Safety Analysis of the Advanced Airspace Concept using Monte Carlo Simulation

David M. Blum [1]
Stanford University, Stanford, CA, 94305

David Thipphavong [2], Tamika L. Rentas [3]
NASA Ames Research Center, Moffett Field, CA 94035

and

Ye He [4], Xi Wang [5], and M. Elisabeth Pate-Cornell [6]
Stanford University, Stanford, CA 94305

Safe separation of aircraft is a primary objective of all air traffic control systems. A probabilistic risk analysis of a next-generation air traffic control operational concept was conducted using Monte Carlo simulation. Results indicate that this safety-critical system would have a mean time between mid-air collisions of about 281 years in enroute airspace, which is around the desired target for an automated separation assurance function. Among the simulation trials that ended with mid-air collision, most involved failure to locate and track aircraft through radar or satellite positioning systems with the remainder due to conflict detection failure.

Nomenclature

λ_{NMAC} =   Rate of near mid-air collisions (NMAC)

Acronyms

AAC   =   Advanced Airspace Concept
AC   =   Aircraft
ACES   =   Airspace Concept Evaluation System
ADS-B   =   Automatic Dependent Surveillance-Broadcast
AR   =   Autoresolver
CTAS   =   Center TRACON Automation System
ERAM   =   En Route Automation Modernization
FAA   =   Federal Aviation Administration
FMS   =   Flight Management System
ft   =   Feet
GNSS   =   Global Navigation Satellite Systems
GPS   =   (United States NAVSTAR) Global Positioning System
LoS   =   Loss of Separation
Mode S   =   Data link mode using aircraft transponder

---

[1] Graduate Student, Management Science & Engineering, dmblum@stanford.edu, AIAA Member.
[2] Aerospace Engineer, Flight Trajectory Dynamics and Controls Branch, david.p.thipphavong@nasa.gov, AIAA Member.
[3] Aerospace Engineer, Flight Trajectory Dynamics and Controls Branch, tamika.l.rentas@nasa.gov, AIAA Member.
[4] Graduate Student, Applied Physics, yehe@stanford.edu.
[5] Graduate Student, Management Science & Engineering, jxiwang@stanford.edu.
[6] Professor, Management Science & Engineering, mep@stanford.edu.

NAS	=	National Airspace System
NASA	=	National Aeronautics and Space Administration
NMAC	=	Near Mid-Air Collision
nmi	=	Nautical Miles
SA	=	Separation Assurance
SSR	=	Secondary Surveillance Radar
TSAFE	=	Tactical Separation-Assured Flight Environment
TCAS	=	Traffic (alert) Collision Avoidance System
VDL2	=	Very high frequency Data Link Mode 2

I. Introduction

AIR traffic demand is expected to increase substantially over the next 20 years, but controller workload limits airspace capacity [1]. It is expected that higher levels of automation for separation assurance will be required to accommodate future demand growth. NASA is currently investigating both ground-based and airborne automation concepts for separation assurance [2-6]. One of the most crucial steps on the road toward implementation is to determine the level of safety provided by candidate separation assurance systems. The subject of this analysis is the Advanced Airspace Concept (AAC), a proposed multi-level ground-based separation assurance system [4-6]. As a point of reference, currently about nine critical near mid-air collisions (NMAC) can be attributed to air traffic control operational errors occur each year based on pilot reports [7-8].

In a recent study of separation assurance in today's National Airspace System (NAS), researchers Hemm and Busick decomposed the separation assurance function performed by today's air traffic controllers into its many hardware, software, and operational subcomponents [9]. However, the focus of their study was on today's operations, and their methodology cannot be readily extended to analyze the risk posed by a highly automated air traffic control system such as AAC that will need to be able to handle future traffic up to three times current demand.

Blom, Obbink, and Bakker also investigated the collision risk of an airborne self-separation concept using rare event Monte Carlo simulation [10]. Their results show that coordination in conflict resolution is necessary to prevent conflict clusters from growing faster than the resolution algorithm can handle.

Andrews, Welch and Erzberger performed a safety analysis of AAC using a fault tree methodology to study four fault types: (1) nominal conditions, (2) information fault non-conformance, (3) control fault non-conformance, and (4) service interruption [11]. Their results suggest that the ground-based AAC could achieve the airspace capacity, efficiency, and safety levels that are expected in 20-30 years given the deployment of appropriately designed safety features. The authors computed a mean time between collisions under AAC directly through application of the Law of Total Probability.

The current study differs from previous research because it decomposes each of AAC's safety layers into their respective sub-components. This allows for identification of dependencies among the safety layers by determining common failure modes at the sub-component level. Furthermore, instead of conducting fault tree analysis in isolation, this study employs Monte Carlo simulation with fault tree-style logic built into it to capture the time-dependent uncertainty surrounding detection and resolution together with the probabilistic relationships between events. Simulation makes for a natural means of tracking a conflict through an event tree and is easily scalable to analyze forecasted levels of air traffic. The mean time between collisions under AAC is computed using the sample statistics of the simulation output

Section II provides additional detail about AAC and the components that comprise each of its subsystems. Sections III and IV describes the failure analysis and Monte Carlo simulation methodologies that were employed to determine the rate of mid-air collisions in enroute airspace in the NAS given implementation of AAC. Section V presents the results of the Monte Carlo simulation. Section VI analyzes the simulation results, presents a discussion of external factors that could affect the safety performance of AAC, and offers suggestions for reducing the overall risk of mid-air collision. Section VII concludes the paper.

II. Background

The Advanced Airspace Concept (AAC) is a proposed ground-based separation assurance system that monitors and maintains safe separation between aircraft automatically. Trajectory changes initiated by pilots and/or the ground system are communicated via air-ground data link. AAC includes two safety layers, Autoresolver (AR) and Tactical Separation-Assured Flight Environment (TSAFE), in addition to the two that are already present in today's system, Traffic alert Collision Avoidance System (TCAS) and visual avoidance by pilots. It should be emphasized that while AAC will be responsible for resolving all routine conflicts, controllers will still be responsible for special

American Institute of Aeronautics and Astronautics

pilot requests and emergency situations. Figure 1 shows the conflict detection and resolution timeline for AAC. The times listed represent the predicted window of time before an impending conflict inside of which each safety layer operates.
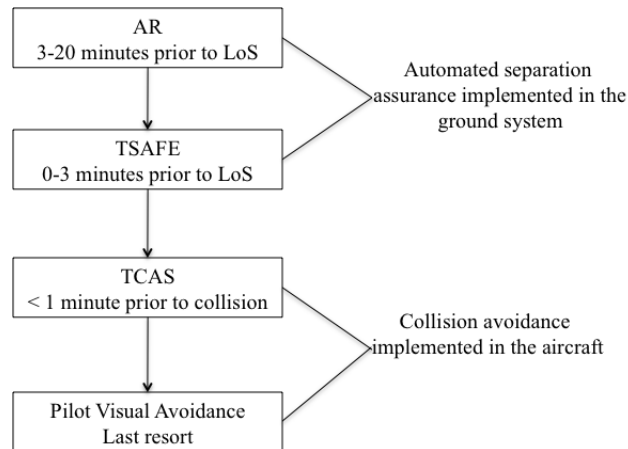

Figure 1: AAC conflict detection and resolution timeline

The Autoresolver (AR) is the workhorse of AAC. It is designed to detect and prevent losses of separation (LoS), defined as two aircraft having less than 5 nautical miles (nmi) of horizontal separation and less than 1000 feet (ft) of vertical separation, in the time frame of about 3 to 20 minutes. Its purpose is to handle nearly all of the conflicts that arise. It receives a constant feed of positional data for all aircraft in the NAS from radar as well as Global Positioning System (GPS) units onboard most aircraft, from which it generates trajectory predictions. Its conflict detection function probes the trajectory predictions two-at-a-time for all pairs of flights in the region of airspace for which it is responsible. Then, its conflict resolution algorithm searches for conflict resolution maneuver trajectories that are conflict-free for up to 20 minutes into the future. Fast-time simulations of AR indicate that it can handle the full breadth and variety of conflict situations that occur in enroute airspace including climbs and descents to arrival fixes [4-5, 12-15].

TSAFE is the primary backup system to AR and is also implemented on the ground. Its purpose is to detect and resolve conflicts in the shorter tactical time horizon of about zero to three minutes prior to LoS that were undetected at an earlier time to LoS and/or unresolved by AR. Since its main objective is to prevent LoS in this shorter time frame, it places a higher priority on safety over efficiency. It is designed to be a simple and reliable means of detecting and resolving tactical conflicts [16-18]. TSAFE is an autonomous system on the ground that runs in parallel to AR and relies on the same data. As such, it has more accurate resolutions compared to TCAS (described below). However, similar to TCAS, TSAFE issues a heading or altitude change to pilots via voice synthesizer.

