EVALUATING PROTOTYPE SENSE AND AVOID ALTERNATIVES
IN SIMULATION AND FLIGHT

Paul J. Wehner,* Jonathan Schwartz,* Deihim Hashemi,*
Charles T. Howell III,† Harry A. Verstynen,‡ Cathy Buttrill,‡
Mark Askelson§, and William Semke**

NASA Langley Research Center (LaRC) and The MITRE Corporation developed an integrated simulation and flight test capability for testing prototype sense and avoid (SAA) system elements. In fiscal year 2012, the integrated test capability was used by a joint team of investigators from NASA LaRC, the University of North Dakota (UND), and MITRE to evaluate the technical and operational feasibility of using the Federal Aviation Administration’s NextGen Automatic Dependent Surveillance – Broadcast (ADS-B) technology as a cooperative†† surveillance source for on-board, automatic SAA alternatives.

Over a two week period in September 2012, two prototype SAA alternatives were evaluated in flight to help validate the results of extensive modeling and simulation studies conducted by MITRE, UND, and other algorithm development research organizations. During the flight tests, the on-board, automatic SAA alternatives under test were subjected to one-on-one (1v1) flight encounters between NASA LaRC’s surrogate unmanned aircraft system (UAS) and a UND owned and piloted Cessna 172. Using ADS-B messages as the sole-surveillance source for the other aircraft, the SAA alternatives under test identified conflicts, issued maneuver commands, and routinely maintained the desired separation between ownship and the intruder aircraft. Moreover, results to date suggest reasonable congruence between the MITRE simulation studies and the flight encounters examined.

INTRODUCTION

The purpose of this paper is to describe how the integrated test capability developed by NASA Langley Research Center (LaRC) and The MITRE Corporation (MITRE) supports the evaluation of prototype sense and avoid (SAA) alternatives in simulation and in flight. Herein, illustrative

* The MITRE Corporation, McLean, VA. Emails: pwehner, jschwartz, and dhashemi@mitre.org
† NASA Langley Research Center, Hampton, VA. Email: charles.t.howell@nasa.gov
‡ Unisys Corp., Hampton, VA. Emails: h.a.verstynen@gmail.com and catherine.w.buttrill@nasa.gov
§ University of North Dakota Aerospace - Atmospheric Sciences, Grand Forks ND. Emails: askelson@atmos.und.edu and william.semke@engr.und.edu
** University of North Dakota – Mechanical Engineering, Grand Forks ND. Email: william.semke@engr.und.edu
†† Here, 'cooperative' refers to the source of the surveillance information. In cooperative applications (e.g., ADS-B), aircraft must be appropriately equipped to report their location information and/or interpret location reports from other aircraft.
examples are drawn from the test capability’s initial use (i.e., the evaluation of on-board, automatic SAA alternatives that use cooperative-surveillance sources such as the Federal Aviation Administration’s [FAA’s] NextGen Automatic Dependent Surveillance – Broadcast [ADS-B] technology).

Sense and avoid alternatives obtain traffic-situational awareness information using sensors that are either on-board the aircraft (i.e., airborne-based) or ground situated (i.e., ground-based). Using this information, SAA alternatives move to directly (via onboard automation) or indirectly (via remote pilot action) ensure safe separation and collision avoidance from proximate traffic. For times when the command and control (C2) link between the remote pilot and his unmanned aircraft is not available, the sense and avoid mechanism needs to be automatic (i.e., empowered to act without pilot involvement) to ensure needed compliance with existing rules governing both operations near other aircraft and right-of-way (i.e., U.S. Code of Federal Regulations [CFR] Title 14, Part 91, Paragraphs 91.111 and 91.113).

To evaluate prospective SAA alternatives, MITRE and NASA LaRC collaborated to develop an integrated simulation and flight test capability for testing prototype SAA system elements (e.g., algorithms, sensors, architecture, communications, autonomous systems), concepts, and procedures [1]. The capability, which consists of a simulation testbed (Figure 1) and a flight-test platform and testbed (Figure 2), promotes interoperability via the use of well-defined standards and interface requirements [2].

![Figure 1. Simulation Testbed Developed by MITRE.](image)

The simulation testbed, shown in Figure 1, uses a MITRE developed fast-time computer simulation to evaluate the performance of SAA algorithms across a wide array of flight encounters and conditions. With interfaces to existing encounter models (e.g., MIT Lincoln Laboratory’s uncorrelated encounter model of the NAS [3]), recorded flight data, and a custom encounter generator, performance studies can be executed using a wide variety of encounter geometries, previously flown geometries, and/or a specific geometry of interest. The testbed documents algorithm
responses to stimuli and generates a fitness report that describes strengths, weaknesses, and overall performance. The simulation testbed’s initial-operating capability enables researchers to isolate and explore SAA algorithm sensitivities to the following key variables of interest:

• Sources of surveillance information;
• Quality of surveillance information (e.g., availability, integrity, and accuracy);
• Density and composition of proximate traffic;
• Sense and Avoid performance and test envelope selections (e.g., separation-threshold setting); and
• Own-ship flight characteristics.

