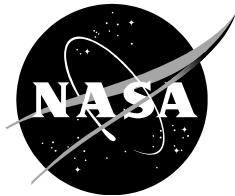
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- E-mail your question to [help@sti.nasa.gov](mailto:help@sti.nasa.gov)
- Phone the NASA STI Information Desk at 757-864-9658
- Write to:  
NASA STI Information Desk  
Mail Stop 148  
NASA Langley Research Center  
Hampton, VA 23681-2199



# **UAS Integration in the NAS Project: Flight Test 3 Data Analysis of JADEM-Autoresolver Detect and Avoid System**

*Chester Gong*

*Ames Research Center, Moffett Field, California*

*Minghong G. Wu*

*Ames Research Center, Moffett Field, California*

*Confesor Santiago*

*Ames Research Center, Moffett Field, California*

National Aeronautics and  
Space Administration

*Ames Research Center  
Moffett Field, California, 94035-1000*

---

**December 2016**

This report is available in electronic form at

<http://www.aviationsystemsdivision.arc.nasa.gov/publications/index.shtml>

## Table of Contents

Nomenclature .....	4
Introduction .....	4
Autoresolver Detect and Avoid System Description .....	5
Java Architecture for Detect And Avoid (DAA) Extensibility and Modeling (JADEM)....	5
Well Clear and Alerting Threshold Definitions .....	5
Autoresolver Directive Guidance Algorithm.....	8
Flight Test Operations .....	9
Test Aircraft .....	9
Encounter Scenario Design.....	9
Flight Test Data Description .....	12
Overview of Flight Test Data Collection System .....	12
Data Collection Anomalies During Flight.....	13
Post Processed Flight Data .....	13
Additional Computed Data.....	14
Analysis and Results .....	15
Predicted Separation Error .....	16
Alerting Analysis.....	21
Turn Prediction Accuracy Analysis .....	26
Conclusions .....	29
References.....	29
Appendix – Scenario Summary.....	30

## Nomenclature

ADS-B	Automatic Dependent Surveillance - Broadcast
AFRC	Armstrong Flight Research Center
ATC	air traffic control
CPA	closest point of approach
DAA	detect and avoid
DMOD	distance modification
FT3	Flight Test 3
GPS	Global Positioning System
HMD	horizontal miss distance
ICAO	International Civil Aviation Organization
IP	initial point
JADEM	Java Architecture for DAA Extendibility and Modeling
LVC	Live Virtual Constructive
MOPS	Minimum Operating Performance Standards
NAS	National Airspace System
NASA	National Aeronautics and Space Administration
nmi	nautical miles
SAAP	Sense-and-Avoid Processor
TCAS	Traffic Alert/Collision Avoidance System
$T_{\text{CPA}}$ , $T_{\text{mod}}$	time at closest point of approach modified tau
VSCS	Vigilant Spirit Control System
UAS	Unmanned Aircraft System

## Introduction

The Unmanned Aircraft Systems Integration in the National Airspace System project, or UAS Integration in the NAS, aims to reduce technical barriers related to safety and operational challenges associated with enabling routine UAS access to the NAS. The UAS Integration in the NAS Project conducted a flight test activity, referred to as Flight Test 3 (FT3), involving several Detect-and-Avoid (DAA) research prototype systems between June 15, 2015 and August 12, 2015 at the Armstrong Flight Research Center (AFRC). This report documents the flight testing and analysis results for the NASA Ames-developed JADEM-Autoresolver DAA system, referred to as "Autoresolver" herein. Four flight test days (June 17, 18, 22, and July 22) were dedicated to Autoresolver testing.

The objectives of this test were as follows:

1. Validate CPA prediction accuracy and detect-and-avoid (DAA, formerly known as self-separation) alerting logic in realistic flight conditions
2. Validate DAA trajectory model including maneuvers
3. Evaluate TCAS/DAA interoperability
4. Inform final Minimum Operating Performance Standards (MOPS)

Flight test scenarios were designed to collect data to directly address the objectives 1-3. Objective 4, inform final MOPS, was a general objective applicable to the UAS in the NAS project as a whole, of which flight test is a subset. This report presents analysis

results completed in support of the UAS in the NAS project FT3 data review conducted on October 20, 2015. Due to time constraints and, to a lesser extent, TCAS data collection issues, objective 3 was not evaluated in this analysis.

## **Autoresolver Detect-and-Avoid System Description**

### **Java Architecture for DAA Extendibility and Modeling (JADEM)**

Java Architecture for Detect-And-Avoid (DAA) Extensibility and Modeling (JADEM) was developed at NASA Ames Research Center as a research and modeling tool for the Unmanned Aircraft System (UAS) Integration in the National Airspace System (NAS) Project. Unmanned aircraft will be equipped with a DAA system that enables them to comply with the requirement to "see and avoid" other aircraft (i.e., remain "well clear"), an important layer in the overall set of procedural, strategic and tactical separation methods designed to prevent mid-air collisions. JADEM supports research on technical requirements and MOPS for UAS DAA systems by providing a flexible and extensible software platform that includes models and algorithms for all major DAA functions, as well as interfaces that allow for refinements to existing DAA functions. JADEM contains an abstract interface allowing for the integration of external DAA model implementations. JADEM is written in Java, and provides an Application Programming Interface for modeling DAA functions in a user's simulation environment of flight test infrastructure.

There are seven functions within the typical DAA functional architecture handled by JADEM: Detect, Track, Evaluate, Prioritize, Declare, Determine, and Command. The Detect and Track functions model the surveillance system's sensors on-board the UAS used to detect other aircraft. JADEM is able to model multiple sensor configurations including parameters for sensor errors, range, and field of regard. For the flight test environment, JADEM provides a pass-through surveillance mode that uses actual surveillance data from on-board sensors instead of modeling them. The Evaluate, Prioritize, and Declare functions are responsible for evaluating each intruder detected by the surveillance system and determining whether to provide an alert and the severity of the alert to the pilot. The Determine function is the process by which maneuver guidance is supplied to the UAS pilot and a maneuver is determined in order to resolve alerted threats. Lastly, the Command function is the process by which the pilot enters a maneuver into their ground control station, and this maneuver is transmitted through a link between the ground control station and the unmanned aircraft.

There is an array of different alerting methodologies and maneuver guidance algorithms within JADEM; those used in FT3 are discussed below.

### **Well Clear and Alerting Threshold Definitions**

The DAA system on an UAS, noted as the ownership, suggests guidance that aims to maintain "Well Clear" with intruders. The definition of Well Clear is based on three separation metrics:

1. Modified Tau, denoted by  $\tau_{mod}$  with a unit of time
2. Horizontal Miss Distance at the CPA, denoted by  $HMD$
3. Vertical Separation, denoted by  $d_h$

The first two metrics are computed from the ownship's horizontal position and velocity relative to the intruder. The third metric is the vertical distance by which the two aircraft are separated.

Modified Tau ( $\tau_{mod}$ ) is defined as:

$$\tau_{mod} = \frac{-(r^2 - DMOD^2)}{r\dot{r}}, \quad \text{for closing geometries where } r > DMOD,$$

$$\tau_{mod} = 0, \quad \text{for } r \leq DMOD,$$

$$\tau_{mod} = \inf, \quad \text{for non-closing geometries where } r > DMOD,$$

where  $r$  and  $\dot{r}$  are the horizontal range and range rate, respectively.  $DMOD$  is a distance parameter.

$HMD$  is the predicted closest horizontal distance between now and the future, if the two aircraft fly straight paths with their current velocities.

The ownship is said to have lost Well Clear with an intruder when

$$[0 \leq \tau_{mod} \leq \tau_{mod}^* \text{ and } HMD \leq HMD^*] \text{ and } [-h^* \leq d_h \leq h^*]$$

with  $\tau_{mod}^* = 35 \text{ sec}$ ,  $HMD^* = 4000 \text{ ft}$ ,  $DMOD = 4000 \text{ ft}$ , and  $h^* = 450 \text{ ft}$ . Note that  $DMOD$  is always set to equal  $HMD^*$ . Figure 1 depict a schematic diagram of the Well Clear zone.

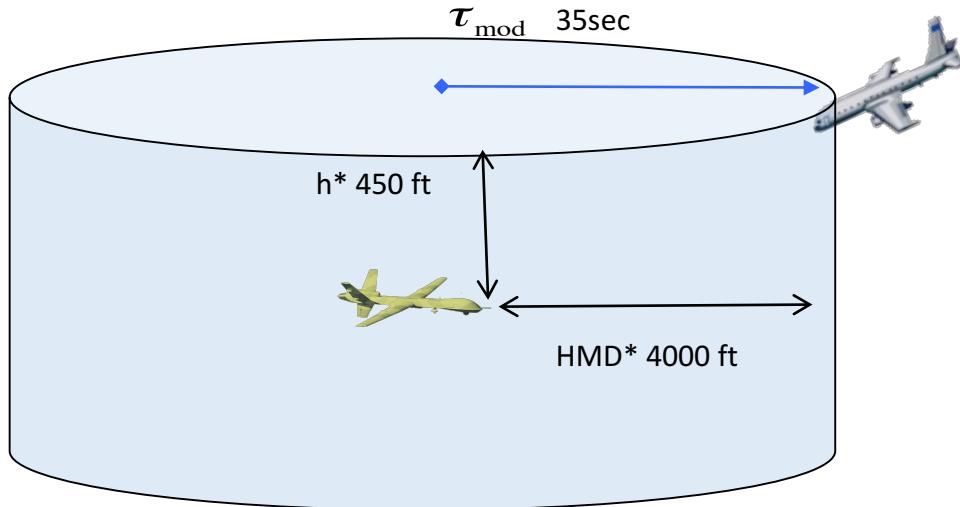


Figure 1: A schematic representation of the Well Clear zone.

Table 1: Alert types and thresholds

Alert Level	Name	Pilot Action	DAA Alert Threshold	Alert Time (Time Until Penetrating Alert Threshold)	Symbology	Aural Alert Verbiage
4	Self Separation Warning Alert	Immediate action required to avoid a well clear violation, notify ATC as soon as practicable after taking action	HMD* = 0.75 nmi h* = 450 ft $\tau_{mod}^*$ = 35 sec	25 sec (TCPA approximate: 60 sec)		"Traffic, Maneuver Now"
3	Corrective Self Separation Alert	Action to remain well clear will be necessary if the encounter does not change, coordinate with ATC to determine an appropriate maneuver	HMD* = 0.75 nmi h* = 450 ft $\tau_{mod}^*$ = 35 sec	75 sec (TCPA approximate: 110 sec)		"Traffic, Separate"
2	Preventive Self Separation Alert	Action to remain well clear will be necessary only if one or both aircraft make both a horizontal and vertical maneuver, do not climb/descend or turn into the intruder and be prepared to respond if the intruder begins climbing/descending or turning towards you. You may want to coordinate with ATC about the intentions of the intruder.	HMD* = 1.0 nmi H* = 700 ft $\tau_{mod}^*$ = 35 sec	75 sec (TCPA approximate: 110 sec)		"Traffic, Monitor"
1	Self Separation Proximate Alert	No action necessary to avoid this aircraft, but its presence should be considered when determining a resolution maneuver to avoid other aircraft.	HMD* = 1.5 nmi h* = 1200 ft $\tau_{mod}^*$ = 35s	85 sec (TCPA approximate: 120 sec)		N/A
0	None (Target)	No action necessary, There is an aircraft within your sensor range, but it is not expected to present a threat.	Within surveillance field of regard	N/A		N/A

## Autoresolver Directive Guidance Algorithm

The Autoresolver is one of many DAA maneuver guidance algorithms within the JADEM architecture. The Autoresolver originated from the Autoresolver algorithm within the Advanced Airspace Concept (Erzberger, Lauderdale, & Chu). Autoresolver was designed as a strategic separation assurance algorithm that addresses four air traffic control problems—separation conflicts, weather avoidance, arrival sequencing, and out-of-conformance—in an integrated fashion. The separation conflict resolver function of the Autoresolver was envisioned to identify efficient trajectories that resolve projected separation losses (i.e., CPA of less than 5 nmi horizontally and 1,000 ft vertically) for communication to aircraft operators or air traffic controllers.