TCAS is the safety net beneath AR and TSAFE. It is an airborne collision avoidance system that is intended to be independent of onboard navigation equipment and ground-based air traffic control systems, including AR and TSAFE [19]. Each TCAS-equipped aircraft interrogates all nearby aircraft about their respective current altitudes and also responds with its own altitude when interrogated. Each individual aircraft's TCAS system repeats the interrogation process continually to determine the range and range rate of nearby aircraft. An on-board computer analyzes the data to generate collision avoidance maneuvers for aircraft. Resolution advisories (climb or descend) are issued via cockpit display and synthesized voice commands. If this fails, the final safety layer is pilot visual avoidance. This is the last resort if the other automated separation assurance systems are all unable to detect and resolve an impending conflict. The AAC system is summarized in a high-level fault tree diagram in Figure 2 below.
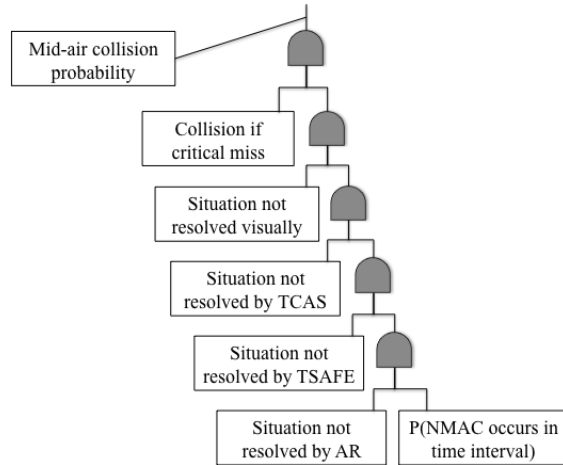
American Institute of Aeronautics and Astronautics

Figure 2: High-level fault tree diagram of AAC

III. Failure Analysis Methodology

A. AAC Subsystem Functional Block Diagram Models

An AR-TSAFE functional block diagram is illustrated in Figure 3, with additional detail shown in Figures 4 through 7. These diagrams are static representations of the combined AR and TSAFE subsystems where each individual block represents a component or process whose potential failure could impact the performance of the larger AAC system. The AR and TSAFE subsystems are combined because they both require positional data for each flight in order to function. This process is illustrated on the left side of Figure 3 and is referred to elsewhere in this paper as aircraft locatability. The key feature of these diagrams is that they convey which components/processes operate in series and in parallel, and thus identify failures that could cause an entire function or subsystem to fail.
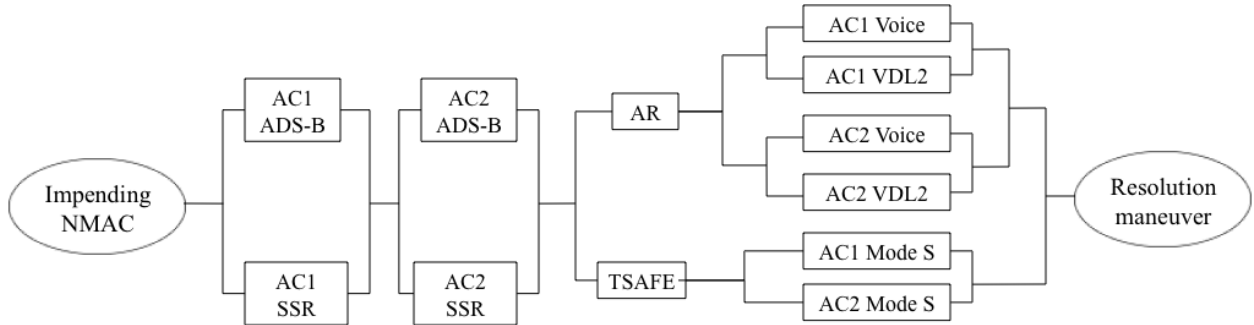

Figure 3: Overview of AR and TSAFE

AR and TSAFE obtain the location of the aircraft from Automatic Dependent Surveillance-Broadcast (ADS-B) or Secondary Surveillance Radar (SSR). ADS-B receives the aircraft's location from a Global Navigation Satellite System (GNSS) such as GPS and then transmits it via data link to the ground station [20]. SSR transmits interrogations to the aircraft's transponder, which replies promptly. The ground station detects the reply, computes the aircraft's location, and passes the location to the Host computer. This is illustrated in Figure 4. Alternatively, VHF Digital Link Mode 2 (VDL2) is a future data communication method with comparable performance that also could be used.
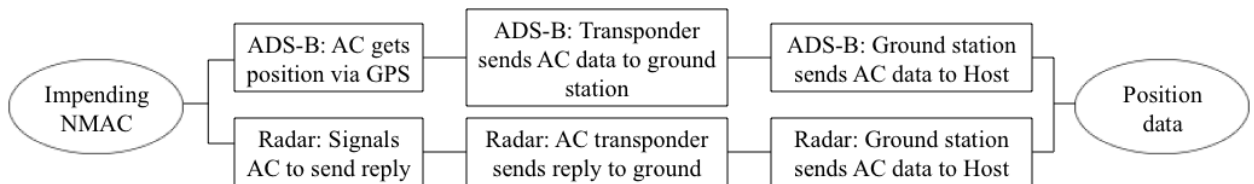

Figure 4: Detailed chart of ADS-B and SSR in parallel

AR and TSAFE generate trajectory predictions based on the position, velocity, and flightplan information for both aircraft. As seen in Figure 5, AR generates four-dimensional (4-D) flightplan trajectories for each flight that are used to determine if the flights will be in conflict 3 to 20 minutes in the future. A similar process is performed by TSAFE, which generates dead-reckoning trajectories to determine if two aircraft will be in conflict in the next one to three minutes. If either subsystem detects a conflict, then its respective conflict resolution module searches for a resolution maneuver.
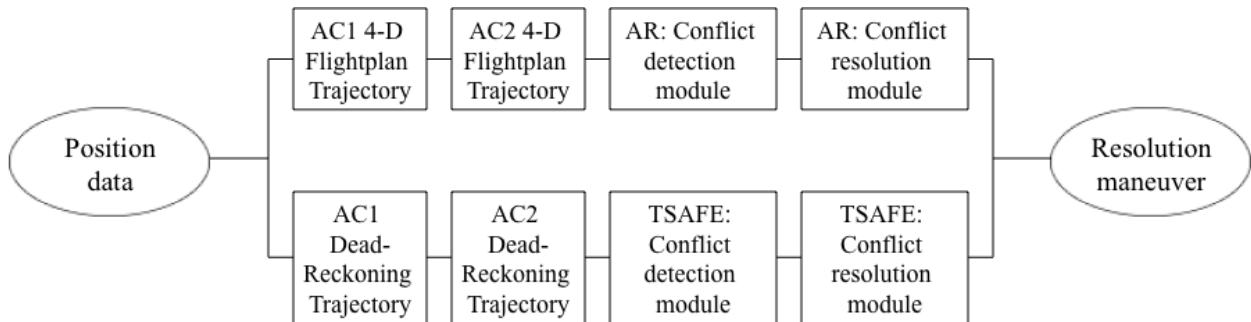

Figure 5: Detailed Chart of AR and TSAFE in parallel

When AR finds a conflict-free resolution, it is communicated to the aircraft primarily by data link through VHF Digital Link Mode 2 (VDL2), then to a resolution reader on-board the aircraft. It is assumed that the pilot checks the resolution trajectory, gives approval to the flight management system (FMS), and sends a 'wilco' to the ground system. The resolution reader sends the resolution to the aircraft, which in turn generates and executes the resolution trajectory. Alternatively, the resolution from AR may be sent by voice radio communication from a controller to the pilot.
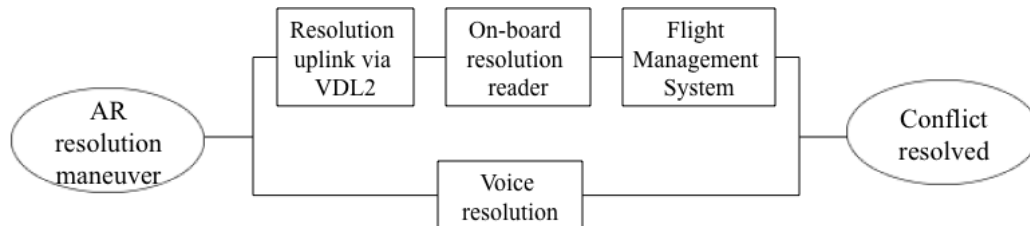

Figure 6: Voice/VDL2 for AR

On the other hand, when TSAFE finds a conflict-free resolution, it is communicated to the aircraft by the Mode S data link, which uses the aircraft's transponder. The message is then routed to a speaker on board the aircraft, which gives the resolution aurally. Although it is technically feasible for TSAFE resolution maneuvers to be communicated via voice from a controller to the pilot, this would not be as practical given TSAFE's shorter tactical time horizon of about one to three minutes. Note that VDL2 could also be utilized to uplink TSAFE resolution maneuvers if Mode S data link has failed or is not available, but it is not modeled here.
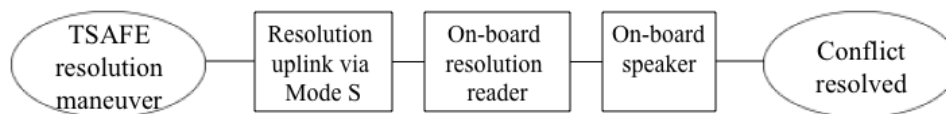

Figure 7: Mode S for TSAFE

TCAS is a mature system that has been operating in the national airspace system for several decades, and therefore empirical failure rate data exists for the system as a whole [9]. This analysis makes use of the empirical data and uses a fault tree decomposition of TCAS only to the extent necessary to take into consideration components that AR and TSAFE have in common. Two components in particular, the aircraft transponder and the speaker used for aural annunciations, are shown in order to facilitate identifying dependencies and common failure modes among the subsystems. All other TCAS components are treated as one block. Refer to Section D of the Appendix for a formal description of how this is done.