The simulation testbed’s initial-operating capability enables researchers to isolate and explore SAA algorithm sensitivities to the following key variables of interest:

- Sources of surveillance information;
- Quality of surveillance information (e.g., availability, integrity, and accuracy);
- Density and composition of proximate traffic;
- Sense and Avoid performance and test envelope selections (e.g., separation-threshold setting); and
- Own-ship flight characteristics.

Figure 2. Flight Test Platform and Testbed Developed by NASA LaRC.

The flight-test platform and testbed uses a general aviation aircraft* that has been modified by NASA LaRC to serve as a surrogate unmanned aircraft system (UAS) [4]. The highly instrumented research platform, shown in Figure 2, can be controlled remotely via generic Ground Station uplink or automatically via onboard systems. Furthermore, its operations concept—which re-

* The basis for the NASA LaRC Surrogate UAS is a Cirrus Design SR22 single engine, four-place, composite construction, general-aviation aircraft.
quires a NASA Safety Pilot/Pilot in Command and Systems Operator on-board for all research flights—and its numerous safety features make it possible to “file and fly” almost anywhere in the National Airspace System (NAS). This unique ability affords investigators the opportunity to perform research and test systems in the NAS under real-world conditions; interoperating with other air traffic under existing air traffic control (ATC) rules. Notably, this ability is not afforded to actual UAS which must be operated in a manner that is either segregated from other manned aircraft in restricted airspace, or with other operational restrictions under a Certificate of Waiver or Authorization (COA).

In fiscal year (FY) 2012, a progressive series of experiments and flight tests were conducted by a joint team of investigators from NASA LaRC, the University of North Dakota (UND), and MITRE. The investigation, which served as the initial application for the MITRE-NASA LaRC integrated test capability, had three primary goals:

- Evaluate the technical and operational feasibility of on-board, automatic, ADS-B based SAA alternatives;
- Produce data to inform the UAS community and on-going standards development efforts; and
- Support ongoing validation efforts for analysis tools and infrastructure.

The remainder of the paper describes the FY2012 case study in detail. It concludes with a description of future studies and a summary.

**CASE STUDY: ON-BOARD, AUTOMATIC, ADS-B BASED SAA ALTERNATIVES**

The following sections describe the FY2012 case study’s background, objective, scope, execution, and initial findings.

**Background**

In 2010, the FAA mandated the use of ADS-B transmitters (i.e., ADS-B Out) by 2020 on all aircraft operating in airspace that today requires Mode C or Mode S transponders [5]. In effect, less than a decade from now most of the aircraft operating in the NAS will be subject to ADS-B Out equipage requirements.

ADS-B is a well-defined, accessible surveillance technology that is ‘cooperative’ in design (i.e., aircraft must be appropriately equipped to report their location information and/or interpret location reports from other aircraft). ADS-B equipped aircraft use global positioning system (GPS) signals and aircraft avionics to transmit own-ship information (e.g., position, altitude, velocity, aircraft ID) to other aircraft and ground receivers using one of two frequencies (i.e., 1090 MHz extended squitter [1090 ES] or 978 MHz Universal Access Transceiver [UAT]). In-range ADS-B In equipped aircraft operating on the same frequency receive the ADS-B transmission directly via air-to-air messages. In-range ADS-B In equipped aircraft operating on dissimilar frequencies receive the ADS-B transmission indirectly via a ground-to-air ADS-B uplink service called ADS-R (re-broadcast). To complete the situational awareness picture in the cockpit*, radar derived information on the location of non-ADS-B Out, transponder-equipped aircraft (e.g., Mode C or S equipped aircraft) is provided via another uplink service, Traffic Information Ser-

---

* UAT equipped ADS-B In aircraft also receive Flight Information Services-Broadcast (FIS-B) messages. This uplink service provides meteorological and aeronautical information in the cockpit.
vice-Broadcast (TIS-B), when these aircraft are within the service volume of secondary surveillance radar and ADS-B ground stations.

ADS-B Out equipped aircraft transmit own-ship updates at least once per second. The time interval between successive ADS-R and TIS-B transmissions is longer. While ADS-R transmissions vary as a function of ground report update internals, 95% of the time their time interval is 5 seconds in the terminal domain and 10 seconds in the en route domain [6]. Whereas TIS-B transmissions, which vary as a function of radar source, provide position updates (within radar coverage areas) approximately once every 3-13 seconds [6,7].