For DAA applications, the Autoresolver algorithm was adapted for pilot-in-the-loop DAA guidance, where the goal is to avoid losses of well clear (as distinct from violations of the legal spatial separation standard). The principal attributes of the DAA problem that distinguish it from the conventional ATC separation problem are shorter look-ahead-times, lack of intent information for intruder aircraft, smaller spatial separation standards, introduction of time-based separation standards in addition to spatial, more frequent update rates, and single “ownship” point-of-view vs. centralized separation management of multiple aircraft in an airspace.

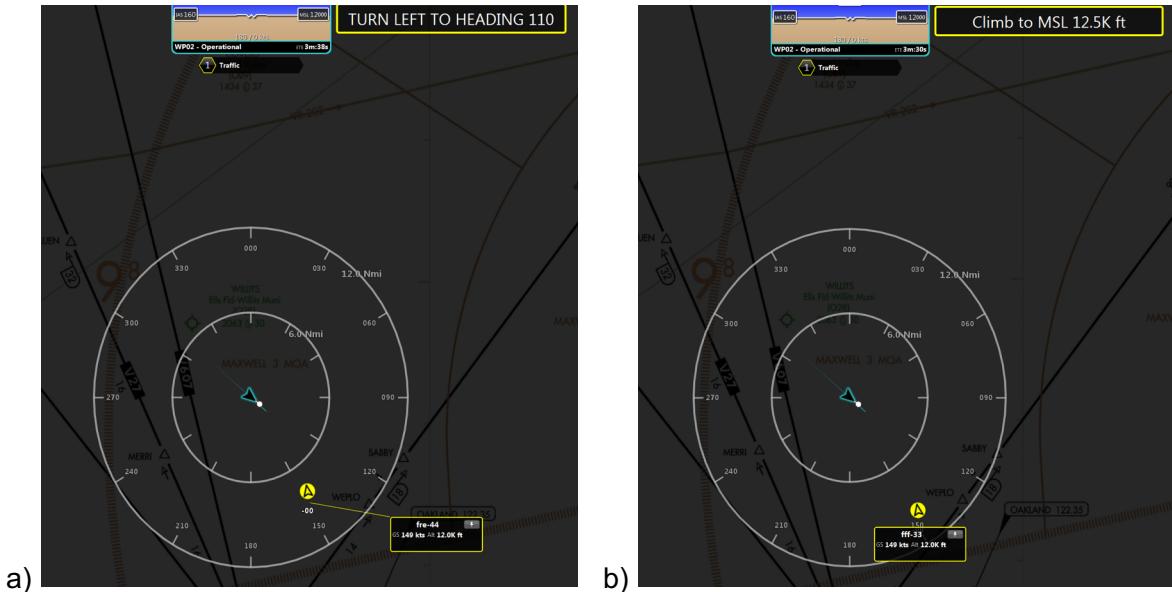


Figure 2: Autoresolver directive guidance: a) turn maneuver, b) altitude maneuver

With any DAA system, if an intruder causes a Corrective Self-Separation Alert (level 3) or a Self-Separation Warning Alert (level 4), the pilot needs to take action. For the DAA system here, JADEM, the Autoresolver computes a resolution in the form of a turn or altitude maneuver to assist the pilot in remaining well clear of the intruder. The Autoresolver was integrated with the Vigilant Spirit Control System (VSCS) ( Feitshans, Rowe, Davis, Holland, & Berger) developed by the Air Force Research Laboratory. **Figure 2a** and **Figure 2b** show examples of the Autoresolver directive guidance for a turn and an altitude maneuver, respectively.

## Flight Test Operations

This section provides a brief description of the flight test operations as it pertains to flight test data collection. A detailed description of flight test operations including test aircraft equipage, flight test procedures and actual flight test cards can be found in (Marsten, Sternberg, & Valkov).

### Test Aircraft

The UAS under test was NASA's Ikhana (NASA 870), referred to as the ownship herein. NASA's Ikhana is based on a General Atomics MQ-9 Reaper UAS but with test-specific avionics associated with the DAA and radar systems installed. The primary intruder aircraft was a Beechcraft C90 King Air owned by Honeywell (N3GC). The AFRC's T-34 (NASA 865) was also used as an intruder aircraft.

### Encounter Scenario Design

The encounter scenarios were designed to collect data to specifically address flight test objectives while abiding by the required flight safety constraints. Scenarios covered a broad range of encounter angles and vertical profiles. Figures 3 through 5 depict the encounter angles and vertical profiles for the various scenario types. A scenario nomenclature that combined the vertical profile with the encounter angle was used throughout this flight test. For example, scenario L12A designated a head-on encounter ("A") between a level ownship and a level intruder ("L12"). A summary of the scenarios, including encounter angles, speeds, etc., is provided in Appendix Table A1. The sections below describe how each scenario type supports the flight test objectives.

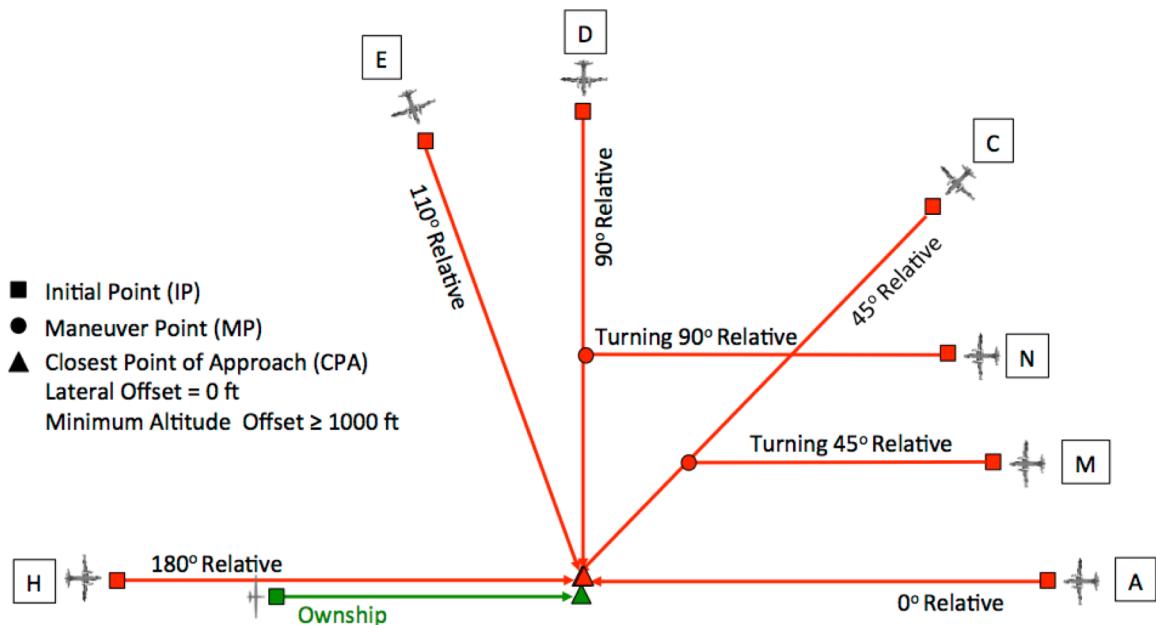


Figure 3: Pairwise encounter angles

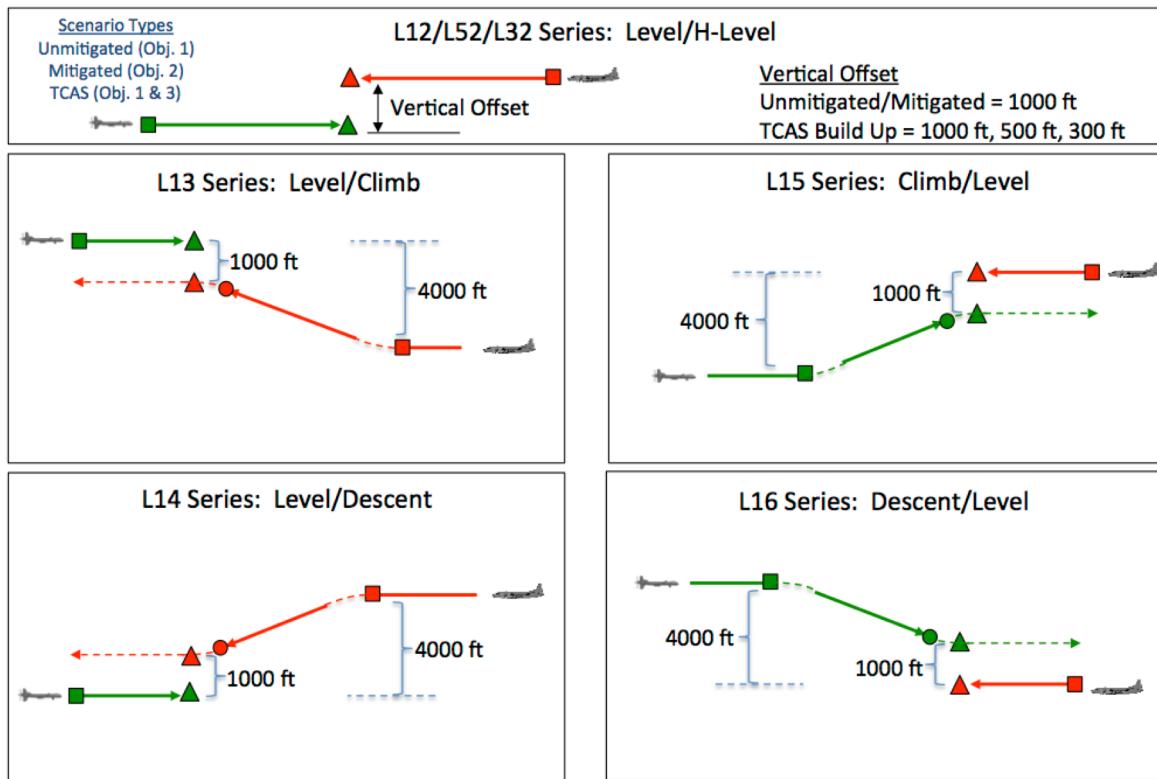


Figure 4: Vertical profiles of encounters

### ***Unmitigated and mitigated scenarios***

Objective 1 and 2 were associated with Autoresolver trajectory prediction accuracy as measured at the CPA and during a DAA maneuver as directed by the Autoresolver guidance, respectively. The encounter angles for these objectives are shown in Figure 3.

In order to collect data to measure actual separation at the CPA and analyze prediction accuracy for objective 1, scenarios were designed so that the ownship and intruder would maintain the flight condition specified in the test cards from the planned initial point (IP) to the CPA. These encounters were referred to as “unmitigated” because they were designed with the requirement that neither aircraft make any maneuver to mitigate the impending loss of well clear.

At the CPA, the planned minimum horizontal separation was zero nautical miles (nmi) and the minimum vertical separation was 1,000 feet, i.e., the primary flight safety constraint. An artificial vertical offset was applied to the Autoresolver algorithm in order to negate the actual safety offset and make the ownship and intruder appear co-altitude. The resulting Autoresolver alerts and guidance were recorded for post-flight analysis. However, the UAS pilots were instructed to ignore any Autoresolver guidance for these specific unmitigated encounters.

The same encounter angles and vertical profiles were used to collect data to address objective 2. However, for these “mitigated” encounters, the UAS pilot was asked to comply with the guidance cues provided by the DAA system (Autoresolver), thus mitigating the impending loss of well clear. Data were collected to measure how

accurately the flown trajectory adhered to the DAA-advised trajectory and to measure the corresponding separation between aircraft.

### **TCAS interoperability scenarios**

The TCAS interoperability scenarios were designed to explore the relative timing between TCAS and DAA (i.e., Autoresolver) alerts. Because TCAS is an operational system, it is not possible to fly an encounter with a vertical safety offset and artificially “trick” the TCAS system to think there is a threat, as was done for the DAA tests described previously. Therefore, these TCAS encounters had to be flown with actual separations that would trigger a TCAS alert. These encounters were flown unmitigated, with pilots ignoring Autoresolver and TCAS advisories. The minimum flight safety requirements for these TCAS encounters were 0.5 nmi lateral and 300 ft vertical offsets. The TCAS encounter angles and lateral offset are shown in Figure 5. Only the level-level vertical profile was flown for these TCAS encounters (Figure 4). For safety and pilot comfort, multiple TCAS encounters were flown using a “build-down” approach that decreased vertical separation from 1,000 ft to 500 ft and then to the minimum vertical separation of 300 ft. TCAS alert was expected at 300 ft vertical separation. As mentioned earlier, TCAS interoperability scenarios were flown but were not analyzed with respect to objective 3 due to TCAS data collection and time synchronization issues. However, data from these scenarios contributed to the set of unmitigated encounters used to evaluate trajectory prediction accuracy (Objective 1).