Not included in the model are events whose occurrence will cause failure of the subsystems of AAC but are not properly part of AAC. Three such events are wide spread power loss, computer hacking, and pilot non-conformance to resolutions given by the automation. These factors are important failure modes that are not directly attributable to the design of AAC, and so are omitted. A power outage will affect all ground systems in the outage area comprised of ADS-B and SSR systems, the computers for AR and TSAFE, and ground-based data link or voice communication equipment among other things. Hacking refers to any criminal interference with computer systems that causes these systems to be non-operational. Vulnerabilities include host computers receiving the aircraft position information as well as computers running AR and TSAFE. The third significant external factor is aircraft non-conformance. If AAC provides a resolution for an impending conflict, there is still a chance of collision if either aircraft does not execute the resolution correctly. AR may be able to detect conflicts resulting from non-conformance and issue a resolution maneuver if time to first loss is greater than three minutes. Furthermore, conflicts due to non-conformance will be detected by TSAFE given that it is functional. Andrews, Welch, and Erzberger address the service interruption and non-conformance failure modes [11]. The authors of this study are not aware of any comprehensive treatment of the vulnerabilities of air traffic control systems to power loss or computer hacking.

B. Fault Tree Logic

Fault trees for each AAC subsystem can be derived directly from the functional block diagrams illustrated in the previous section. The fault tree construction elucidates the logical interrelationships between various components and subsystems that lead to AAC failure. In this way, common failure modes are identified and preventative actions can be recommended. Detailed descriptions of the fault trees for aircraft locatability, AR conflict detection and resolution, TSAFE conflict detection and resolution, and TCAS functions can be found in Sections A-C of the Appendix.

The fault tree analysis yields over 100 failure modes for AAC. Instead of enumerating all of these failure modes, Table 1 lists only the components that are common to multiple failure modes. The most critical component is the onboard transponder. A flight's transponder is necessary for broadcasting its position data for use in AR, TSAFE, and TCAS, and it is also used to receive resolutions in TSAFE. If the transponder on either flight in a simulation trial fails, then AR, TSAFE, and TCAS cannot function to keep aircraft safely separated.

Component	Dependent Functions/Subsystems
Mode S Transponder	ADS-B: Sending location data to ground
	SSR: Sending altitude data to ground
	TSAFE: Resolution maneuver communication
	TCAS: Interrogations and range broadcast
Resolution Maneuver Reader	AR: Receives resolution from data link and routes to FMS
	TSAFE: Receives resolution from data link and routes to speaker
Speaker	TSAFE: Announces resolution to the pilot
	TCAS: Announces resolution to the pilot

Table 1: Common components to AAC failure modes

Note that the fault tree analysis method estimates failure probabilities in a static environment (by definition). Monte Carlo simulation analysis is used to extend the fault tree analysis and address the time-varying nature of the AAC system. For instance, the probability of conflict detection varies with time because trajectory prediction uncertainties decrease as a function of time-to-go (i.e. the remaining time before the conflict). The logic of the AAC subsystem fault trees is integrated into the Monte Carlo simulation in order to capture the interrelationships between AAC components and subsystems while enabling the time-dependent probabilities in the AAC system to vary.

IV.   Monte Carlo Simulation

The goal of this analysis is to estimate the time between mid-air collisions in enroute airspace given that AAC is used to detect and resolve impending collisions. To accomplish this, we construct a model from the perspective of two flights that will experience a NMAC eight minutes from now with complete certainty in the absence of a collision avoidance maneuver. Andrews, Welch, and Erzberger defined NMAC by considering the aircraft cross-section as rectangular, and set the NMAC vertical and horizontal miss distance as 500 ft and 100 ft respectively [11]. This analysis treats the probability of collision given NMAC as a simple Bernoulli event, which implies that the cross-sectional shape of the aircraft is immaterial. To simplify computations, this analysis uses a circular cross-

section for aircraft, and defines NMAC to be two aircraft passing within 100 ft of one another. A detailed explanation is given in Section G of the Appendix. Note, however, that this definition of NMAC is probably more limited than what pilots report, and so does not correspond neatly with FAA reference data.

Monte Carlo simulation is employed to resolve what happens to the two aircraft that will experience NMAC, where each simulation trial represents a situation involving two flights that will experience NMAC eight minutes in the future without air traffic control intervention. An eight-minute maximum first detection time is used because previous research on conflict prediction algorithms found that detections made with predicted time to first loss of separation of eight minutes or less were accurate enough for resolutions to be issued without an excessive false alarm rate [21-22]. The time between each NMAC follows a Brownian motion model described below in Section IV, Subsection B. After simulating 10 billion trials together with the time since the last trial (in accordance with the Brownian motion model), we compute the sample mean and sample variance of the time between trials where the result was a safety-critical event such as an NMAC or a mid-air collision.

Each individual trial consists of three parts. First, the simulation resolves the time into each flight that the NMAC will take place, determines at what times during each flight AAC components are functional, and employs the fault tree logic described in Section III to determine which AAC subsystems are functional during the eight minutes prior to NMAC. Second, the simulation initiates an eight-minute countdown until NMAC occurs and simulates the time until detection and avoidance, where AR operates in the three- to eight-minute minute time horizon, TSAFE operates in the one- to three-minute horizon, and TCAS operates in the zero- to one-minute horizon. If any AAC subsystem fails, the countdown skips ahead. Third, if detection, resolution, and avoidance do not occur by the time the countdown reaches zero, NMAC occurs and the simulation determines whether or not an actual collision occurs given the NMAC. A tally is kept of the numbers of trials resulting in safety-critical events as well as the times since previous safety-critical events to enable the computation of sample statistics.

A. Simulation Assumptions and Simplifications

The inherent complexity of the NAS and AAC require various assumptions and simplifications to be made for the Monte Carlo simulation. Certain assumptions ought to be examined in future research because they can affect estimates of the rates of safety-critical events, for example, the assumption that the probability of detecting a conflict only depends upon look-ahead time. A complete list of assumptions is given in the Section E of the Appendix.

B. Estimating the Rate of NMAC under No Air Traffic Control

The first step in the Monte Carlo simulation is to estimate the rate at which NMACs would occur in the NAS in the absence of air traffic control. Following the analogy to Brownian motion that was utilized by Andrews, Welch and Erzberger [11], NMAC events are independent, identically distributed based on a Poisson distribution whose rate parameter is a function of traffic density, aircraft cross-section and relative speed, λ_{NMAC}. From the perspective of a single flight in a fixed volume containing a total of n flights, the rate at which this flight experiences NMAC is:

$$\lambda_{NMAC/AC} = 4DFE[v_{ij}]A_{NMAC} \tag{1}$$

The parameters used in Eqn. 1 are defined in Table 2. Each of the n aircraft has n-1 potential collisions it could experience. Therefore, λ_{NMAC} when summed over all n aircraft is:

$$\lambda_{NMAC} = \lambda_{NMAC/AC} \frac{(nf-1)s}{2} \tag{2}$$

Parameter	Description	Value
D	Current traffic density	0.001 AC/nmi^3
E[v$_{ij}$]	Mean velocity between two flights	560 nmi/hr
A$_{NMAC}$	Average cross-sectional area of NMAC	0.00385 (nmi^2)
s	Number of enroute sectors in NAS	200
n	Average number of aircraft in a sector (maximum for 1x traffic)	12
f	Future traffic demand (relative to today)	3

Table 2: Values used to determine rate parameter of NMACs

The parameter λ_{NMAC} used in all simulations was for 3x current traffic density (i.e., f = 3) for a rate of 90.5 NMACs per hour over the NAS absent air traffic control intervention. As a point of reference, the observed rate of NMACs in today's NAS that can be attributed to air traffic control operational error is about nine per year [7-8].

C. Simulation Initialization

At the beginning of the Monte Carlo simulation, a variable to store the absolute amount of time that has elapsed since the start of the simulation is initialized with a value of zero. After each simulation trial, we add to that variable the time between successive simulation trials, which is randomly generated based on a continuous exponential random variable with parameter λ_{NMAC} (computed in the previous subsection).