As ADS-B equipage levels rise, an increasingly complete and accurate situational awareness picture forms for appropriately equipped users. While leveraging this improved situational awareness picture is quite appealing*, doing so within the context of an on-board, automatic sense-and-avoid application is not a foregone conclusion. As presently defined, ADS-B is restricted to specific applications which do not include self-separation or collision avoidance, while TIS-B and ADS-R (by proxy) are intended to assist solely in the visual acquisition of other aircraft [7].

**Case Study Objective**

A multi-layered solution comprised of one or more on-board, automatic components is likely needed for UAS SAA applications to achieve an acceptable level of risk given the lack of an on-board pilot and vulnerabilities in the C2 link. However, significant research is needed to determine if cooperative signals (e.g., ADS-B, TIS-B, ADS-R) may† serve a role in automatic SAA applications.

In a collaborative effort amongst several entities including MITRE, NASA LaRC, UND, Draper Laboratories, North Dakota State University, and the North Dakota Army National Guard, a multi-year initiative‡ was formed in FY2011 to study cooperative concepts for integrating UAS in the NAS. Under the initiative, participating research organizations§ collaboratively test UAS sense-and-avoid technology, algorithms, and methods of joint interest. In this manner, investigators from NASA LaRC, UND, and MITRE initiated a study in FY2012 to explore the following key research questions:

- If cooperative-surveillance information is available for proximate traffic, can a cooperative automatic sense-and-avoid algorithm perform self-separation?
- What quality of surveillance information (e.g., availability, integrity, and accuracy) does a cooperative automatic sense-and-avoid algorithm require to perform self-separation?
- Which encounter geometries prove challenging to resolve using cooperative automatic sense-and-avoid algorithms?

---

* A complete and accurate understanding of proximate traffic leads to simpler SAA decision logic, which in turn reduces the complexity and cost for development and certification.
† Both utility and permissibility aspects need to be addressed.
‡ The multi-year initiative is often referred to as the Limited Deployment – Cooperative Airspace Project (LD-CAP).
§ NASA LaRC, MITRE, Draper Laboratory, and North Dakota based research entities collaborate through in-kind support. North Dakota National Guard participates under State direction and funding.
Case Study Scope

In FY2012, the MITRE-NASA LaRC integrated test capability was used to evaluate the performance of prototype cooperative automatic SAA algorithms in 1v1 encounters between general aviation aircraft. As illustrated in Figure 3, the study utilized an iterative approach rooted in simulation studies and augmented by various flight events. Each flight event offered an opportunity to not only collect data to inform ongoing research and development efforts, but also demonstrate concept/technology readiness to interested stakeholders. Key constraints and limitations of the FY2012 study are discussed below.

Data Sources. As illustrated in Figure 3, multiple sources of data were collected throughout each phase of the investigation. Most of the data supported FY2012 post-flight analyses and/or simulation studies; however, some was collected in support of future studies. The SAA algorithms under test only had access to ADS-B air-to-air messages and, at times, own-ship air state data.

ADS-B served as the sole source of cooperative-surveillance information for proximate traffic. On-board own-ship, configuration settings within the algorithm-hosting infrastructure governed whether the ADS-B feed was ‘filtered’ or ‘unfiltered.’ If the ‘filtered’ setting was active, the SAA algorithm under test received only the ADS-B transmissions that matched the participating aircraft’s International Civil Aviation Organization (ICAO) address. If the ‘unfiltered’ setting was active, the SAA algorithm under test received the decoded UAT ADS-B In feed for all proximate traffic. With the exception of one proof-of-capability flight at NASA LaRC, the ‘filtered’ setting was used for all FY2012 flight events.

† See the “Encounter Scenarios” section for details on the altitude offset schema implemented during the FY2012 flight events.

* Data collected from three tactical MPQ-64 Sentinel radars operated by the North Dakota Army National Guard is being used as part of a separate LD-CAP study investigating the concept of converting primary radar returns into TIS-B like messages.
Differential GPS and in-cockpit audio and video collection was limited by flight event; however, air state data were recorded for all flights. The algorithm-hosting infrastructure governed whether the SAA algorithm under test had access to own-ship’s air state data; for the majority of the FY2012 flight evaluation (i.e., approximately 35 of the 39 hours flown), the air state data feed was disabled.

Flight events that included the NASA LaRC Surrogate UAS benefited from the following on-board data collection systems:

- Aircraft navigation data from the dual Garmin GNS-430 GPS navigation receivers;
- Aircraft altitude, attitude, and state vector information from the Athena Air Data and Attitude Heading Reference System (ADAHRS);
- Automated Dependent Surveillance-Broadcast data, including own aircraft Wide Area Augmentation System (WAAS)/GPS derived position, state vector, and status information, track data on other proximate aircraft received from other aircraft or from ground station data, and broadcast weather and system status information;
- Dual digital video tape recorders; one recording NTSC 480P, 24 frame per second video and digitalized audio of the cockpit environment and the other recording NTSC 480P video of the cockpit large format Avidyne Multi Function Display; and
- An independent, high accuracy differential GPS (DGPS) positioning system used to provide baseline aircraft position data.