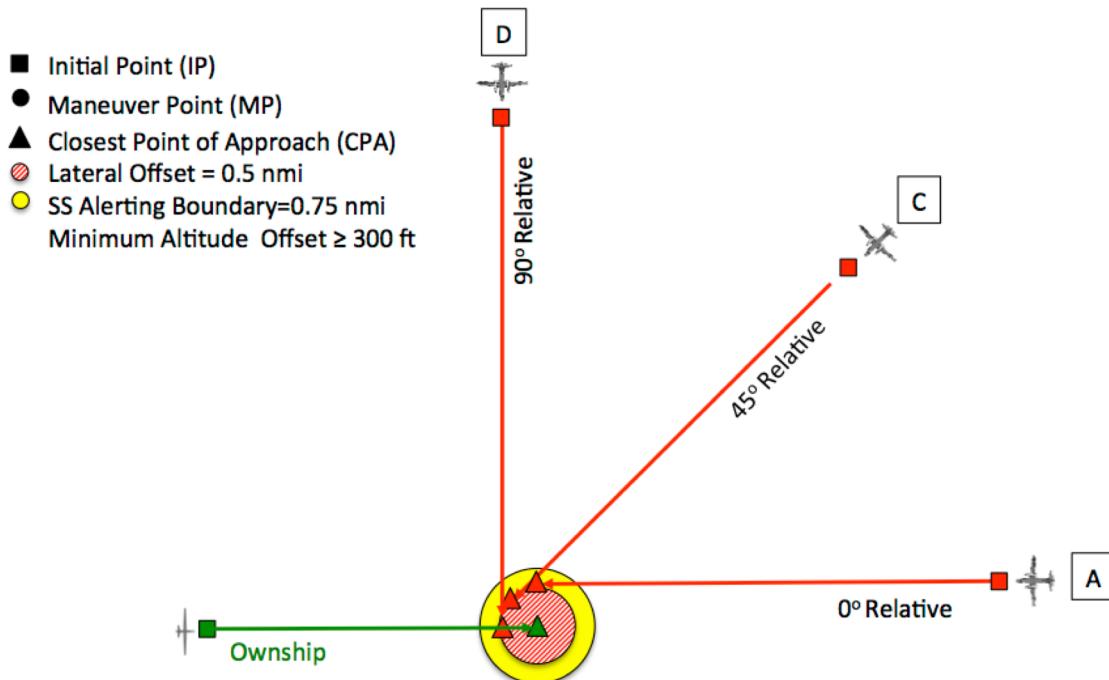


Figure 5: TCAS encounter angles

## Flight Test Data Description

Data were collected during the flight test and further processed afterwards. Figure 6 depicts a high-level diagram about the data flow and types of data recorded. The diagram is simplified to illustrate only those elements related to data collection. The dashed boxes represent processes or devices, whereas the solid boxes represent data storage. The solid arrows represent data flow in real time during the flight test, whereas the dotted arrows represent data flow in post-processing. The following sections describe data collection during the flight test and in post-processing, respectively.

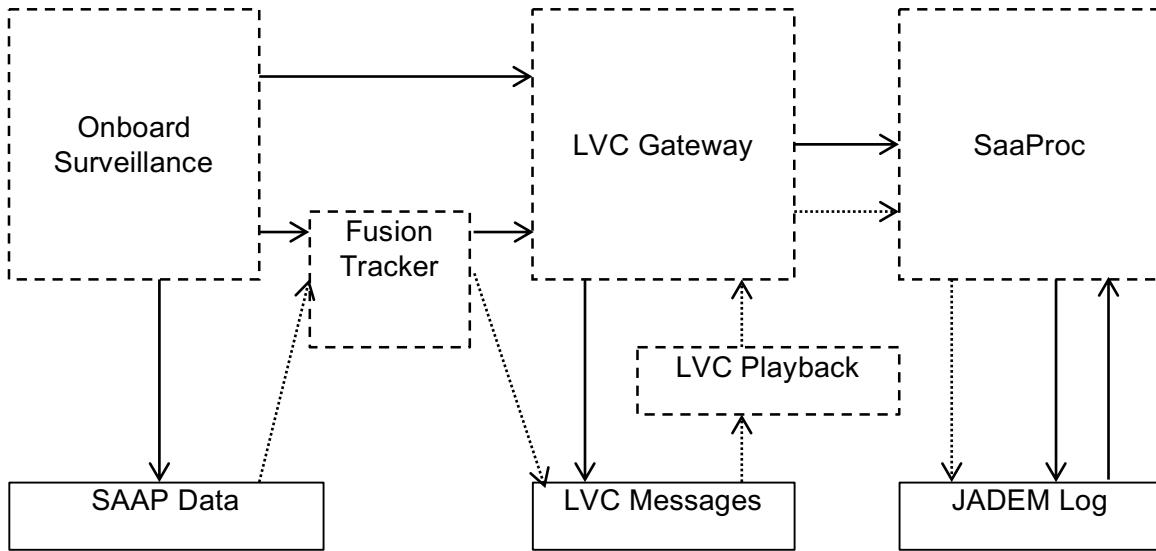


Figure 6: Data flow and types of flight test data for analysis: dashed boxes for processes or devices, solid boxes for data storage, solid arrows for real-time data flow, and dotted arrows for post-processing data flow.

## Overview of Flight Test Data Collection System

Surveillance data from individual sensors onboard the ownship were recorded and stored by the Sense-and-Avoid Processor (SAAP). The onboard surveillance sensors included the airborne radar, TCAS, and ADS-B. SAAP also stored the ownship's position and velocity states measured by the ownship's GPS and other sensors. The surveillance data were sent to an onboard fusion tracker from Honeywell that performed track coordinate transformation, filtering, association, and fusion. Using the SAAP control panel, the UAS flight crew could selectively turn on inputs from specific sensors that were sent to the fusion tracker or bypass the tracker entirely. The processed data were then transmitted as flight state messages to the ground. The content of these messages included the ownship and the intruders' ID, time, position, and ground velocity. The ownship's state also contained its velocity with respect to the wind. This allowed estimates of winds experienced by the ownship. The flight state message had an update rate of 1 hertz.

The Live, Virtual, and Constructive (LVC) Gateway (middle-top entity in Figure 6) (Soler, Jovic, & Murphy) received flight state messages, time-stamped them, and forwarded

them to the SaaProc, a real-time message-handling layer that wrapped JADEM. The LVC Gateway logged these flight state messages in LVC message files (bottom middle entity). The LVC Gateway also received guidance messages from SaaProc and forwarded them to the display system, VSCS (not shown).

The SaaProc (top-right box in Figure 6) read the flight states and computed alerts and guidance maneuvers accordingly. It sent the guidance maneuvers as messages to the LVC Gateway, which forwarded them to VSCS. SaaProc recorded the flight states, alerts, and maneuvers as log files.

### **Data Collection Anomalies During Flight**

One of the primary functions of Honeywell's fusion tracker was to associate track data received from the SAAP for multiple targets and surveillance sensor sources with the appropriate intruder aircraft. For aircraft equipped with ADS-B, an ICAO callsign was available, and it was associated with the intruder target. Unfortunately, this callsign-to-target association logic within the fusion tracker did not work properly with the input SAAP data during flight test data collection for the Autoresolver system. This problem caused the callsign for a given track to constantly change and, thus, appeared as multiple targets to the Autoresolver system. Consequently, the SAAP control panel was configured with the fusion tracker deselected (i.e., bypassed) and ADS-B selected as the only surveillance sensor. The result was Autoresolver received track targets from ADS-B surveillance only but with the consistently correct callsign associated with it. However, flight data from all other surveillance sources were still recorded by the SAAP and were processed post-flight.

Mitigated scenarios flown on June 22 were impacted by unexpectedly noisy vertical speed and altitude intruder state data. As described above, mitigated scenarios required the UAS pilot to execute maneuvers based on Autoresolver guidance. The noisy state data caused the Autoresolver guidance to be unstable, thus confusing the pilots. The resulting scenarios flown that day were considered unsuccessful and were repeated on July 22. Mitigated scenarios flown on July 22 benefitted from the implementation of a Kalman filter to the Autoresolver system. This Kalman filter smoothed the vertical speed and altitude measurements, thus stabilizing the trajectory predictions and resulting Autoresolver guidance cues.

### **Post Processed Flight Data**

Plans were made prior to flight for Honeywell to post-process and extract individual surveillance sensor data from SAAP data recorded during flight. These data were reformatted into LVC flight state messages so that they could be replayed through Autoresolver for additional analysis. Four sets of LVC flight state messages were created by Honeywell by running the SAAP data through the fusion tracker. Each set was characterized by its surveillance sensor type(s):

- Airborne radar only
- TCAS mode S only
- ADS-B only
- Airborne radar, TCAS mode S, and ADS-B together

The tracker output was converted to flight state messages that complied with the LVC Gateway message format.

The ownship flight states in the LVC message files from Honeywell contained the ownship's ground speed and ground course, but lacked the ownship's true airspeed and true course. While this did not affect the alert detection, it did affect the computation of maneuver guidance. These ownship state messages were therefore replaced with those generated during the flight test that contained both sets of speeds and courses.

An LVC playback tool was used to read the LVC message files and to send them to the LVC Gateway while preserving the relative timing of the messages. The Gateway forwarded the messages to SaaProc for alert and guidance computation. See the dotted arrows in Figure 6 for all the post-processing data flows.

A Kalman filter of the intruder's altitude and vertical speed inside JADEM was applied to investigate the effect of smoothed vertical speeds on the alert stability. Both the measured altitude and vertical speed were used as input. The parameters of the filter were tuned so as to reduce the vertical speed noise without resulting in an altitude lag of more than 10 seconds. For each of the four sets of surveillance data, both a SaaProc with a filter and a SaaProc without a filter were run to collect alert and guidance data.

## Additional Computed Data

For unmitigated encounters, a suite of data analysis tools were developed to compute the following data:

- Predicted trajectories at every time step within an encounter's time window-- These data were computed by JADEM but not logged due to performance impacts. The predicted trajectories created by this analysis tool were identical to those generated by JADEM in real time.
- Horizontal and vertical prediction errors at the ownship and intruder's actual time of CPA, as a function of the look-ahead time prior to the actual CPA
- The alert metrics  $\tau_{\text{mod}}$ ,  $h$ , and  $HMD$  along predicted trajectories. These metrics were used to investigate alert discontinuity and identify which separation metric went out of its threshold.

Analysis of mitigated encounters focused on comparing actual trajectories to predicted ones. Special care was taken to remove two types of intent errors. The first intent error was a turn angle error due to a wind-related discrepancy between VSCS and JADEM in translating the turn advisory. As a result, the actual maneuver executed by the ownship deviated from the Autoresolver advisory by 5 to 10 degrees in ground course. The second intent error was from the unknown time an advisory was actually executed. A time must be picked for each encounter in order to select appropriately a predicted trajectory for trajectory comparison.

To remove these two types of intent errors from the comparison, the ownship's actual trajectory was used to derive a turn angle and a turn start time. The turn start time was taken as the time the predicted trajectory was created. The re-creation of predicted trajectories called the JADEM library functions using the same turn modeling schemes configured for Autoresolver.

## Analysis and Results

A significant amount of flight test data were collected during flight which, in turn, were further processed post-flight. This report only presents analysis results completed in support of the UAS in the NAS project FT3 data review conducted on October 20, 2015. Due to time constraints and, to a lesser extent, TCAS data collection and time synchronization issues, objective 3 was not evaluated in this analysis. However, data collected during scenarios that were originally designed to meet the TCAS interoperability objective were used to support other objectives. Table 2 summarizes the encounter data used in support of this analysis.

Table 2: Summary of encounters performed on the four flight test days. Encounters in white cells represent data analyzed in this report.