D. Simulation Trial Run

Each simulation trial proceeds according to the logic detailed below.

➢ Determine how far into each of two flights the NMAC occurs, using a uniform distribution.

➢ Initialize a clock that counts backwards starting at 8 minutes prior to NMAC, which will keep track of the passage of time as AR and TSAFE each attempt successively to detect and resolve the impending NMAC (declare variable t = 8). During the interval (3, 8] minutes, AR attempts detection and resolution. TSAFE then attempts detection and resolution in the interval (1, 3] minutes. If the clock reaches 1 minute, we assume LoS has occurred and simulate whether TCAS detects and resolves the conflict, or NMAC or mid-air collision occurs.

➢ Determine which components of AAC fail by simulating according to known failure rates. Components whose failures are simulated are listed in Tables A1, A2, and A3 in the Appendix. If a component fails, simulate the time of component failure using a uniform distribution. These failures are only in effect in the simulation if they occur prior to the time into each flight when the NMAC is determined to occur, as described above.

➢ Use fault tree logic presented in Sections A, B, and C of the Appendix to determine first if both aircraft are locatable, second if AR is functional, and third if TSAFE is functional. If either aircraft is not locatable, advance the clock to t = 1. If AR is not functional, advance the clock to t = 3. If TSAFE is not functional, once the clock reaches t = 3, immediately advance it to t = 1.

➢ If t > 3, attempt to generate 4-D trajectories for both aircraft by generating two discrete random numbers Bernoulli distributed with the same probability p_{FPT} = 0.000001. If trajectories are generated for both aircraft, then attempt to detect the LoS that is t-1 minutes out by generating a discrete random number Bernoulli distributed by probability $p_{ACD}(t)$. See Section H of the Appendix for a description of how to compute $p_{ACD}(t)$. If AR fails either to generate 4-D trajectories for both aircraft or to detect the conflict, subtract 30 seconds from the clock and repeat both steps (i.e. set t = t - 0.5). This process stops when t = 3. Note that AR has ten attempts to detect the conflict given that it is functional, and each successive attempt has less prediction uncertainty than the previous attempt. If AR succeeds in detecting the conflict, then the conflict is resolved, since it was already determined that AR is capable of uplinking the resolution maneuver. Begin a new trial. If not, continue.

➢ If t = 3, determine whether or not TSAFE successfully overrides AR by generating a discrete random number Bernoulli distributed with probability p_{TSOAR} = 0.000001. If over-ride is unsuccessful, set t = 1, otherwise continue.

➢ If 1 < t ≤ 3, attempt to generate dead reckoning trajectories for both aircraft by generating two discrete random numbers Bernoulli distributed with the same probability p_{DRT} = 0.000001. If trajectories are successfully generated for both aircraft, attempt to detect the LoS that is t - 1 minutes out by generating a discrete random number Bernoulli distributed by probability $p_{TCD}(t)$, where 0 represents failure to detect the LoS. We assume that $p_{TCD}(t) = p_{ACD}(t)$. While this might not be strictly true, the polynomial function for both probabilities is fitted using empirical data, so the different trajectory generation schemes used by AR and TSAFE are reflected in the polynomial's coefficients. If TSAFE either fails to generate dead reckoning trajectories for both aircraft or fails to detect the conflict, subtract 30 seconds from the clock and repeat both steps (i.e. set t = t - 0.5). This process continues until t = 1, at which time a LoS is assumed to occur. If TSAFE succeeds in detecting the conflict, the conflict is resolved, since it was already determined that TSAFE is capable of uplinking the resolution maneuver. Begin a new trial. If not, continue.

➢ If t = 1, determine whether or not TCAS successfully over-rides TSAFE by generating a discrete random number Bernoulli distributed with probability p_{TCOTS} = 0.000001. If over-ride is unsuccessful an NMAC occurs with near certainty. Skip ahead to the collision determination. Determine if TCAS systems on both aircraft are

functional by generating a discrete random number Bernoulli distributed with probability p_{OTHER} = 0.100112. Apply the probabilistic reasoning described in Section D of the Appendix and illustrated in Figure A4 to discover if TCAS has failed on either aircraft for any reason. Without a functional TCAS it is conceivable that the pilot of an aircraft could visually detect and avoid a second aircraft that he is on course to hit, but given the extremely high relative airspeed and absence of any warning, such an event can be regarded an act of providence. Skip ahead to the collision determination.

➢ If TCAS systems on both aircraft are functional and TCAS over-rides TSAFE, determine whether or not the pilots avoid the impending NMAC with the assistance of TCAS by generating a discrete random number Bernoulli distributed with probability p_{VA} = 0.3. This parameter is the joint probability that TCAS collision avoidance logic works as designed and that the pilots translate a given collision avoidance alert into action. If collision avoidance is successful, begin a new trial. If it fails, the NMAC occurs. Continue on to the collision determination.

➢ Given that the NMAC occurs, generate a discrete random number Bernoulli distributed with probability p_{CNMAC} = 0.28. The reasoning behind this calculation is that the two aircraft will pass within a certain number of feet of one another, and whether or not they impact each other is no longer in the hands of the AAC system or the pilots. Collision is now a matter of chance, and can be resolved using simple geometry. See Section G of the Appendix for a description of how p_{CNMAC} is calculated. If collision does not occur, begin a new trial. If collision does occur, record the time since the previous collision and begin a new trial, or if it is the first collision, record the time since the simulation was initialized and then begin a new trial.

V. Simulation Results

Presented below are the results of a single Monte Carlo simulation with 10 billion trials. A large number of trials was necessary so as to generate at least 100 instances of safety-critical events such as TCAS alerts, NMAC, and mid-air collision. For AR to resolve a conflict, it needs to be able to locate both aircraft by either ADS-B or SSR, generate 4-D flightplan trajectories for both aircraft, detect the conflict and find a conflict-free resolution for it, and upload the resolution maneuver via VDL2 data link or voice commands. In this Monte Carlo simulation run, AR successfully detected and resolved conflicts 99.98% of the time. Analysis of the remaining .02% (2,082,241 trials) is summarized in Table 3. In nearly all cases, the reason that AR failed to resolve the conflict is that it was unable to detect the conflict.

# of Trials AR Unable to Resolve Conflict (out of 10 billion): 2,082,241		
Reason: At Least One Aircraft not Locatable	Reason: No AR Detection (though both aircraft locatable)	Reason: No AR Resolution (though both aircraft locatable and successful detection)
0.05%	99.47%	0.47%

Table 3: Breakdown of reasons AR unable to resolve conflict

Of these 2,082,241 cases where AR did not resolve the conflict, approximately 99.92% were resolved by TSAFE. For TSAFE to resolve the conflict, it needed to be able to locate both aircraft via ADS-B or SSR, generate 4-D flightplan trajectories for both aircraft, detect the conflict and find a conflict-free resolution, and upload the resolution maneuver via Mode S data link. Analysis of the other .08% (1491 trials) found that about 74% resulted from failures to locate and track aircraft, and all but one of the remaining cases were attributable to conflict detection failures. These results are summarized in Table 4.

# of Trials TSAFE Unable to Resolve Conflict (out of 10 billion): 1491		
Reason: At Least One Aircraft not Locatable	Reason: No TSAFE Detection (though both aircraft locatable)	Reason: No TSAFE Resolution (though both aircraft locatable and successful detection)
73.98%	25.96%	0.08%

Table 4: Breakdown of reasons TSAFE unable to resolve conflict given AR unable to resolve conflict

Table 5 shows what happened in each of the 1491 cases that required TCAS and pilot visual avoidance. There were 311 cases that were successfully resolved by TCAS and 829 by pilot visual avoidance; as such, neither NMAC nor collision occurred. The number of cases resolved by TCAS is relatively low because about 75% of the 1491 cases analyzed for Table 5 involved transponder failure on at least one flight. Another 262 cases were not resolved

by either TCAS or pilot visual avoidance and resulted in NMAC but not collision. Assuming that AAC would need to handle traffic equal to 3 times density in today's busiest sectors about 12 hours each day, the mean time between these NMAC-only events is estimated to be about 96 years. In comparing this figure to the current reference point of 9 controller-attributable NMACs per year, recall that for purposes of this study an NMAC is defined as separation less than or equal to 100 ft, while the reference data uses subjective pilot reporting. The remaining 89 cases were unresolved by TCAS and pilot visual avoidance and resulted in mid-air collision. Using the same assumption regarding the distribution of traffic over time as in the NMAC-only case, the mean time between these collision events is about 281 years. Of the trials that resulted in mid-air collision, more than 85% involved failure to locate and track aircraft through radar or satellite positioning systems, and the rest involved conflict detection failure only.