In addition to the systems described above, the NASA LaRC Surrogate UAS is equipped with three general-purpose computers, GPC-1, GPC-2 (not used for this project), and GPC-3. GPC-3 is used to integrate a broad range of aircraft and systems data with data logging for the ADAHRS, autopilot, the ADS-B system, and to provide data formatting and management for research applications running on GPC-1. GPC-1 is used to host the research algorithms and provide both internal application data logging and input/output data logging of all data used by the research algorithm(s). Additionally, steering commands generated by the research algorithms running in GPC-1 are logged, formatted, and passed back through GPC-3, which then transmits the commands to the S-TEC 55X Autopilot and the throttle servo.

Data logs from all operational data collection systems and the digital video and audio were transferred after each flight day to the project data repository or transcribed to digital media for storage.

Notably, both aircraft in the In-Flight Data Collection Flights and the intruder aircraft in the North Dakota 2012 Flight Evaluations were each equipped with Garmin G1000 Integrated Flight Management System (IFMS) with dual Wide Area Augmentation System (WAAS) GPS navigation receivers, digital Air Data Computer and Aircraft Heading Reference systems, and integrated data logging system. The in-flight data log for each flight included time-stamped aircraft position, geometric and barometric altitude, other state vector data, such as rate of climb/descent, and navigation receiver and communications system status at a 1 second update rate. After each flight, the G1000 IFMS data log was downloaded and added to the project data repository for use in flight test data analysis.

An additional, aircraft independent system developed by MITRE was used in all In-Flight Data Collection Flights and during select flights on-board the intruder aircraft in the North Dakota 2012 Flight Evaluations. The Portable Airborne Data Collection System (PADCS) consists of a MITRE-developed portable UAT (with transmitter function disabled) with internal altitude and
WAAS GPS sensor and a laptop computer running MITRE-developed software. While the PADCS’s primary purpose is to provide a portable algorithm server to allow execution of UAS sense and avoid algorithms aboard any aircraft used for research data collection, the system also logs all ADS-B messages received from proximate aircraft and ground broadcast stations, as well as records the location of the intruder aircraft at a 1 Hz data rate.

**Research Algorithms.** To accomplish the study, the research team solicited candidate SAA algorithms for test. A handful of candidates were identified; however, only two compliant algorithms were made available in time to support FY2012 test activities. Both of the compliant algorithms were developed by independent teams and subjected to simulation and flight testing within the MITRE-NASA LaRC integrated test capability. UND’s Unmanned Aircraft Systems Engineering (UASE) team developed one of the candidate SAA algorithms and MITRE technical staff developed the other.

The UND UASE research algorithm was initially developed in 2008 and incrementally enhanced thereafter in support of multiple UND research projects [8,9,10]. The research algorithm uses interval programming (IvP) techniques to calculate the instantaneous ‘risk’ around own-ship and propose mitigations that seek to balance avoidance needs against other desires (e.g., comply with FAA manned flight right of way standards and execute mission objectives). In support of the FY2012 study, the UND UASE team modified the SAA algorithm to comply with the defined standards and interface requirements of the MITRE-NASA LaRC integrated test platform.

The MITRE research algorithm leverages prior research performed in support of ATC maneuver planning [11,12,13], but it was developed specifically to support the FY2012 study. It detects potential conflicts and uses the reported navigational accuracy of available ADS-B messages to calculate the least costly and disruptive maneuver capable of achieving the desired separation. By design, the research algorithm does not introduce additional buffers and/or safety regions to the proposed resolution to enable the direct evaluation of achievable performance given available signals and their self-reported accuracy.

**Separation Criteria.** The FY2012 study used separation thresholds that were user defined. While time and/or distance values were acceptable, all FY2012 flight evaluations used distance-based criteria. Specifically, a horizontal threshold of 2 nautical miles (nmi) and a vertical threshold of 500 feet.

Under basic VFR weather minimums, the Aeronautical Information Manual notes a regulatory basis for flight visibility of 3 statute miles (approximately 2.6 nmi) for Class E airspace operations below 10,000 feet MSL [6]. As such, notwithstanding the known limitations of visual acquisition [15], the selected thresholds enabled the participating pilots to visibly acquire one another in many of the encounter geometries flown (overtake geometries did not generally provide the opportunity for intervisibility).

**Flight Test Operations and Approvals.** Software did not serve as a mitigation for any hazard; all hazards were addressed via design features or operational procedures.

The overarching safety documentation and hazards analysis for the Langley Flight Tests and North Dakota 2012 Flight Evaluations were agreed to by all partner organizations participating in the operational execution, and approved by the NASA Langley Airworthiness and Safety Review.