Encounters	Unmitigated		Mitigated
	DAA	TCAS	
6/17/2015	6	9	
6/18/2015	20	2	
6/22/2015			20
7/22/2015	5		12

While the encounters were planned to have the intruder well within the alert threshold to ensure stable and consistent alerts, execution errors brought some encounters closer to or over the edge of the alert threshold. Figure 7 shows the actual horizontal and vertical separations of the ownship and the intruder at the CPA for each of the 26 unmitigated,

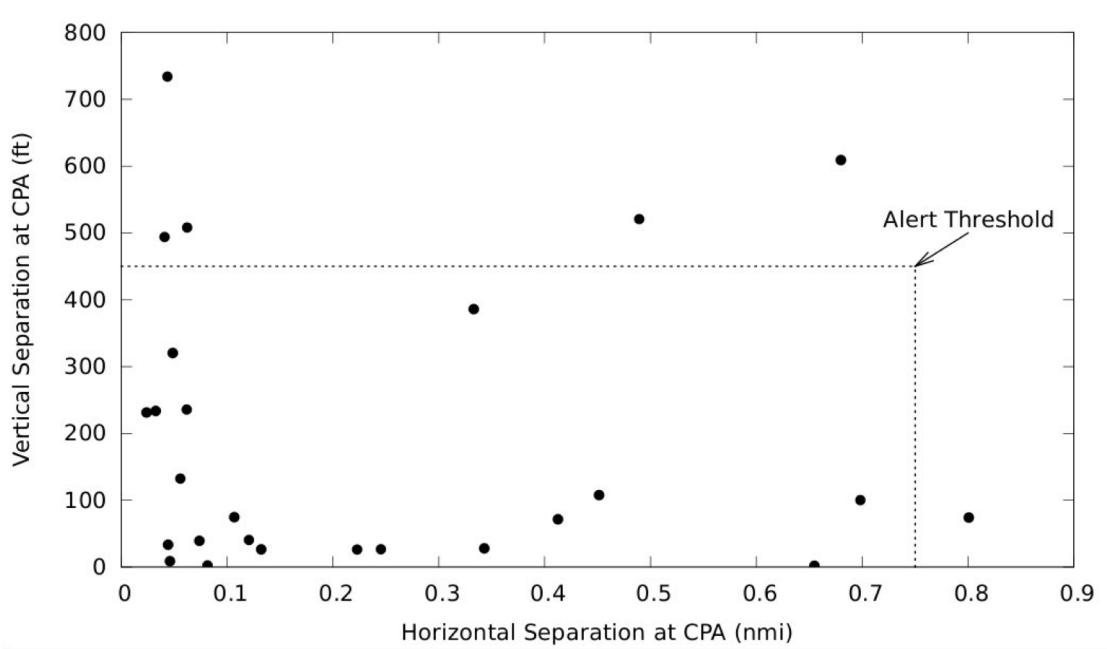


Figure 7. Separation at the actual CPA for each DAA encounter

DAA (non-TCAS) encounters analyzed. Most of the encounters resulted in small separation distances at their CPAs. A few resulted in separation values close to the alert threshold represented by the dotted lines. In fact, six encounters resulted in values outside the alert threshold zone. For encounters close to or outside the alert threshold, the alert stability was expected to be more sensitive to surveillance noise. Note that the computation of the CPA here utilizes both horizontal and vertical components of the separation distance.

### Predicted Separation Error

The unmitigated encounter served as the basis for predicted separation error analysis. As described above, unmitigated encounters were those encounters in which both the ownship and the intruder flew in a stable manner from the IP to the planned CPA. Actual separation can then be measured at select points of interest (e.g., the CPA and at the point of first loss off well clear) and be compared to the predicted separation at various look-ahead times prior to the point of interest.

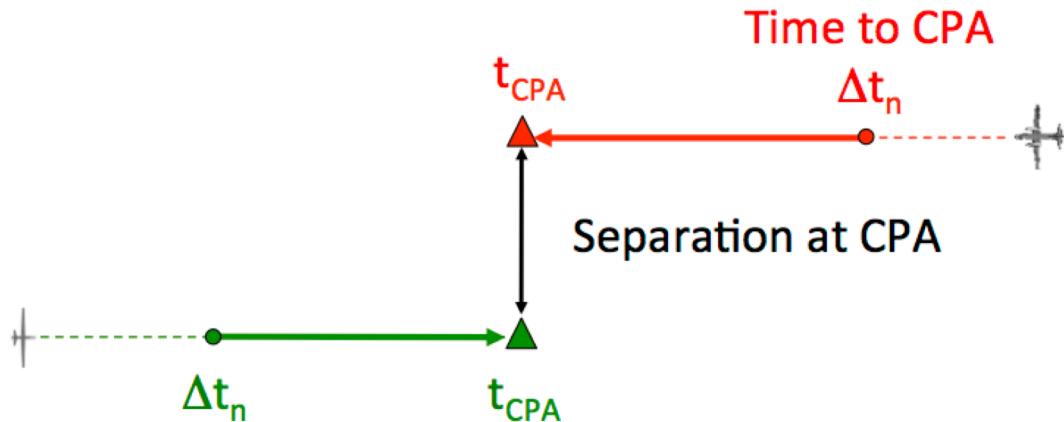


Figure 8. Predicted Separation Error Analysis

A notional example of separation analysis methodology using CPA as the point of interest is shown in Figure 8. Actual separation was measured at the time of CPA,  $t_{CPA}$ , using ADS-B surveillance data as the reference. Trajectories were generated with each individual surveillance sensor as input. Predicted separation at the CPA is then calculated at each look-ahead time prior to the CPA,  $\Delta t_n$ . Separation error at the CPA for each look-ahead time is defined as the difference between the predicted separation at  $\Delta t_n$  and the actual ADS-B separation at  $t_{CPA}$ . Similar predicted separation errors were calculated for the point of first loss of well clear instead of CPA. Figure 9 shows the predicted horizontal and vertical separation errors at the CPA for a single encounter (L32C) with trajectories generated with ADS-B and radar data.

For encounter L32C, Figure 9 shows predicted vertical separation error for radar-based trajectories was significantly larger than errors for ADS-B based trajectories. As expected, predicted error decreases as time-to-CPA, or look-ahead time, decreases. The predicted horizontal separation errors were similar in magnitude for ADS-B and radar-based trajectories, however radar data did appear noisier.

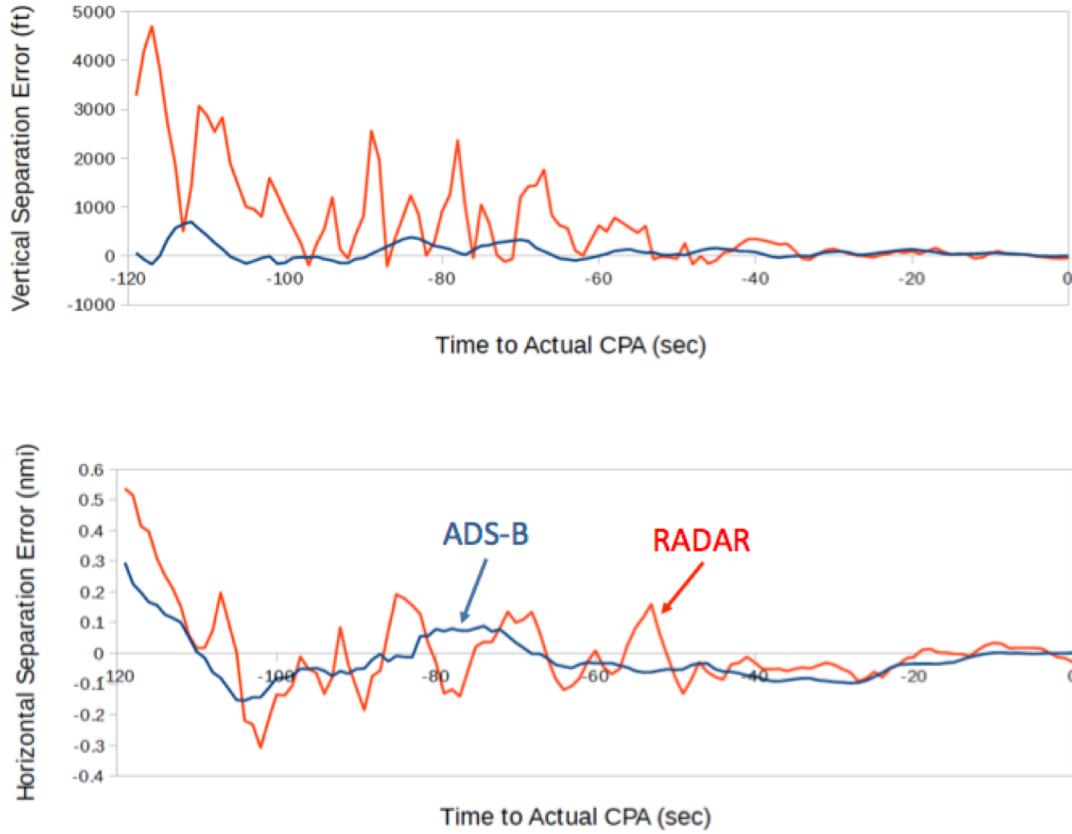


Figure 9. Predicted separation error for encounter L32C

A box-whisker format was used to plot the aggregate prediction errors for multiple encounters. Figure 10 shows an example of a box-whisker plot for the aggregate CPA prediction errors for ADS-B data.

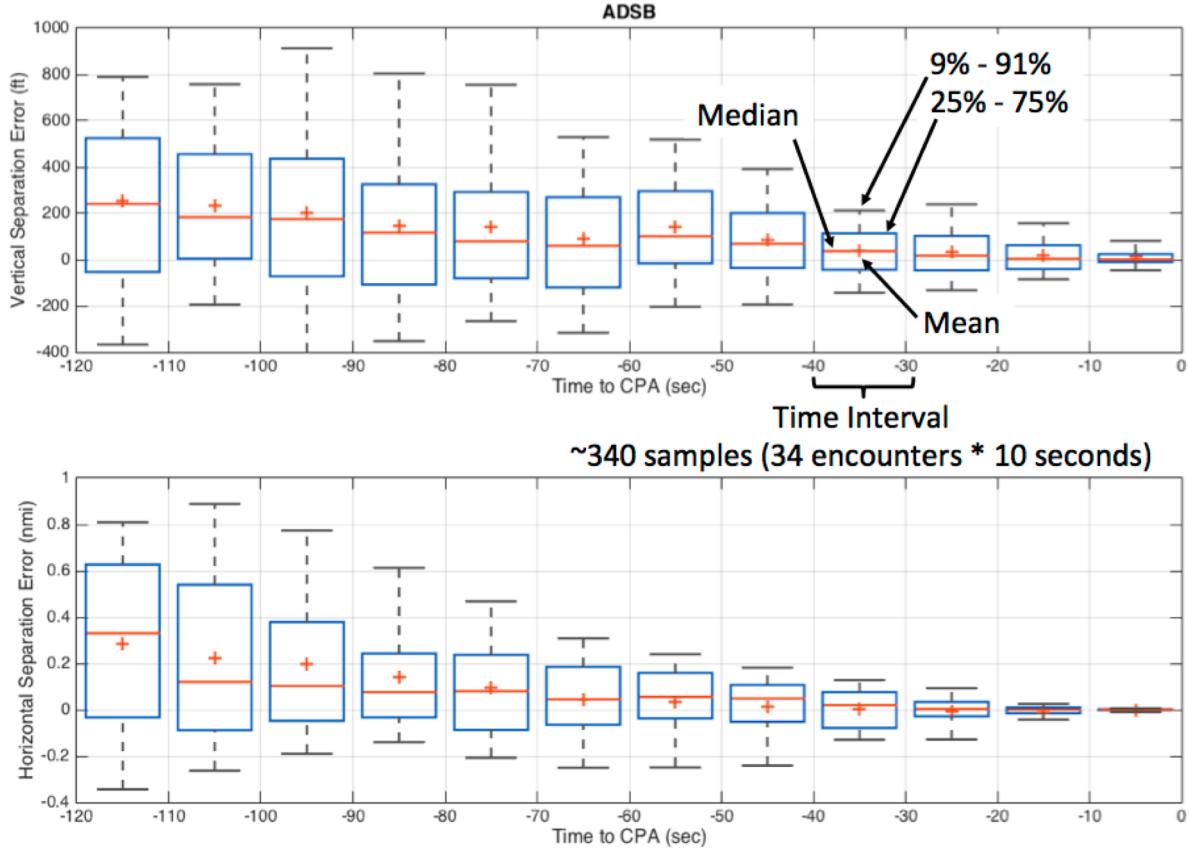


Figure 10. Aggregate CPA prediction error for ADS-B surveillance data

Each box-whisker combination is an aggregate of all samples from the set of 34 unmitigated DAA and TCAS encounters (excluding turning scenarios L12M/N) over a look-ahead time interval (i.e., time-to-CPA) of 10 seconds. The mean of those samples is represented with a red "+" and the median is represented with a red line. The blue box represents the range of samples between the 25<sup>th</sup> percentile and the 75<sup>th</sup> percentile, while the whiskers encompass data between 9<sup>th</sup> and 91<sup>st</sup> percentile.