	TCAS Resolves	VA Resolves	NMAC Occurs	Collision Occurs
Number of Trials (out of 10 billion)	311	829	262	89
Mean Time between Events	81.3 years	30.5 years	95.7 years	281.4 years
Sample Standard Deviation of Time Between Events	77.3 years	30.3 years	95.6 years	244.0 years

Table 5: Safety-critical trials involving TCAS resolution, NMAC, or mid-air collision

VI. Simulation Results Discussion

The results of this study suggest that the current design of AAC does provide a level of safety that is around the desired threshold for mean time between mid-air collisions [11]. However, because of computational limitations, only 89 collision inter-arrival times were obtained, leading to a sample variance that is quite high. As a point of reference, the NTSB Aviation Accident Database lists 91 distinct accidents involving aircraft inside the NAS occurring between November 1983 and July 2010, with the key phrase "midair collision" in the record [26]. The data, while suggestive of a collision rate that is many orders of magnitudes higher than the simulated midair collision rate using AAC, is not directly comparable. It includes collisions involving military aircraft, aircraft in the takeoff and landing phases of flight, aircraft not equipped with TCAS II, and aircraft not detectable by SSR.

Several important factors were excluded from this study that merit consideration in future research. These include: (1) situations where AR or TSAFE issues a maneuver that resolves a conflict between two aircraft but unknowingly causes a secondary conflict with a separate aircraft whose location was unknown to the automation; (2) external factors such as power failures and computer hacking; and (3) the possibility that pilots do not conform to their flightplan or correctly perform resolution maneuvers, which is especially significant when pilots fly beyond their assigned flight level.

The model used in this study draws on many inputs, any one of which could be studied further to improve its accuracy. Two inputs in particular deserve attention. The first is the rate at which NMACs would occur absent any air traffic control. This input is fundamental to the model and directly drives the performance measure. NASA Ames is able to estimate λ_{NMAC} through simulation using its Airspace Concept Evaluation System (ACES)[12-15]. The result of an ACES simulation will be more accurate than the estimate used here. The second is the failure rate of the Mode S transponder. Fault tree analysis of the AAC system indicates that failure of the transponder would cause AR, TSAFE, and TCAS all to fail. It is a critical component, which is why 74% of the trials that were unresolved by AR and TSAFE were the result of failure to locate and track a flight. Aircraft typically have two transponders, but switching between the two does not occur automatically, diminishing the effectiveness of the redundancy. While it would have been straightforward to simulate the time until a crew member notices a transponder failure and switches to the backup, the distribution of this random variable has never been studied, and so would have been arbitrary. The chosen approach to model transponder redundancy was instead to consider the total time between transponder failures to be one order of magnitude less than the reported time for a single transponder. It should be pointed out, however, that these failures would also have the same effect on a manual system and are not specific to AAC. Notwithstanding, it is important to perform a sensitivity analysis on these parameters.

In this study, we did not posit any particular computer platform on which AR and TSAFE will reside. However we did make the assumption that sufficient redundancy exists within the ground-based computer hardware that the only plausible cause of total hardware failure would be an external event such as computer hacking, terror attack, or natural disaster. The FAA is currently replacing its NAS computer backbone with the En Route Automation Modernization (ERAM) computer system, and it is logical that ERAM would serve as the computer platform to host

AR and TSAFE. ERAM is designed to function as part of the FAA's "System Wide Information Management" program, which will allow computers at FAA air traffic control centers across the country to back each other up via a data network [24]. ERAM is currently being tested at multiple FAA centers including Seattle and Salt Lake City [25].

Finally, the results of this study highlight the importance of the extent to which trajectory uncertainty affects AR and TSAFE. In instances where AR and TSAFE failed to resolve a conflict, the most significant source of failure besides failure to locate aircraft was the ability of AR and TSAFE to successfully detect the conflict. We therefore recommend that if resources are to be invested to improve the design of AAC, those resources go toward (1) improving the redundancy of transponders onboard aircraft, and (2) enhancing conflict detection by improving trajectory prediction accuracy. There are several ways the latter could be accomplished. First, there is room to improve the trajectory prediction logic itself, in particular by taking advantage of data provided by ADS-B to incorporate more precise velocity estimates. Second, flight performance models could be improved, especially with regard to climb and descent. Third, new or existing weather models could be linked to trajectory generation to take into consideration uncertainty due to atmospheric conditions. Finally, various policies could be implemented to reduce trajectory uncertainty, such as requiring aircraft to disclose their gross weight prior to takeoff and requiring flights to conform more tightly to their flightplans both vertically and laterally.

The following are recommended areas for future research to improve the fidelity of the failure estimates presented in this study, as well as to assess its robustness to the driving parameters:

    a) Estimate uncontrolled NMAC rate using ACES simulation
    b) Perform sensitivity analysis on transponder failure probability
    c) Develop a model for pilot non-conformance to resolution maneuvers, possibly focusing on human factors such as cockpit distractions and instrument ease-of-use
    d) Define distinct spheres of responsibility for when AAC is responsible for safety-critical events such as loss of separation, near mid-air collision, and mid-air collision
    e) Perform risk analysis of external factors, especially catastrophic power failures and computer hacking
    f) Model weather events such as electromagnetic interruption of aircraft data transmission
    g) Relax the assumption that probability of detection is a function of look-ahead time only

VII.   Conclusions

A probabilistic risk analysis of the Advanced Airspace Concept, a candidate next-generation air traffic control system, was conducted using Monte Carlo simulation. Results indicate that the workhorse of AAC, the Autoresolver, was able to successfully detect and resolve conflicts and communicate resolution maneuvers to pilots in the strategic time frame of three to eight minutes prior to collision about 99.98% of the time. Analysis of the remaining 0.02% of the trials found that the next safety layer, Tactical Separation-Assured Flight Environment, resolved 99.92% of outstanding conflicts in the tactical time frame of one to three minutes prior to collision.

The 1491 conflicts that were not resolved by the Autoresolver or TSAFE were also analyzed. TCAS resolved 311 of these cases, and pilot visual avoidance was successful in another 829 cases; as such, neither NMAC nor mid-air collision occurred in these cases. However, 262 cases were not resolved by TCAS or visual avoidance and resulted in NMAC but not collision. Assuming that the system would have to handle traffic equal to three times current density in today's busiest sectors about 12 hours each day, the mean time between NMAC events was estimated to be approximately 96 years. Mid-air collision resulted in the remaining 89 conflicts in which Autoresolver, TSAFE, TCAS, and pilot visual avoidance all failed. The mean time between these mid-air collision events in enroute airspace was about 281 years, which meets the desired performance level for an automated separation assurance function. Of the simulation trials that ended with mid-air collision, most involved failure to locate and track aircraft via radar or satellite positioning systems, and the rest were due to conflict detection failure.

Appendix

A.   Location of AC1 (or AC2)

Flights are considered locatable if and only if they can be tracked either via ADS-B or SSR. A flight is only considered locatable via ADS-B if all of the following hold true: 1) aircraft receives position data via GPS; 2) aircraft broadcasts data to ground station via data link; 3) ground station sends the flight's data to Host computers. Likewise, a flight is only deemed locatable via SSR if all of the following hold true: 1) aircraft receives signal from ground station to send its data; 2) aircraft uses its transponder to send data to the ground; 3) aircraft data is sent from ground station to Host computers. If one or both flights are not locatable, then the conflict cannot be handled strategically by AR or tactically by TSAFE. Then, the last two safety layers available in the model are TCAS and

pilot visual avoidance. It should be noted that SSR does provide a "skin paint" return in the event of transponder failure that could also be used by air traffic controllers, but it is not modeled here.
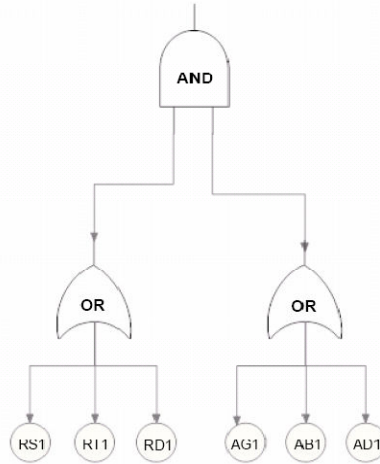

Figure A1: Aircraft 1 locatability

Component	Description	Probability (failure)
AG1	ADS-B: AC1 gets position via GPS	0.0005 (ref. [20])
AB1	ADS-B: AC1 broadcasts data to ground station	0.0000097 (ref. [9])
AD1	ADS-B: Ground station sends AC1's data to Host	0.00002 (ref. [20])
RS1	Radar: Ground signals AC1 to reply	0.00682 (ref. [9])
RT1	Radar: AC1 transponder replies to ground	N/A (RT1=AB1)
RD1	Radar: Ground station sends AC1's data to Host	N/A (RD1=AD1)

Table A1: Aircraft locatability components and failure probabilities for an aircraft (AC1)