---

1 Disruption/cost is measured in terms of consistency with established “rules of the road” and maneuver severity. That said, biases among speed, altitude, lateral, and combination maneuvers were managed via input parameters to afford analysts greater flexibility in testing.
All operations were conducted in accordance with applicable NASA LaRC operational guidelines, UND operational guidelines, and FAA/local regulations and procedures for each operational area, as applicable. In addition, safety procedures included preflight briefings for all participants, checklists, and prepared flight test cards. Furthermore, flight operations were limited to day visual meteorological conditions (VMC) and the modified autopilot was only engaged at altitudes of 500 feet or higher above ground. Flight test procedures ensured a minimum of 500 feet positive vertical separation between participating aircraft at all times through use of assigned minimum and maximum altitudes within the flight test area.

During In-Flight Data Collection flights, MITRE pilots flew general-aviation aircraft in scripted passes to obtain an early look at SAA algorithm function under actual flight conditions. Using the PADCS, all triggers, log files, and commands created/issued by the SAA algorithm under test were recorded but not executed.†

Similar to the Langley and North Dakota flights referenced above, all In-Flight Data Collection operations were conducted in accordance with applicable FAA/local regulations and procedures. In addition, safety procedures included preflight briefings for all participants, an individual and all-up operational risk assessment conducted using formal risk assessment procedures and checklist, prepared flight test cards, hard ceilings/floors, and flight operations limited to day VMC. Furthermore, flight test procedures ensured a minimum of 500 feet positive vertical separation between participating aircraft at all times.‡ As such, the collection activity presented no greater hazard than one would find in normal IFR and VFR aircraft separations in today’s ATC system.

**Encounter Scenarios.** Using the simulation testbed, sensitivity analyses were performed to understand the strengths and weaknesses of the research algorithms in co-altitude, zero-miss-distance encounters. The results suggested that the research algorithms under test were sensitive to the following encounter parameters: time to predicted loss of separation, initial speed settings, initial headings, and the accuracy of surveillance information. To the degree possible, effort was taken to explore these parameters in flight.

In the FY2012 flight events, encounter parameters were varied to create trajectories that—if unmitigated—resulted in “virtual” collisions. In part, this was accomplished via the use of the altitude offset schema illustrated in Figure 4.

---

* Vertical separation between participating aircraft is based on pressure altitude from IFR-certified pitot-static systems in each aircraft, not from ADS-B altitude.
† Unlike the system on-board the NASA LaRC Surrogate UAS, the algorithm-hosting infrastructure used during In-Flight Data Collection flights was not integrated with flight systems. Consequently, no commands could be passed or executed in an automatic fashion.
‡ Vertical separation between participating aircraft is based on pressure altitude from IFR-certified pitot-static systems in each aircraft, not from ADS-B altitude.
The Pilot in Command of the intruder aircraft flew at fixed altitude and fixed speed between a given encounter’s first and second waypoint. While a positive vertical separation of at least 500 feet was always maintained between the research aircraft and the actual intruder aircraft, the ADS-B altitude information presented to the SAA algorithm (via the algorithm-hosting infrastructure) was offset such that the resulting “virtual intruder aircraft” appeared co-altitude with the research aircraft. Actual intruder aircraft altitude information remained available to the surrogate UAS flight crew throughout the encounter to enhance situational awareness and aid in encounter geometry setup and execution.

To explore the performance of the research algorithms with respect to initial speeds and headings, scripted encounters were created. While achievable speeds varied by aircraft pairs and flight conditions, attempts were made to orchestrate generic-speed profiles (i.e., own-ship faster than the intruder, own-ship slower than the intruder, and own-ship at equal speed with the intruder).

Callout “A” in Figure 5 notes the aircraft types and pairs used in support of the FY2012 flight events. Notably, all flight-event aircraft were equipped with UAT ADS-B In and Out technology. Callout “B” in Figure 5 depicts the initial headings investigated during the FY2012 flight events.

![Figure 5. Summary of Aircraft and Encounter Geometries for the FY2012 Flight Events.](image)

Case Study Execution

The MITRE-NASA LaRC integrated test capability was used throughout the FY2012 study. Illustrative examples by phase (i.e., pre-flight, flight, and post flight) are provided in the following sections.

**Pre-Flight Phase.** The simulation testbed supported algorithm-development efforts and flight planning throughout the Pre-Flight phase. Iterative studies were executed to expose vulnerabili-
ties and explore potential mitigations (e.g., decision-logic modification, updated buffer settings). Figure 6 provides an illustrative example of two iterative test cycles for one of the research algorithms under test.