Figure 11 shows a side-by-side comparison of CPA prediction error for ADS-B and radar-based trajectories for the same set of 34 unmitigated encounters. As shown earlier for the single encounter in Figure 9, the aggregate predicted vertical separation error for radar-based trajectories was significantly greater than for ADS-B based trajectories. Mean predicted vertical separation error for radar trajectories was nearly 3000 feet for look-ahead times between 110 and 120 seconds, while mean ADS-B errors were less than 500 feet. After further analysis, the increased predicted vertical separation error for radar data was attributed to increased noise (i.e., uncertainty) in the radar-derived vertical speed measurements when compared to ADS-B measurements. This difference in vertical speed measurements of the intruder aircraft is quantified in Figure 12, which shows a plot of the vertical speed measurements for all unmitigated encounters when the intruder was flying level. For the 25<sup>th</sup>-75<sup>th</sup> percentile data, the vertical speeds for ADS-B data ranged from -151 ft to 164 ft, compared to -729 ft to 881 ft for the radar data from the same encounters. Additional discussion and analysis with respect to the effect of vertical separation error on alerting are provided later in this report.

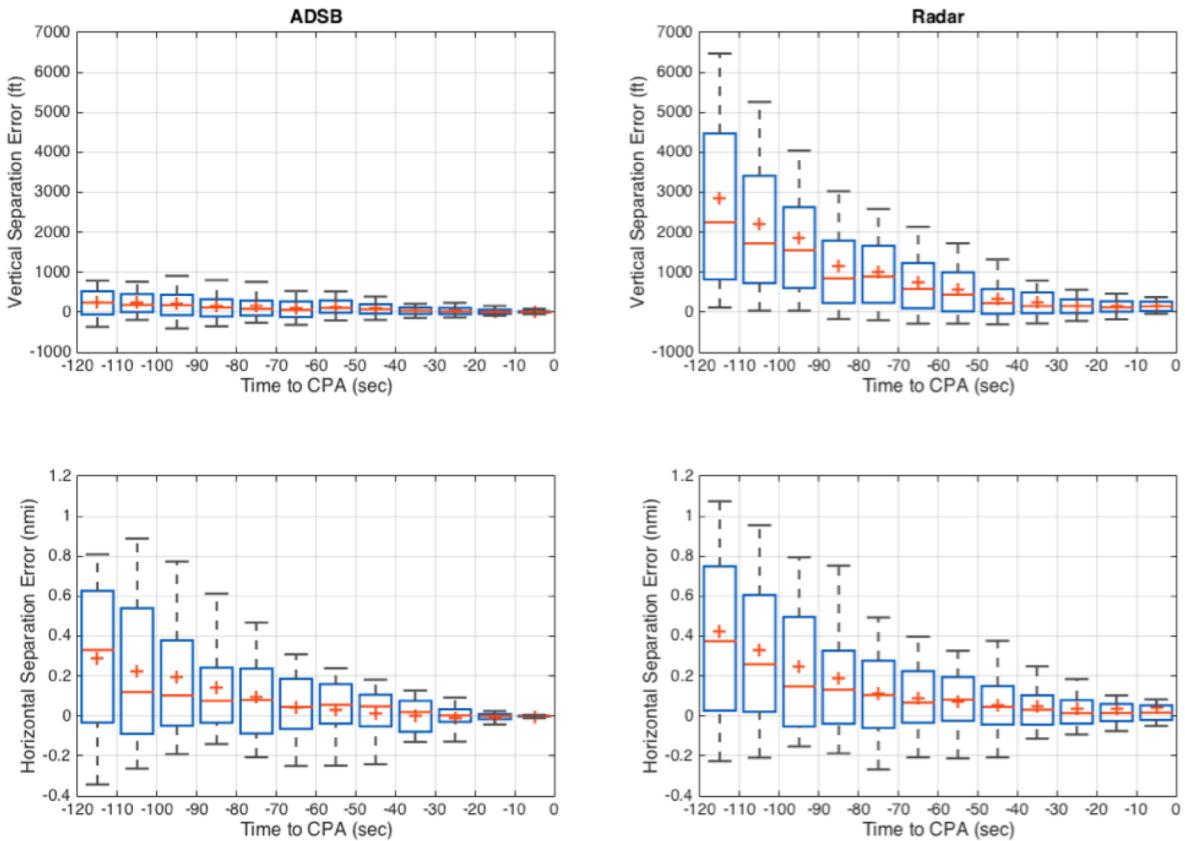


Figure 11. Comparison of aggregate CPA prediction error for ADS-B and RADAR

The magnitude of the predicted horizontal separation error is similar for both ADS-B and radar-derived trajectories. It should be noted that overall trajectory prediction errors include other sources of error other than surveillance sensor type such as wind error, real variation in aircraft speed and altitude, etc. Consequently, known difference in ADS-B and radar sensor performance, especially in horizontal position, may not be fully realized due to the predominance of other real-world errors.

Predicted separation error analysis was also performed using the point of first loss of well clear, referred to simply as "first loss" herein. This analysis was similar to the CPA analysis described in

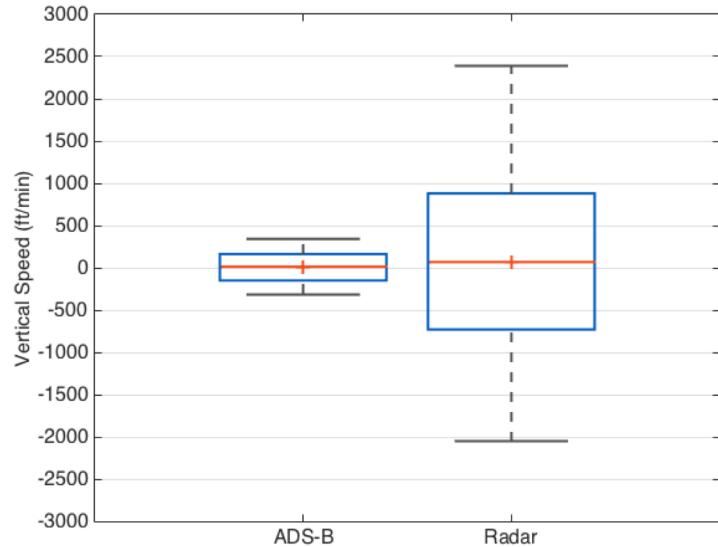


Figure 12. Vertical speed comparison between ADS-B and Radar.

Figure 8, except the actual first loss DAA alert parameters described in Table 1 (alert level 3 and 4) were measured. The first loss parameters were 0.75 nautical mile horizontal miss distance, 450 feet minimum vertical separation, and a modified tau (i.e., horizontal closing rate) of 35 seconds. Figure 13 shows a comparison of first loss prediction error between ADS-B and radar-derived trajectories. Similar to the CPA prediction error analysis, radar-derived trajectories showed notably more predicted vertical separation error than did ADS-B-derived trajectories. Note that the point of first loss occurs earlier than the CPA. Predicted horizontal miss distance and modified tau errors for both ADS-B and radar trajectories were more comparable.

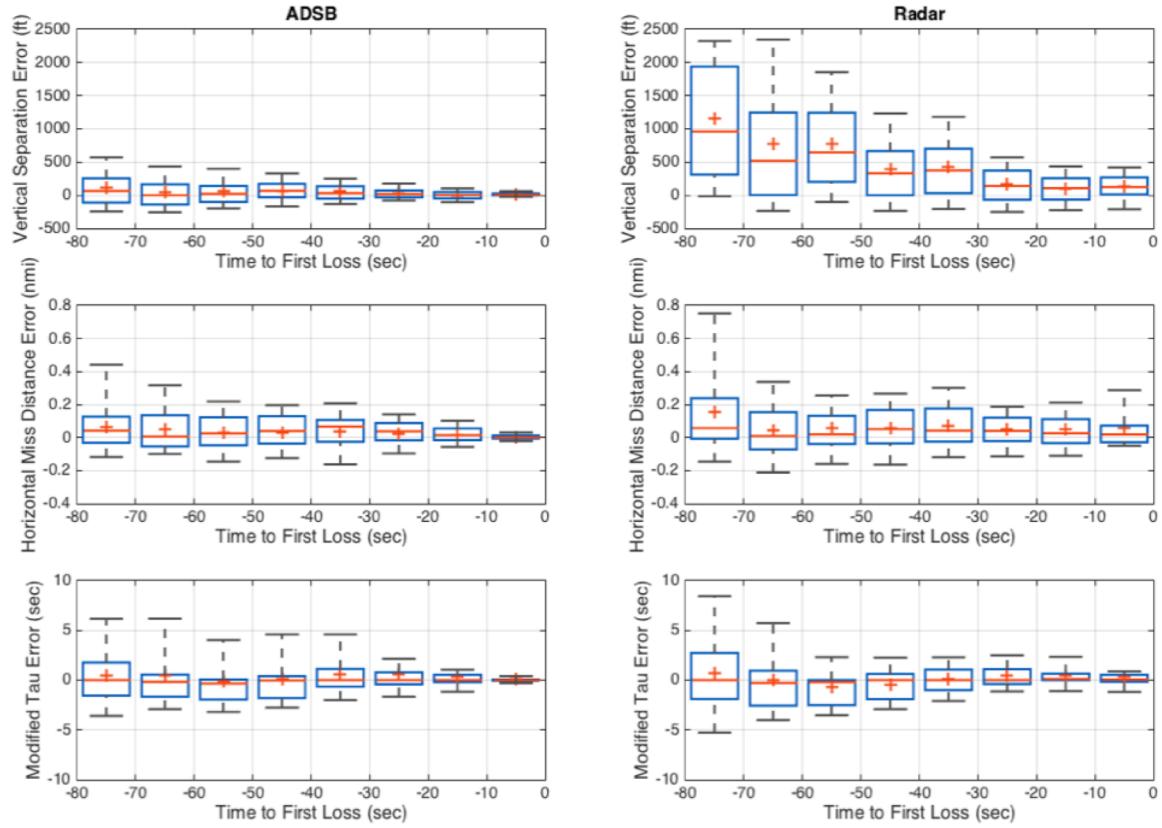


Figure 13. Comparison of aggregate first loss prediction error for ADS-B and Radar

Bearing measurements (i.e., horizontal position) from a Mode S transponder are known to be less accurate than either ADS-B or radar. This difference in horizontal position accuracy between ADS-B and Mode S can be measured in terms of trajectory prediction accuracy. Figure 14 shows a comparison of first-loss prediction error between ADS-B and Mode S derived trajectories. Predicted vertical separation errors were similar for both ADS-B and Mode S, but HMD and modified Tau errors were larger for Mode S derived trajectories.