B. Strategic Detection, Resolution, and Maneuver Communication (Autoresolver)

Assuming that the track data for both flights are available, then the strategic layer of AAC (i.e., Autoresolver) is functional only if all of the following hold true: 1) flightplan-based trajectories can be computed for both flights (i.e., FPT1 and FPT2); 2) strategic conflict detection and resolution (i.e., ACD and ACR) are successful; 3) the strategic resolution maneuver can be communicated by voice (i.e., VC1 or VC2) or datalink (i.e.,VDL2, RR, and FMS all operational on either flight). If any of these strategic functions fails, then Autoresolver fails.
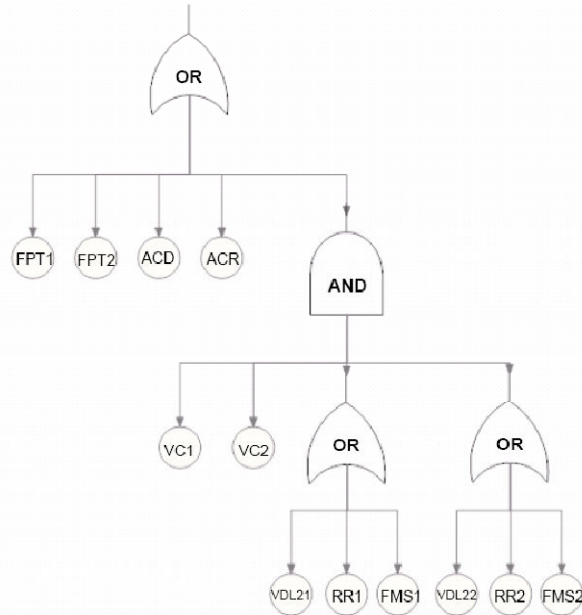
Figure A2: AR detection, resolution, and communication

Component	Description	Probability (failure)
FPT1	Generate 4-D flightplan trajectory (AC1)	0.000001 (proxy from ref. [11])
FPT2	Generate 4-D flightplan trajectory (AC2)	0.000001 (proxy from ref. [11])
ACD	Autoresolver: Conflict detection module	Varies (see Appendix, Section H)
ACR	Autoresolver: Conflict resolution module	0.000001 (ref. [11])
VDL21	VDL2: Resolution upload via data link (AC1)	0.00004 (ref. [23])
VDL22	VDL2: Resolution upload via data link (AC2)	0.00004 (ref. [23])
VC1	AC1's data sent to host comps via voice communication	0.00055 (ref. [9])
VC2	AC2's data sent to host comps via voice communication	0.00055 (ref. [9])
RR1	Onboard resolution reader (AC1)	0.000001 (ref. [23])
RR2	Onboard resolution reader (AC2)	0.000001 (ref. [23])
FMS1	Onboard resolution trajectory generator (AC1)	0.000097 (proxy from ref. [9])
FMS2	Onboard resolution trajectory generator (AC2)	0.000097 (proxy from ref. [9])

Table A2: AR components and failure probabilities

C. Tactical Detection, Resolution, and Maneuver Communication (TSAFE)

If the aircraft is locatable but Autoresolver fails to resolve the conflict for whatever reason, then it is the responsibility of the tactical portion of AAC (i.e., TSAFE) to handle the conflict. TSAFE is quite similar in structure to the strategic layer in that it only functions if 1) dead-reckoning trajectories can be calculated for both flights (i.e., DRT1 and DRT2); 2) tactical conflict detection and resolution (i.e., TCD and TCR) are successful; 3) the tactical resolution maneuver is communicated via Mode S to an on-board resolution reader that translates the maneuver and produces an aural command for the pilot to execute (i.e., MS, RR, and S on either flight). If any of these tactical functions fail, then the tactical portion of AAC fails.
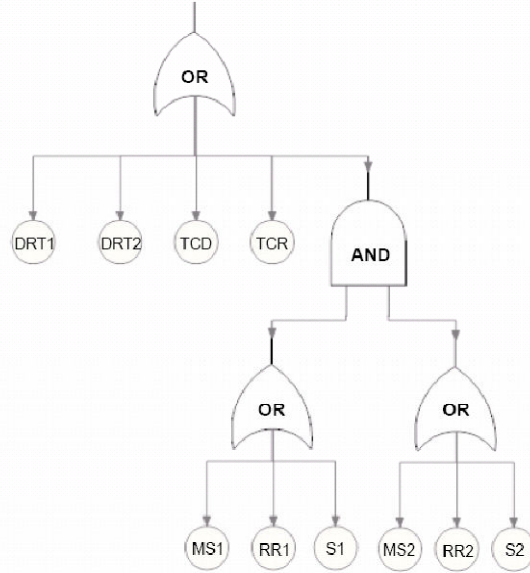
Figure A3: TSAFE detection, resolution, and communication

Component	Description	Probability (failure)
DRT1	Generate 4-D dead reckoning trajectory (AC1)	0.000001 (proxy from ref. [11])
DRT2	Generate 4-D dead reckoning trajectory (AC1)	0.000001 (proxy from ref. [11])
MS1	Mode S: Resolution upload via data link (AC1)	N/A (MS1=AB1)
MS2	Mode S: Resolution upload via data link (AC2)	N/A (MS2=AB2)
TCD	TSAFE: Conflict detection module	Varies (see Appendix, Section H)
TCR	TSAFE: Conflict resolution module	0.000001 (assumed)
RR1	Onboard resolution reader (AC1)	N/A (Same as RR1 in Table A2)
RR2	Onboard resolution reader (AC2)	N/A (Same as RR2 in Table A2)
S1	Onboard speaker (AC1)	0.000001 (assumed)
S2	Onboard speaker (AC2)	0.000001 (assumed)

Table A3: TSAFE components and failure probabilities

D. Estimating TCAS Failure Probability

While both AR and TSAFE are still being developed, the aviation community has almost 30 years worth of empirical data on the failure of TCAS. However, because two subcomponents associated with TCAS, the speaker and the Mode S transponder, are also associated with AR and TSAFE, it is necessary to think of TCAS as comprising three components in series: speaker, Mode S transponder, and all other components.

Define:

p_{TCAS} = the total probability that TCAS fails for a pair of aircraft, either as TCAS fails on AC1, TCAS fails on AC2, or TCAS fails on AC1 and AC2 = 0.2

$p_{TCAS\ ONE\ FLIGHT}$ = the total probability that TCAS fails on a single AC

p_S = the probability that the speaker fails (known from certification) = 0.000001

p_{RT} = the probability that the Mode S transponder fails (known from certification) = 0.000097

p_{OTHER} = the probability that TCAS fails for reasons excluding speaker and transponder failure

Assume that p_S, p_{RT}, and p_{OTHER} are independent. Represent p_{TCAS} as the union of $p_{TCAS\ ONE\ FLIGHT\ (AC1)}$ and $p_{TCAS\ ONE\ FLIGHT\ (AC2)}$

$$p_{TCAS} = 2(p_{TCAS\ ONE\ FLIGHT}) - (p_{TCAS\ ONE\ FLIGHT})^2$$
$$p_{TCAS\ ONE\ FLIGHT} = 0.105557$$

Consider the event space in Figure A4, and note the following relationship:

$$p_{TCAS\ ONE\ FLIGHT} = p_S\ U\ p_{RT}\ U\ p_{OTHER} = p_S + p_{RT} + p_{OTHER} - p_S p_{RT} - p_{RT} p_{OTHER} - p_S p_{OTHER} + p_S p_{RT} p_{OTHER}$$

$$p_{OTHER} = \frac{p_{TCAS\ ONE\ FLIGHT} - p_S - p_{RT} + p_S p_{RT}}{1 - p_S - p_{RT} + p_S p_{RT}} = 0.10549$$
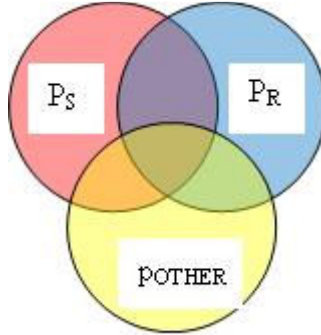

Figure A4: Event space representing p$_{TCAS}$, decomposed into three overlapping regions

And because TCAS failure is memoryless (it is follows a Poisson process), correct p$_{OTHER}$ according to
$$p_{OTHER\ CORRECTED} = 1 - e^{-P_{OTHER}} = 0.100112$$

The logic behind this correction is presented later in the Section F in the Appendix. Its purpose is to allow us to resolve whether or not TCAS fails for this particular flight by generating a random number using a Bernoulli random variable.