![Figure 6. Iterative Development Example (Evaluating the Impact of SAA Algorithm Changes via Simulation Testbed Studies).](image)

In each of the test cycles shown above, SAA algorithm performance as a function of initial headings (0° to 360° in 5° increments), initial speeds (e.g., own-ship faster than intruder, own-ship slower than intruder, own-ship and intruder at like speeds), and ADS-B message quality (e.g., accuracy and timeliness) was explored. The “A” and “B” graphs in Figure 6 depict own-ship’s position relative to the intruder aircraft. The red marks denote areas along the sampled trajectories where the user specified separation threshold was not maintained. The yellow and black marks visible in the Figure 6 “A” graphs denote areas where the user specified separation threshold was maintained (i.e., yellow denotes co-altitude but separated; black denotes vertical separation). The “C” graphs provide a histogram of the minimum lateral miss distance for each of the sampled trajectories.

In the Figure 6 example, decision-logic modifications were introduced from one test cycle (1A, 1B, 1C) to the next (2A, 2B, 2C). Via the simulation testbed, developers evaluate the impact of the resultant change and determine next steps. In the Figure 6 example, the changes resulted in both a net reduction in violations from 717 to 414, and a shift toward less severe breaches of the user specified separation distance. To investigate trajectories of interest further, different analytic views are leveraged. Figure 7 illustrates five such views (i.e., ground, vertical, plan, aircraft state, and encounter metrics).

Each analytic view in Figure 7 is time synched to enable analysts to step through the encounter (forward or backward) as needed. The unique perspectives enable one to clearly observe the progression of SAA algorithm, aircraft, and metrics throughout the encounter. For example, between callout 1 and 2 in Figure 7, the algorithm under test increases own-ship’s speed in an attempt to pass in front of the intruder. As own-ship responds, the predicted miss distance gradually improves from approximately 0 nmi to -1.5 nmi before leveling off between callout 2 and 3. At callout 3, the SAA algorithm commands heading and altitude adjustments in an attempt to achieve the desired separation. At callout 4 in Figure 7, a 0.1 nmi breach of the protection volume occurs (shown in red on the Ground, Vertical, and Plan views). From callout 4 to 5, the aircraft continues to increase separation.
Figure 7. Analytic Views from the Simulation Testbed used for Algorithm Development, Test, and Evaluation.

The ability to evaluate SAA algorithm responses in this manner enables one to identify cause and effect relationships, isolate stimuli of interest, and frame future studies (simulation and/or flight) as warranted. For example, Figure 8 provides an illustrative example of tuning buffer settings via a simulation study. Specifically, graphs “A” to “E” in Figure 8 show the effect of decreasing the horizontal buffer size of one of the research algorithms in increments of 0.01 nmi (approximately 60 feet). As the buffer size decreases, the lateral miss distance distribution gradually shifts toward the right while vertical resolutions become more pronounced. Ultimately, the smallest buffer setting (E = 0.01 nmi) yields 11 shallow violations. Notably, buffer setting E was used during the FY2012 flight events.

Figure 8. An Example of Tuning Algorithm Settings to Achieve Desired Behaviors.
**Flight Time Phase.** When research algorithms are transferred from the MITRE simulation testbed to the NASA-LaRC flight test platform and testbed, hardware-in-the-loop check cases are performed. If the SAA algorithms pass the check cases, flight evaluations proceed in accordance with applicable operational guidelines, regulations, procedures, and releases.

The prioritization schema for which algorithms to fly in which encounters is influenced by many factors (e.g., flight conditions, aircraft capabilities, pre-flight studies, research goals, resources). Since it is neither feasible nor practical to exhaustively test SAA algorithms in flight, the prioritization schema is often revisited during the flight event (i.e., while the aircraft are airborne) based on real-time pilot and command-center observations. For example, if analysts in the command center observe flight responses that appear to match pre-flight simulation studies well (such as in the flight and simulation encounter shown in Figure 9), the desire to execute multiple replications may be deprioritized to allow additional flight time for encounters with poor or no agreement.

![Flight Time Prioritization of Encounters to Optimize Valuable Flight Resources](image)

**Figure 9. Flight Time Prioritization of Encounters to Optimize Valuable Flight Resources.**

**Post-Flight Phase.** After each flight event, flight data is reviewed in detail to compare planned/anticipated versus observed (e.g., encounter setups, flight conditions, SAA algorithm responses). As discoveries are made, additional simulation studies and SAA algorithm modifications often follow.

Figure 10 provides an illustrative example of a 45° crossing encounter in simulation and in flight. Here, the flight trajectory and the simulated trajectories are similar but not identical. The initiation distances and resolutions (i.e., increase speed, turn, and descend) are alike, but the observed flight trajectory differs in its net separation (i.e., 3.2 nmi versus 3.7 nmi) and profile. On review, however, the differences can be attributed in part to differences in achieved air speed and headings.