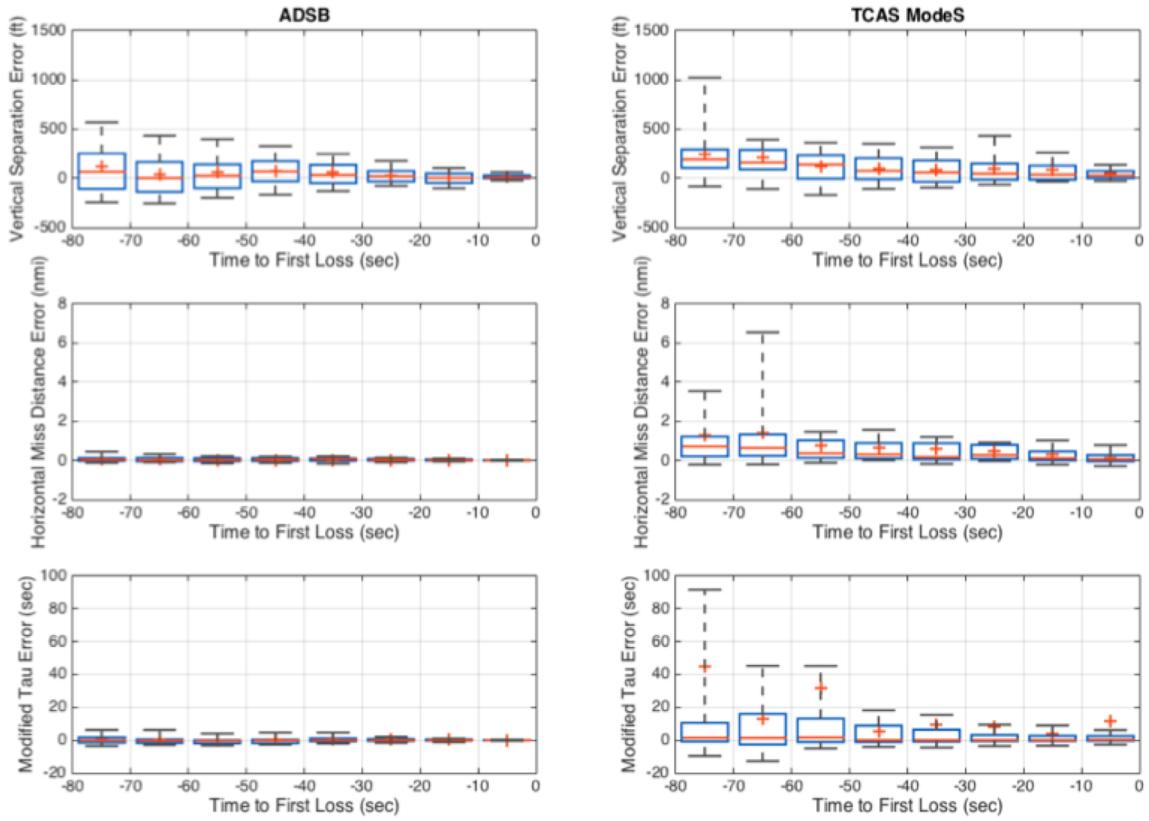


Figure 14. Comparison of aggregate first loss prediction error for ADS-B and TCAS Mode S

## Alerting Analysis

The predicted separation analysis described above provides insight into the performance of the DAA system's trajectory modeling. However, the operational impact of these trajectory prediction errors may be better assessed by analyzing DAA alerting performance. In general, good alerting performance is characterized by stable and consistent alerts with a minimal number of false alerts. Figure 15 is an example from an actual encounter of the various levels of alerting as a function of the alerting parameters. There are four alert levels: proximate, preventative, corrective, and warning (Table 1). Pilot action is only required for corrective (level 3) and warning (level 4) alerts and, therefore, will be the focus of the alerting analysis. Each symbol on a given plot indicates an alert for a given update rate (1 Hz). The line connecting multiple alerts signifies alerts for consecutive updates with the blue line connecting only consecutive corrective and/or warning alerts.

An alerting “gap” between consecutive corrective and/or warning alerts of approximately 20 seconds is highlighted in the upper right plot in Figure 15. In this encounter, for example, a corrective alert was displayed to the pilot for 4 seconds at around elapsed time of 70 seconds. Corrective alerts require pilot action. This initial series of corrective alerts is followed by approximately 20 seconds of lower severity proximate and preventative alerts which are advisory and do not require pilot action. This gap of 20 seconds is then followed by a consistent series of corrective and warning alerts. Alerting gaps such as this are an indication of alerting instability and are considered operationally

undesirable because they may lead a pilot to take premature or unnecessary action. Moreover, alerting instability may degrade pilot trust in the system.

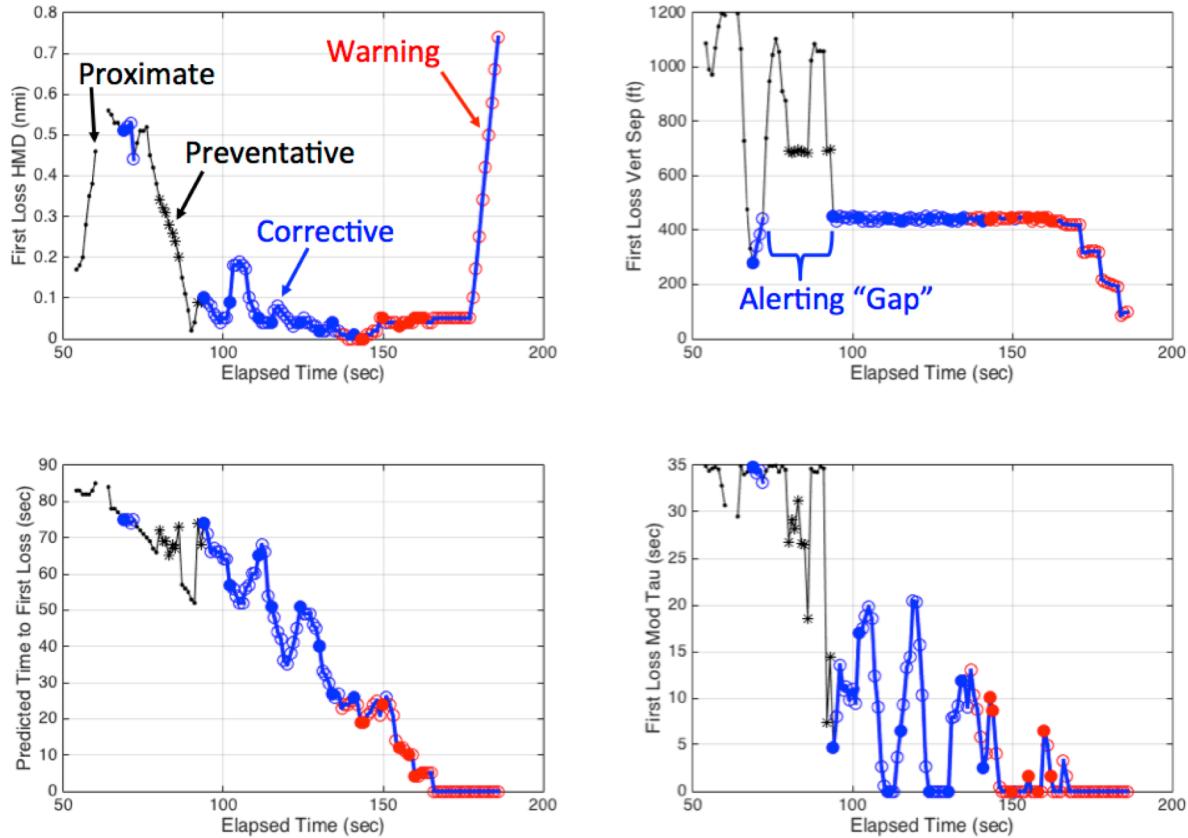


Figure 15. Example of alerting as a function of alert parameters

An aggregate metric was developed to measure the frequency and cause of these alerting gaps across all scenarios. Once an alerting gap was identified, each update within the gap was analyzed to determine which alerting parameter did not meet the corrective or warning criteria (Table 1). In the alert gap example shown in Figure 15, the first update following the initial four seconds of corrective warning had a predicted vertical separation at first loss of 738 feet. This exceeded the vertical separation criteria for corrective/warning alerts of 450 feet by 288 feet, thus resulting in the gap.

Figure 16 shows the aggregate alert gap analysis results for ADS-B and radar-derived trajectories. The upper bar charts show the number of times a given alert parameter went out of bounds to cause an alerting gap. For those instances where multiple parameters went out of bounds during a given update, the parameter that exceeded the alerting criteria by the largest percentage was plotted as the primary cause of the alert gap.

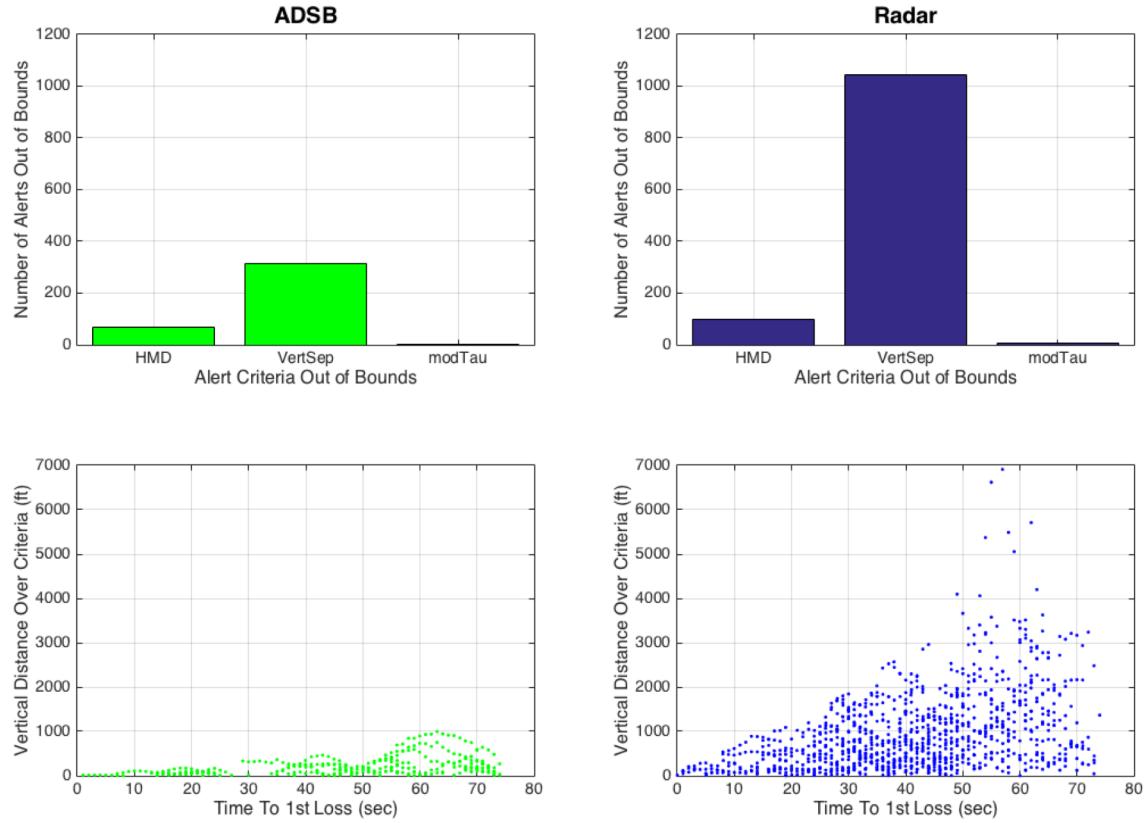


Figure 16. Alert gap analysis for ADS-B and Radar-derived trajectories

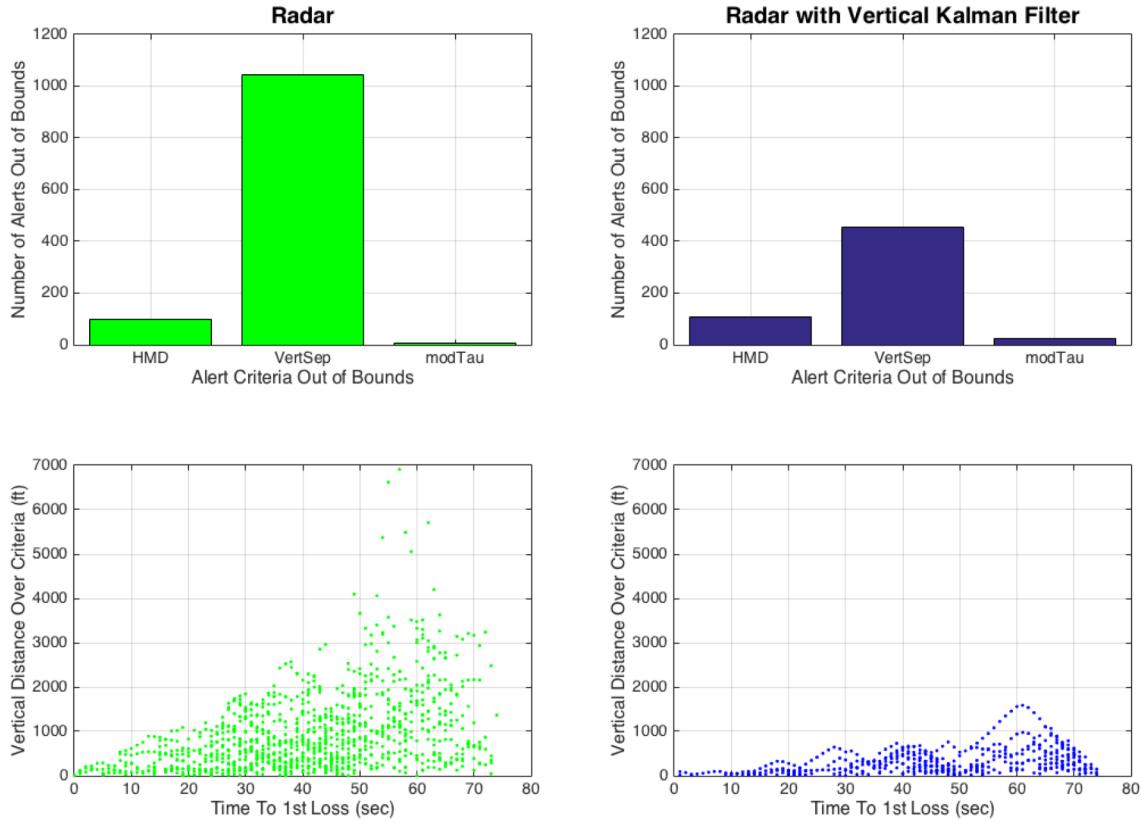


Figure 17. Alert gap analysis for Radar surveillance with and without Kalman filter

The results in Figure 16 show the alerts derived from radar-based trajectories resulted in more alerting gaps than alerts derived from ADS-B based trajectories. Moreover, exceeding vertical separation criteria was the most common cause of alerting gaps, i.e., instability. Vertical separation criteria went out of bounds 1042 times with radar-based alerts compared to 313 times for ADS-B alerts over the same set of encounters. The lower set of plots in Figure 16 shows the magnitude of the vertical separation criteria exceedance as a function of time to first loss. The ADS-B instances exceeded the 450 feet vertical separation criteria by less than 1000 feet whereas radar exceedance was as much as 6908 ft. This was due to the significantly noisier altitude and vertical speed measurements from the radar.