E. Simulation Assumptions
- Given no air traffic control, near-mid-air collisions (NMAC) are independent, identically distributed events based on a Poisson distribution whose rate parameter is a function of traffic density, aircraft cross-section and relative speed (following Andrews, Welch, and Erzberger's analogy to Brownian motion [11])
- Simulation trials are NMAC if AR, TSAFE, and TCAS all fail; some of these are also mid-air collisions
- Failures of AAC components across flights are independent and identically distributed
- Average flight time is 2 hours
- Flights are locatable either via ADS-B or SSR, but primary radar is not used
- Maximum look-ahead time for conflict detection and resolution is 8 minutes
- AR and TSAFE perform conflict detection and resolution and issue maneuvers in 30-second intervals
- If a conflict-free resolution is found and successfully communicated, it is executed correctly, resolves the conflict in all traffic situations, and there is no conflict between the same flight pair in the future
- Probability of detecting a conflict only depends upon look-ahead time (e.g., flight phase is not a factor)
- Altitude intent is known by all AAC subsystems at all times
- AR resolutions can be communicated by either voice (via human controller) or VDL2
- TSAFE only generates and utilizes dead-reckoning trajectories
- TSAFE resolutions can only be communicated via Mode S transponders
- TCAS II is deployed on both flights involved in each simulation trial
- AR and TSAFE ground-based computer hardware is sufficiently redundant that total failure is only possible given an external event (e.g., power failure, computer hacking, natural disasters, terrorist attacks, etc.)
- The effect of other aircraft in the vicinity of the conflict flight pair is negligible
- Three SSR radar units cover each point in the NAS on average
- The mean relative airspeed of two flights is 560 knots (based on 400 knots mean airspeed of any one flight, and uniformly distributed crossing angle)
- An upper bound on commercial aircraft cross-section is 10,000 ft^2
- The most dense sector the NAS at current traffic levels has a mean density less than 0.001 AC/nmi^3
- The current maximum number of aircraft allowable per sector is 12

F. Computation of Failure Probabilities

Where literature reports failure data for an AAC component in the form of a single, fixed value that is independently distributed over all flights, component failure occurs according to a Bernoulli distribution. More commonly, a component's probability of failure is reported in the form of mean time between failures, and is Poisson distributed. In this case, we assume that the average flight length in the United States is 2 hours, and that the component can only wear down and fail during flight. We convert the mean time between failures from hours to flights by dividing the reported value by 2, and derive the failure rate parameter as follows:

Let $T_{i\,MTBF}$ be the component i's mean time between failures, where time is measured in flights.

$1 / T_{i\,MTBF}$ is the rate of component failure per flight.

The time between failures is a random variable with cumulative distribution $F_x(x) = 1 - e^{(-x/T_{iMTBF})}$

Using the same definition of p_i as before, p_i is the probability that component i has failed before the end of the flight, given that it was functional when the flight departed. This probability is simply $F_x(x)$ evaluated at x=1.

$$p_i = 1 - e^{(-1/T_{iMTBF})}$$

According to L'Hopital's Rule, in the limit as $1/T_{i\,MTBF}$ approaches 0, $p_i = 1 / T_{i\,MTBF}$

We find that in practice, for $1/T_{i\,MTBF} < 0.001$, the difference is insignificant, and therefore we approximate $p_i \approx 1 / T_{i\,MTBF} < 0.001$

Where $1 / T_{i\,MTBF} > 0.001$, we compute p_i as $p_i = 1 - e^{(-1/T_{iMTBF})}$, and determine whether or not component i has failed according to a Bernoulli random variable.

G. Computation of Aircraft Cross-Section

Andrews, Welch, and Erzberger defined NMAC by considering the aircraft cross-section as a rectangular, and set the NMAC vertical and horizontal miss distance as 500 ft and 100 ft respectively [11]. This logic is intuitive, but it is not an ideal way to compute P(Collision | NMAC). Instead, imagine two aircraft AC1 and AC2 both have circular cross sections with radius x, and define NMAC to be an event where AC1 and AC2 pass by each other with separation less than or equal to y. This method simplifies the calculation of the probability conditioned on a NMAC without loss of generality and accuracy as long as there is uniform probability that the center of AC2 passes through any particular point inside A_{NMAC} which surrounds AC1. This is because p(Collision | NMAC) reduces to a ratio of two areas, and whether these areas are shaped as circles or airplane silhouettes makes no difference.

We claim that: $P(Collision|NMAC) = \frac{\pi(2x)^2}{\pi(2x+y)^2} = \left(\frac{2x}{2x+y}\right)^2$

Proof: The maximum distance between two aircraft that would result in a NMAC is defined to be y. If the cross section of each aircraft has radius x, it is readily apparent that the maximum distance between the centers of both aircraft which would result in a NMAC is 2x+y. Thus A_{NMAC} is a circle with radius 2x+y. For collision to occur, there must be a distance of 0 between some point inside or on the surface of each of the two aircraft. This implies that the maximum distance between the centers of both aircraft is 2x. If the centers of each aircraft are uniformly distributed inside A_{NMAC}, then P(Collision | NMAC) is the ratio of a circle with radius 2x to A_{NMAC}, as illustrated in Figure A5. We define y to be 100 ft. To compute x, we use the cross section of a B777, which is about 10,000 ft^2, and find the radius of a circle with area 10,000 ft^2 is 56 ft. A_{NMAC} is found to be 141,192.6 ft^2, and P(Collision | NMAC), which we also refer to as p_{CNMAC} is found to be 0.28.
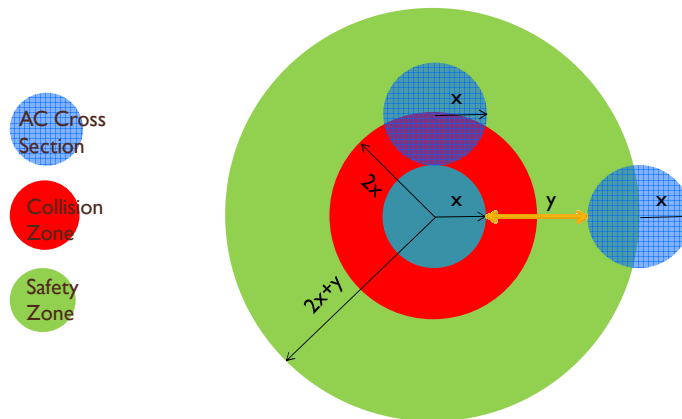

Figure A5: Circular region representing A$_{NMAC}$

American Institute of Aeronautics and Astronautics

H. Estimating Probability of Detection

Each simulation trial consists of two flights that will experience NMAC given no air traffic control intervention. The probability that AR or TSAFE can detect this type of conflict is estimated as a function of time-to-go. The estimation utilizes data for about 500 level flights in Fort Worth Center on September 28-29, 2008 from 8:00 AM to 10:00 PM, local time. The first step is to compute along-track and cross-track trajectory prediction errors at 12-second look-ahead time increments. This was done by generating trajectory predictions using the CTAS Trajectory Synthesizer module and comparing them against actual radar tracks. We required level flights to satisfy the following conditions to be included in the estimation: 1) present in Fort Worth Center airspace for at least 1 minute, 2) cruise altitude at or above 18,000 ft, and 3) no recorded flightplan amendments for at least 12 minutes after entering Fort Worth Center airspace (e.g., no climb or descent segments).

The probability that AR and TSAFE is able to detect a conflict at a given time is a function of the probability that the predicted positions of both flights fall into the LoS region that surrounds their points of closest approach. Because the minimum separation distance between two aircrafts set by FAA, 5 nmi, is much larger than the distance inside which a NMAC is said to have occurred, 100 ft, the points of closest approach for these two aircraft can be approximated as a single point in space, referred to hereafter as the NMAC point.

For a given look-ahead time t *(sec)* in the future, we calculate the proportion $p(t, d)$ of flights that pass within a distance d *(nmi)* of its actual location t seconds in the future. This is equal to the probability that the predicted location for one aircraft t seconds in the future falls into a circular region with radius d and the NMAC point as its center. This sample portion $p(t, d)$ is plotted against d for t= 12 sec, 120 sec, and 600 sec in Figure A6.