---

*The flexibility to reprioritize FY2012 flight encounters based on real-time observables was supported by ground-to-air communications and live situational awareness feeds in the command center.*
Fundamentally, SAA algorithms attempt to predict a miss distance between own-ship and proximate traffic. If that distance is so small as to be considered a loss of separation, a maneuver to avoid is determined. As such, the magnitude and consistency of the predicted miss distance in the lead-in time (i.e., the time immediately prior to when avoidance decisions are made) strongly influences the behavior and performance of a given SAA algorithm.

In the planned flight trajectories, the incident angle between aircraft varied at known intervals (e.g., 15°, 30°, 45°). Prior to beginning a given test encounter, it was assumed that both aircraft would obtain their assigned heading while maintaining a fixed speed and altitude. As such, both aircraft were initialized to trajectories that would cause them—if unmitigated by algorithm commands—to arrive at a predetermined point at the same time (i.e., create a (virtual) zero-miss distance encounter).

In the North Dakota FY2012 flight evaluations, however, the observed minimum miss distances for flown encounters not only exceeded zero consistently but there was also a high degree of variability in the predicted-miss distance immediately prior to the first avoidance action. Figure 11 illustrates this variability by plotting predicted miss distance over time for 10 randomly selected encounters along with the 45° encounter geometry explored in Figure 10.
Figure 11. Predicted Miss Distance Variability Observed in Flight.

To the extent that this real-world flight data can be viewed as representative of pilotage effects, it presents a non-trivial challenge for SAA algorithm developers and a sampling consideration for simulation studies. For example, in the simulation studies performed in the Pre-Flight phase for the North Dakota FY2012 flight evaluations, the predicted miss distances modeled rarely fluctuated more than ± 0.4 nmi during the lead-in time since analysts assumed that the predicted miss distances were relatively well behaved. Such sampling assumptions need to be revisited in light of the observed North Dakota flights.

Two influential factors in the realized miss distance are heading and speed. As these factors change (or are perceived to change), so too does the set of viable solutions for a given encounter. Furthermore, if adherence to a particular rule set (e.g., rules of the road) or resolution hierarchy is strongly favored by the SAA algorithm under test, maneuver severity is also directly affected. Figure 12 provides an illustrative example of this phenomenon using simulation testbed outputs.

Figure 12. Sensitivity to Angle and Speed Variations during Flight.

In the Figure 12 case, the SAA algorithm under test strongly favors the execution of a “pass behind” maneuver. As a result, as heading (“A” graphs) and speed (“B” graphs) settings fluctuate, the maneuver angle required to resolve the conflict via a “pass behind” action varies notably. In the “A” graphs, both own-ship and intruder speeds are fixed at 96 kts but the encounter angle between them is varied from 8° to 16° in 2° increments. In the “B” graphs, the encounter angle be-
between own-ship and the intruder is fixed at 12°, but the speed of the intruder is varied from 101 kts to 91 kts in 2.5kts increment. In both the “A” and “B” graphs, the maneuver angle required to resolve the conflict via the SAA algorithm’s preferred method (i.e., a “pass behind” action) varies notably (i.e., from left (negative) 69° to 26° in“A” and from left (negative) 59° to 18° in “B”). If both settings vary simultaneously, the effect is even more pronounced.

Initial Findings. Both in simulation and in flight, when cooperative-surveillance information was made available for the intruder aircraft, the research algorithms under test were able to identify the conflict, issue maneuver commands, and routinely maintain the desired separation between own-ship and the intruder aircraft.

133 of the 147 encounters flown during the North Dakota FY2012 flight evaluations were ‘graded.’ The 14 flight encounters that were not ‘graded’ consisted of 13 incomplete data runs and 1 out-of-scope run that was flown in preparation for FY2013 flights (e.g., intruder flying a circular pattern). In the encounter scenarios explored and graded, the SAA algorithms under test maintained the desired separation distance 67% of the time in flight with the majority of registered violations (84%) occurring in the region between 1 and 2 nmi. Figure 13 provides summary results for the flight data.

![Figure 13. FY2012 North Dakota Flight Evaluation Results (Flight Data Shown).](image)

As illustrated in callout “D” in Figure 13, the Overtaken encounters (i.e., when ownship is overtaken from behind by a faster moving intruder) and Orthogonal encounters yielded the highest concentration of violations per flight trial (83% and 42% respectively). Callout “E” provides the breakdown for Acute, Orthogonal, and Obtuse encounters based on the intruder aircraft’s orientation to own-ship during the encounter (i.e., left denotes that the intruder was approaching from own-ship’s left side).
In simulation studies, the SAA algorithms under test were also able to routinely maintain the desired separation distance from the intruder aircraft. Figure 14 illustrates the results of a sensitivity analysis that explored the impact of speed and angle fluctuations. Similar to the live flight results shown above (i.e., callout "C" of Figure 13), callout “A” in Figure 14 illustrates a strong tendency for resolutions to distribute around the designated separation threshold (i.e., 2 nmi horizontal). Given the general absence of buffers (by design) in the research algorithms, this is an expected phenomenon.