Alerting performance with radar data can be improved with the application of Kalman filters to the altitude and vertical speed measurements. With the method described earlier, radar data from these encounters were replayed through the DAA system via Kalman filters applied to altitude and vertical speed. The resulting alerting performance is shown in Figure 17.

Alerting performance with radar surveillance was significantly improved with the application of Kalman filters. The instances of out-of-bound vertical separation were reduced from 1042 to 455. The corresponding maximum vertical separation criteria exceedance was reduced from 6908 feet to 1588 feet. When compared to the alerting performance of ADS-B, Figure 18 shows the application of Kalman filters to the radar surveillance resulted in comparable alerting performance.

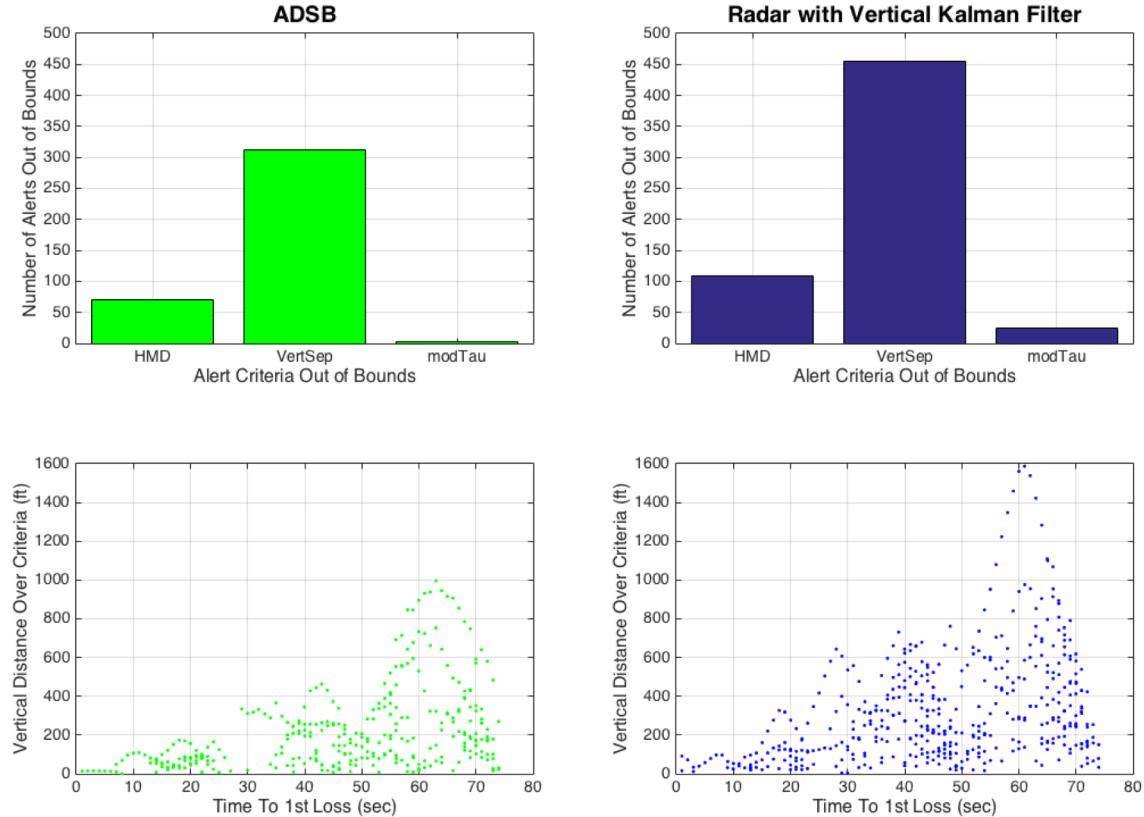


Figure 18. Alert gap analysis for ADS-B and Radar with Kalman filter

The application of Kalman filters can introduce potentially unacceptable amounts of lag to the surveillance data. Although surveillance data lag was not rigorously analyzed in this effort, researchers were cognizant of its effect while tuning the Kalman filter parameters and deemed the resulting lag qualitatively acceptable.

A similar alerting performance analysis was performed with trajectories derived from TCAS Mode S surveillance data which were shown to be less accurate horizontally than ADS-B (Figure 14). This difference can be seen in the alerting performance shown in Figure 19. Unlike the ADS-B and Radar alerting performance where vertical separation was the most critical alerting parameter, horizontal miss distance (HMD) was the most critical alerting parameter with Mode S surveillance. The HMD alerting criteria went out of bounds 247 times compared to only 36 vertical separation criteria out-of-bounds instances with Mode S surveillance. For the same set of encounters with ADS-B surveillance, instances of out-of-bound HMD and vertical separation were 70 and 313, respectively. The lower plots in Figure 19 show the corresponding magnitude of the

HMD alert criteria exceedance. For ADS-B, the HMD exceedance was no more than 0.3 nmi, whereas HMD exceedance for Mode S was as much as 1.5 nmi.

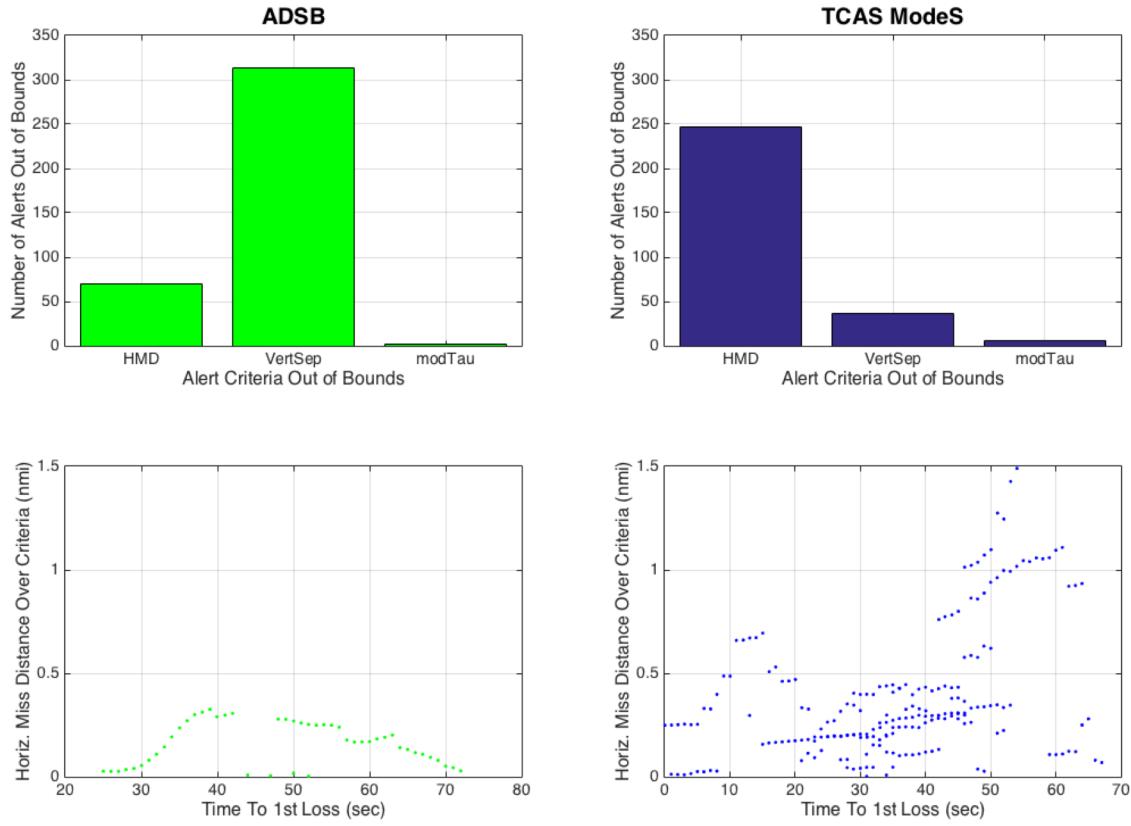


Figure 19. Alert gap analysis for ADS-B and TCAS Mode S

### Turn Prediction Accuracy Analysis

Turn prediction accuracy was assessed by analyzing data from the 12 mitigated encounters from the flight on July 22, 2015. During these mitigated encounters, pilots performed DAA maneuver guidance provided by Autoresolver. Although Autoresolver was able to calculate both turn and altitude maneuvers, only turn maneuver guidance occurred during this set of mitigated encounters.

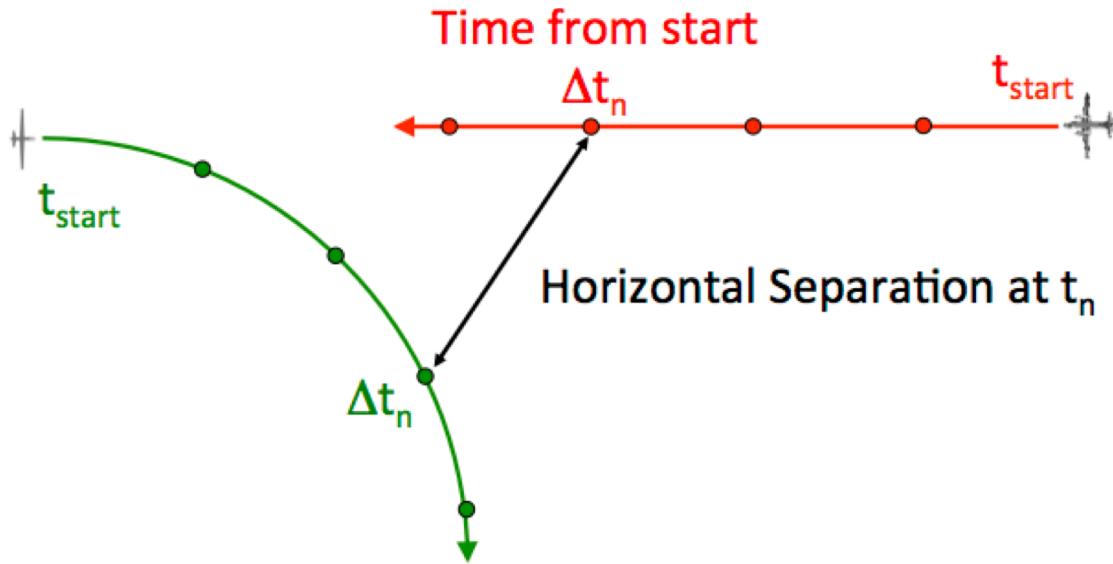


Figure 20. Turn prediction accuracy analysis methodology

The methodology used to analyze turn prediction accuracy is shown in Figure 20. For each encounter, the ownship track data was analyzed to determine the start time of the turn maneuver,  $t_{start}$ . Generally, this meant the ground course began to change steadily following Autoresolver guidance. The trajectories of the ownship and the intruder at  $t_{start}$  were then saved as the reference predictions. Predicted horizontal separation error at various time increments after the start of each turn,  $\Delta t_n$ , was calculated by subtracting the actual measured separation (ADS-B) from the predicted separation for the same time,  $\Delta t_n$ . Turn prediction accuracy measured in terms of predicted horizontal separation error for each mitigated encounter is shown in Figure 21.