Recall that LoS occurs when two flights pass with less than 5 nmi separating them in the horizontal plane. The probability that one aircraft is predicted to pass within the region of LoS for time t ahead is $p(t, d=2.5 \text{ nmi})$. As a result, $p_{ACD}(t)$ and $p_{TCD}(t)$, which are the probability that AR and TSAFE detect (respectively) the impending LoS of the two aircraft in the simulation t seconds prior to NMAC, are just $p^2(t, d=2.5 \text{ nmi})$. Figure A11 shows $p(t, d=2.5$ *nmi)* and $p^2(t, d=2.5 \text{ nmi})$ from t= 60 sec to 12 min plotted as open dots and solid dots respectively, and the 4th polynomial fitting for $p^2(t, d=2.5 nmi)$ with a norm of residuals equaling 0.077271 is shown as well. As expected, the instantaneous probability that AR and TSAFE detect the LoS increases as look ahead time decreases, and is close to 1 when t = 60 sec, at which time the LoS is said to occur and TCAS attempts to over-ride TSAFE. The continuous fitting function, which is illustrated in Figure A7, is the following:

$P(t) = (-6.2 \cdot 10^{-12})t^4 + (1.2 \cdot 10^{-8})t^3 + (-6.3 \cdot 10^{-6})t^2 + (-3.0 \cdot 10^{-4})t + 1.0$, where t is measured in seconds
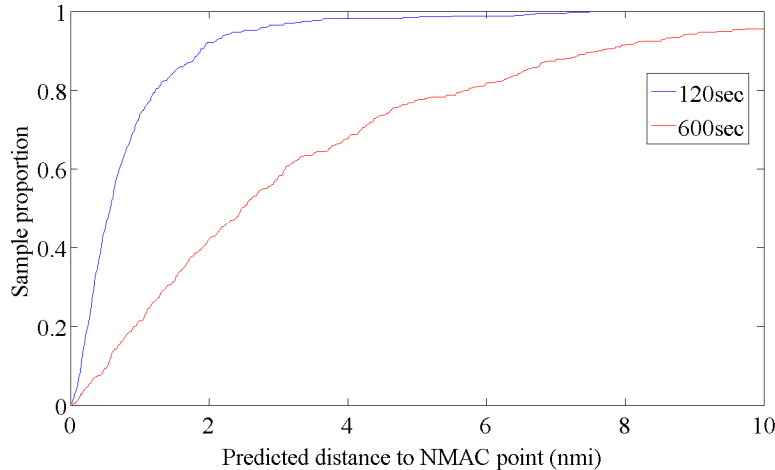

Figure A6: Proportion of aircraft whose predicted locations at the time NMAC will occur are distance d (nmi) from their NMAC point, plotted against distance, for 12 sec, 120 sec, and 600 sec ahead of NMAC

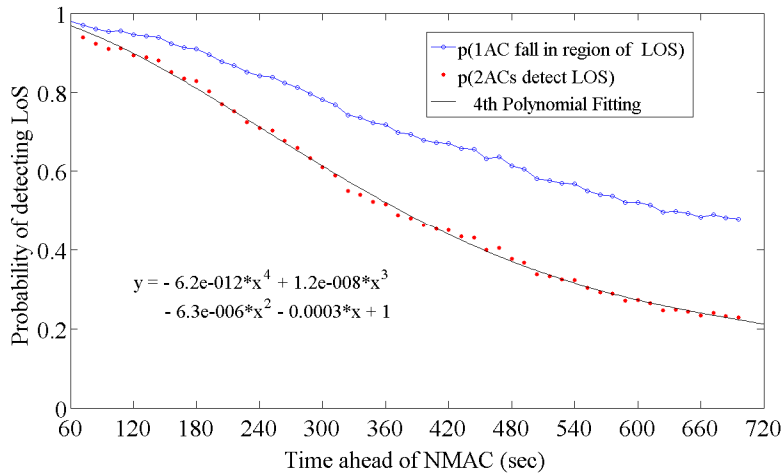American Institute of Aeronautics and Astronautics

Figure A7: Probability that one aircraft is predicted to pass into the region of LoS (blue open dots) and probability of detection of LoS for 2 flights (red solid dots) as a function of time ahead of NMAC.

This trajectory analysis was conducted in two dimensions, using along-track and cross-track error only, and treating the LoS region as a circle. The method may be extended to three dimensions, but the result would be sensitive to probability that two flights, on course to experience a NMAC absent any resolution, will have a level, ascending, or descending flight profile at the time NMAC is set to occur. Calculating this probability is not straightforward, and would necessitate finding the probability that two tracks enter a cylindrical volume as a function of time til NMAC. Alternatively, there is potential to analyze trajectories in three through further simulation using CTAS or ACES.

Acknowledgments

References

[1]Joint Planning and Development Office, "Next Generation Air Transportation System Integrated Plan," Dec. 12, 2004.

[2]Wing, D.J., Ballin, M.G., Krishnamurthy, K., "Pilot in Command: A Feasibility Assessment of Autonomous Flight Management Operations," 24th International Congress of the Aeronautical Sciences, Aug. 29 – Sep. 3, 2004.

[3]Battiste, V., Johnson, W.W., Holland, S., "Enabling Strategic Flight Deck Route Re-Planning Within A Modified ATC Environment: The Display of 4-D Intent Information on a CSD," Proceedings of the 2000 SAE World Aviation Congress and Exposition, Oct. 10-12, 2000.

[4]Erzberger, H., "The Automated Airspace Concept," 4th USA/Europe Air Traffic Management R&D Seminar, Dec. 3-7, 2001.

[5]Erzberger, H., Paielli, R.A., "Concept for Next Generation Air Traffic Control System," Air Traffic Control Quarterly, Vol. 10(4), pp. 355-378, 2002.

[6]Erzberger, H., "Transforming the NAS: The Next Generation Air Traffic Control System," 24th International Congress of the Aeronautical Sciences, Aug. 29 – Sep. 3, 2004.

[7]Bureau of Transportation Statistics, "Table 2-15: Number of Pilot-Reported Near Midair Collisions (NMAC) by Degree of Hazard," http://www.bts.gov/publications/national_transportation_statistics/html/table_02_15.html.

[8]Scarborough, A., Pounds, J., "Retrospective Human Factors Analysis of ATC Operational Errors," Federal Aviation Administration Civil Aerospace Medical Institute.

[9]Hemm, R.V., Busick, A.C., "Safety Analysis of the Separation Assurance Function in Today's National Airspace System," Report NS801T2, Jan. 2009.

[10]Blom, H.A.P., Obbink, B.K., Bakker, G.J.B., "Simulated Collision Risk of an Uncoordinated Airborne Self Separation Concept of Operation," 7th EUROCONTROL Innovative ATM Research Workshop and Exhibition, Bretigny, France, Dec. 2-4, 2008, pp. 33-42.

[11]Andrews, J.W., Welch, J.D., Erzberger, H., "Safety Analysis for Advanced Separation Concepts," 6th USA/Europe Air Traffic Management R&D Seminar, Jun. 27-30, 2005.

[12]Erzberger, H., "Automated Conflict Resolution for Air Traffic Control," 25th International Congress of the Aeronautical Sciences, Sep. 3-8, 2006.

[13]Farley, T., Erzberger, H., "Fast-Time Simulation Evaluation of a Conflict Resolution Algorithm under High Air Traffic Demand," 7th USA/Europe ATM 2007 R&D Seminar, Jul. 2-5, 2007.

[14]Farley, T., Kupfer, M., Erzberger, H., "Automated Conflict Resolution: A Simulation Evaluation Under High Demand Including Merging Arrivals," AIAA Aviation Technology Integration and Operations Conference, Sep. 18-20, 2007.

[15]Kupfer, M., Farley, T., Chu, Y., Erzberger, H. (2008), "Automated Conflict Resolution – A Simulation-Based Sensitivity Study of Airspace and Demand," 26th International Congress of the Aeronautical Sciences, Sep. 14-19, 2008.

[16]Paielli, R.A., Erzberger, H., Chiu, D., Heere, K.R., "Tactical Conflict Alerting Aid for Air Traffic Controllers," AIAA Journal of Guidance, Control and Dynamics, Vol.32(1), Jan.-Feb. 2009.

[17]Paielli, R.A., "Tactical Conflict Resolution using Vertical Maneuvers in En route Airspace," AIAA Journal of Aircraft, Vol.45(6), Nov-Dec 2008.

[18]Erzberger, H., Heere, K., "Algorithm and Operational Concept for Resolving Short-Range Conflicts," 26th International Congress of the Aeronautical Sciences, Sep. 14-19, 2008.

[19]U.S. Department of Transportation and Federal Aviation Administration, "Introduction to TCAS II: Version 7," Nov. 2000.

[20]Hammer, J., Caligaris, G., Llobet, M., "Safety Analysis Methodology for ADS-B Based Surveillance Applications," 7th USA/Europe ATM 2007 R&D Seminar, Jul. 2-5, 2007.

[21]Erzberger, H., "Automated Conflict Resolution for Air Traffic Control," 25th International Congress of the Aeronautical Sciences, Sep. 3-8, 2006.

[22]Erzberger, H., Paielli, R.A., Isaacson, D.R., and Eshow, M.M., "Conflict Detection and Resolution In the Presence of Prediction Error," 1st USA/Europe Air Traffic Management R&D Seminar, Jun. 17-20, 1997.

[23]Eriksson, M., Lundmark, J., "Technical Verification and Validation of ADS-B/VDL Mode 4 for A-SMGCS," SCAA_NUP_WP33_TVV_ADS_B_A-SMGCS_1.0, Dec. 2002.

[24]Thomas,R.,"FAA ATC Communications Services – System Wide Information Management (SWIM)," Aeronautical Communications Panel WG1 / Networking Subgroup / Internet Communications Services, Aug 27-31, 2007.

[25]Federal Aviation Administration, "En Route Automation Modernization (ERAM) Fact Sheet", Feb. 19 2009, accessed at http://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=7714

[26]National Transportation Safety Board, "Aviation Accident Database", accessed Jul. 27 2010, accessed at http://www.ntsb.gov/aviation/aviation.htm