In the Figure 14 example, the SAA algorithms under (simulation) test maintained the desired separation distance 97% of the time with the majority of registered violations (99%) occurring in the region between 1 and 2 nmi. Callout "B" illustrates the effect of incident angle on the frequency of the recorded violations. Notably, in this ~750K simulation sample, the encounter geometry and orientation biases observed in the limited run, North Dakota FY2012 flight evaluations do not appear to hold on first glance (i.e., Acute and Overtaken geometries emerge as the geometries with the highest concentration of violations, and orientations which feature an ‘intruder approaching from the right’ exhibit elevated violation rates). However, as previously discussed, the realized angles commonly fluctuated due to flight conditions. As such, it was not uncommon for a flown encounter to present with a resultant incident angle 5-15° off planned (e.g., planned 90° presents as a realized 75° encounter). This reality of flight further emphasizes the importance of evaluating algorithm performance over a rich set of encounter/challenge scenarios.

In related simulation studies, as the self-reported quality of the surveillance information degraded, the SAA algorithms under test typically issued more severe maneuvers earlier in the encounter timeline. In all flight trails, however, the quality of the surveillance information was consistently high (i.e., aircraft were operating in precision navigation mode at all times) so a comparative pattern could not be established.

FUTURE STUDIES

Flight evaluations of three candidate SAA algorithms (i.e., updated versions of the FY2012 UND and MITRE research algorithms along with a third research algorithm developed by Draper Laboratories with support from NASA LaRC) are planned for July 2013 and August 2013 using...
the NASA LaRC Surrogate UAS testbed. The July 2013 flight evaluations will occur in eastern North Dakota and the August 2013 flight evaluations will occur in southern Virginia.

FY2013 flight evaluations will investigate the performance of the cooperative automatic SAA algorithms in more complex encounters (e.g., descent/ascend/turn into conflict). In addition, the FY2013 flight evaluations will explore the effect of alternative cooperative information sources (e.g., ADS-R and TIS-B) on SAA algorithm performance. Together with the FY2012 flights, these flights will help to further validate the results of extensive modeling and simulation studies conducted by MITRE and other algorithm development research organizations.

SUMMARY

The integrated MITRE-NASA LaRC test capability was used by a joint team of investigators from UND, NASA LaRC, and MITRE to evaluate the technical and operational feasibility of using the FAA’s NextGen ADS-B technology as a cooperative surveillance source for on-board, automatic SAA alternatives. Over a two week period in September 2012, two research SAA algorithms were evaluated in flight to help validate the results of extensive modeling and simulation studies conducted by MITRE, UND, and other algorithm development research organizations. During the 1v1 flight encounters tested, the on-board, automatic SAA alternatives identified conflicts, issued maneuver commands that were executed by the NASA LaRC Surrogate UAS, and routinely maintained the desired separation between the NASA LaRC Surrogate UAS (i.e., own-ship) and the UND owned and piloted Cessna 172 (i.e., the intruder aircraft). Moreover, results to date suggest:

- Reasonable congruence between the MITRE simulation studies and the flight encounters examined;
- ADS-B (air-to-air) provides sufficient information for the automatic SAA algorithms tested to identify conflicts and suggest self-separation maneuvers;
- Small buffers around protection zones may notably improve overall SAA algorithmic performance;
- Encounter geometries where the UAS is being overtaken by a faster moving intruder are a challenge if we assume that the UAS is the burdened aircraft;
- Headings & speed perturbations due to factors such as wind and pilotage create algorithmic challenges; and
- Algorithms need to be robust to missing inputs to ensure adequate degraded operation

A multi-layered solution comprised of one or more on-board, automatic components is likely needed for UAS SAA applications to achieve an acceptable level of risk given the lack of an on-board pilot and vulnerabilities in the C2 link. However, realizing the final solution’s composition will likely require a non-trivial level of effort. An integrated test capability, such as that developed by MITRE and NASA LaRC, offers investigators the ability to perform the necessary research efficiently. Furthermore, the iterative build-test-build approach demonstrated throughout the FY2012 Case Study can serve as a model for the development and evaluation of systems and algorithms needed to enable the safe integration of UAS in the NAS.

ACKNOWLEDGEMENTS

The authors wish to thank Frank Jones, Glenn Roberts, Andy Lacher, and Al Palmer for their vision, enthusiastic support, and valuable feedback. The authors also wish to acknowledge the invaluable contributions of Todd Stock, Chris Theisen, Gregory Orrell, Trevor Woods, Ganghuai Wang, Rick Yasky, Wayne Schindler, Joshua Carbonneau, John Nordlie, Steve Cook, Michael
REFERENCES


5. 14 CFR §92.225 and §91.227


DISCLAIMER

The views, opinions, and/or findings contained in this presentation are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.