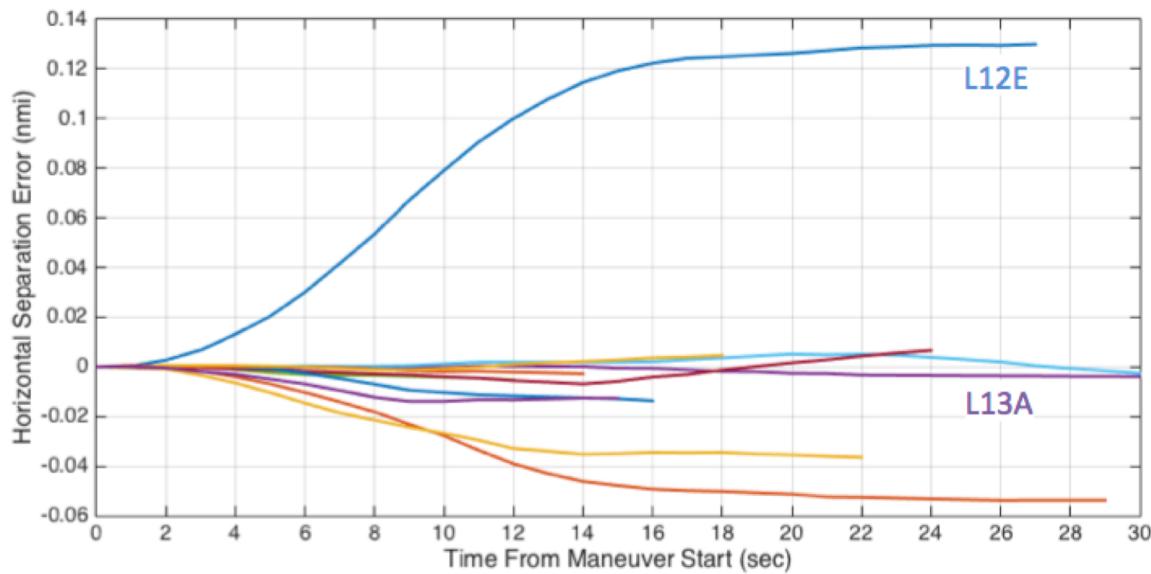


Figure 21. Turn prediction accuracy

A majority of the turns had an absolute maximum horizontal separation error of less than 0.014 nmi. However, there were three outliers that exhibited as much as 0.13 nmi of error. In order to understand the difference between a poor turn prediction such as encounter L12E and a low-error turn prediction such as encounter L13A, the actual and predicted turn rates of each of the two encounters were plotted in Figure 22. For encounter L12E, the actual turn appears to have been initiated at a slow rate, perhaps inadvertently, before being completed at the prescribed standard rate of 3 degrees per second. Although the increased error could be attributed to incorrectly selecting the turn start time, encounter L12E does serve to illustrate the impact of pilot response delay on horizontal separation.

Encounter L13A did not exhibit a similar slow or delayed turn initiation as seen with encounter L12E. The relatively small error is attributable to the difference between the computed turn rate of 2.6 degrees per second and the actual turn rate which varied up to about 3.1 degrees per second.

At the time of this flight test, Autoresolver modeled turning DAA maneuvers with a constant bank angle of 20 degrees. As a result, there is a difference in predicted turn rates for encounters L12E and L13A. Currently, the DAA MOPS specifies maneuvers at a standard rate turn of 3 degrees per second or, in some situations when a turn rate of 3 degrees is infeasible, a half rate turn at 1.5 degrees per second.

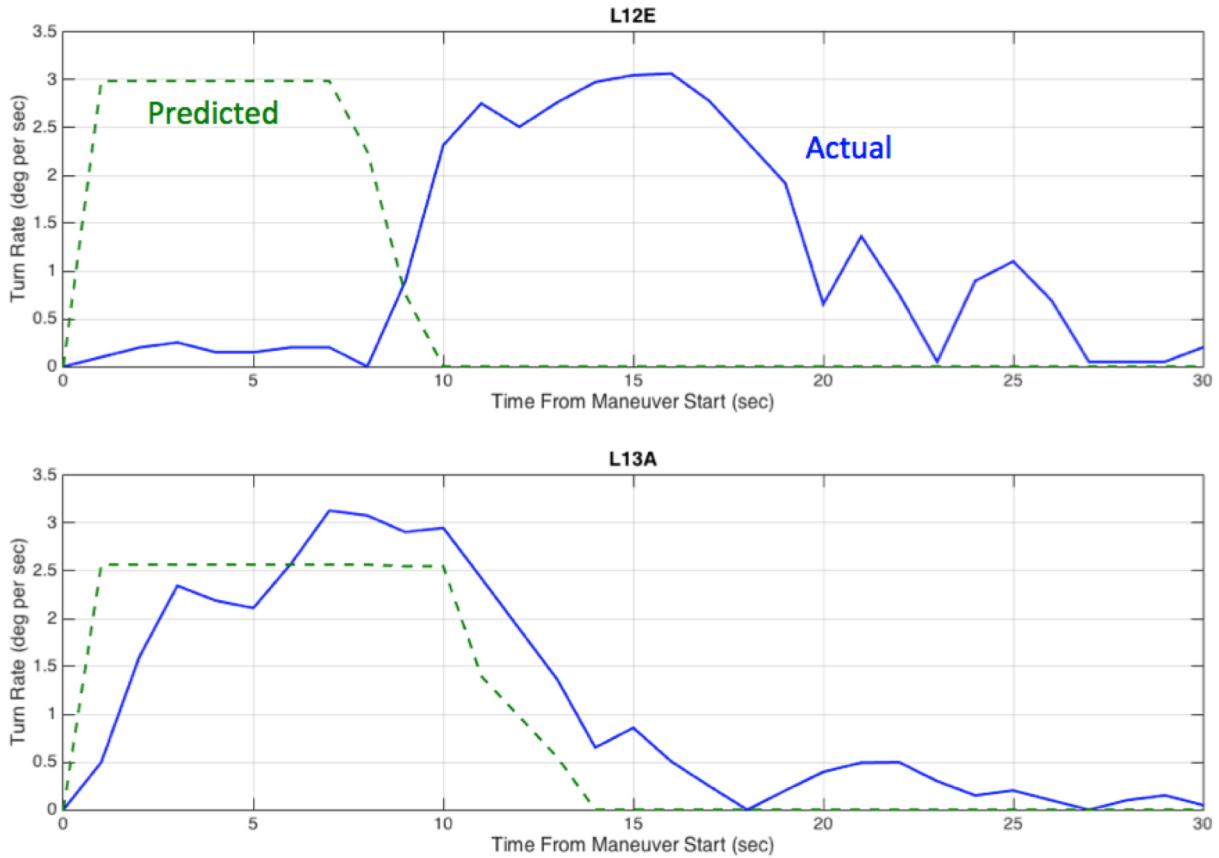


Figure 22. Effect of turn rate on turn prediction accuracy

## Conclusions

The analysis of Flight Test 3 data was primarily focused on assessing the effects of surveillance sensors on trajectory prediction accuracy and DAA alerting performance. An analysis methodology was developed to take full advantage of post-flight surveillance sensor data processing and JADEM/Autoresolver playback capability to maximize the use of data collected during the flights.

Prediction accuracy of trajectories derived from each of the individual surveillance sensors was measured using ADS-B-derived trajectories as a comparative reference. Vertical separation errors for radar-derived trajectories were found to be an order of magnitude greater than ADS-B trajectories at look-ahead times between 110 and 120 seconds. Mean predicted vertical separation error was nearly 3000 ft with the radar compared to less than 500 ft with ADS-B for the same scenarios.

These larger vertical separation errors were attributed to more uncertainty and noise in the altitude and vertical speed measurements from the radar. Additional analysis showed the application of a Kalman filter to the radar altitude and vertical speed measurements could reduce the predicted vertical separation errors to levels comparable to that of ADS-B without an unacceptable amount of lag.

Trajectory predictions derived from TCAS Mode S transponders had greater errors than trajectory predictions derived from ADS-B in terms of horizontal miss distance and modified tau. This flight test result confirms the trajectory accuracy impact of the known bearing inaccuracies associated with TCAS Mode S surveillance data.

A methodology and metrics were developed to quantify DAA system alerting performance. This alerting analysis methodology quantitatively showed the vertical separation alerting criteria to be the most sensitive to trajectory prediction uncertainty and noise, especially with respect to radar surveillance.

## References

- Feitshans, G. L., Rowe, A. J., Davis, J. E., Holland, M., & Berger, L. (2008.). Vigilant spirit control station (VSCS)—‘The face of COUNTER’. In *Proceedings of AIAA Guidance, Navigation and Control Conference Exhibition*. AIAA.
- Erzberger, H., Lauderdale, T. A., & Chu, Y.-C. (2012). Automated conflict resolution, arrival management, and weather avoidance for air traffic control. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 226, pp. 930-949.
- Marsten, M., Sternberg, D., & Valkov, S. (2015). *Flight Test Series 3 Flight Test Report*. NASA Armstrong Flight Research Center, UAS-NAS IT&E Subproject. NASA.
- Soler, G., Jovic, S., & Murphy, J. R. (2015). RUMS-Realtime Visualization and Evaluation of Live, Virtual, Constructive Simulation Data. *AIAA Infotech@Aerospace*. 2015-1648. Moffett Field: AIAA.

## Appendix – Scenario Summary

A summary of the scenarios flown for the JADEM/Autoresolver DAA system is shown in Table A1. Additional scenario details as well as actual flight test cards can be found in (Marsten, Sternberg, & Valkov).

Table A1 – Summary of JADEM/Autoresolver scenarios flown

Scenario Name	Min Vertical Separation (ft)	Angle Into (deg)	Lateral Offset (ft)	Ownship Speed* (kt)	Intruder Speed* (kt)	Ownship Initial Altitude (ft)	Ownship Vertical Velocity (fpm)	Ownship Final Altitude (ft)	Intruder Initial Altitude (ft)	Intruder Vertical Velocity (fpm)	Intruder Final Altitude (ft)
L12A	1000	0	3000	150	180	12000	0	12000	13000	0	13000
L12C	1000	45	3000	150	180	12000	0	12000	13000	0	13000
L12D	1000	90	3000	150	180	12000	0	12000	13000	0	13000
L12E	1000	110	0.0	150	180	12000	0	12000	13000	0	13000
L12M	1000	45	0.0	150	180	12000	0	12000	13000	0	13000
L12N	1000	90	0.0	150	180	12000	0	12000	13000	0	13000
L13A	1000	0	0.0	150	140	16500	0	16500	12500	1000	15500
L13C	1000	45	0.0	150	140	16500	0	16500	12500	1000	15500
L13D	1000	90	0.0	150	140	16500	0	16500	12500	1000	15500
L14A	1000	0	0.0	150	140	12000	0	12000	16000	-1000	13000
L14C	1000	45	0.0	150	140	12000	0	12000	16000	-1000	13000
L14D	1000	90	0.0	150	140	12000	0	12000	16000	-1000	13000
L15A	1000	0	0.0	120	150	12000	1000	15000	16000	0	16000
L15C	1000	45	0.0	120	150	12000	1000	15000	16000	0	16000
L15D	1000	90	0.0	120	150	12000	1000	15000	16000	0	16000
L16A	1000	0	0.0	120	150	16000	-1000	13000	12000	0	12000
L16C	1000	45	0.0	120	150	16000	-1000	13000	12000	0	12000
L16D	1000	90	0.0	120	150	16000	-1000	13000	12000	0	12000
L32A	300	0	3000	150	180	12000	0	12000	12300	0	12300
L32C	300	45	3000	150	180	12000	0	12000	12300	0	12300
L32D	300	90	3000	150	180	12000	0	12000	12300	0	12300
L52A	500	0	3000	150	180	12000	0	12000	12500	0	12500
L52C	500	45	3000	150	180	12000	0	12000	12500	0	12500
L52D	500	90	3000	150	180	12000	0	12000	12500	0	12500

\* - Indicated airspeed denoted by italics, otherwise specified as ground speed