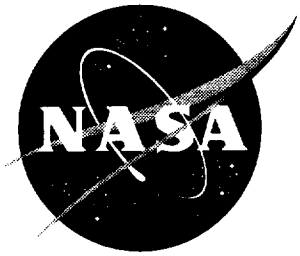
NASA/TM-2001-210864



On the Formal Verification of Conflict Detection Algorithms

César Muñoz
Institute for Computer Applications in Science and Engineering
Langley Research Center, Hampton, Virginia

Ricky W. Butler and Víctor A. Carreño
Langley Research Center, Hampton, Virginia

Gilles Dowek
Institute for Computer Applications in Science and Engineering
Langley Research Center, Hampton, Virginia

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

May 2001

Available from:

NASA Center for AeroSpace Information (CASI)
7121 Standard Drive
Hanover, MD 21076-1320
(301) 621-0390

National Technical Information Service (NTIS)
5285 Port Royal Road
Springfield, VA 22161-2171
(703) 605-6000

On the Formal Verification of Conflict Detection Algorithms*

César Muñoz† Ricky Butler‡ Víctor Carreño§ Gilles Dowek¶

May 14, 2001

Abstract

Safety assessment of new air traffic management systems is a main issue for civil aviation authorities. Standard techniques such as testing and simulation have serious limitations in new systems that are significantly more autonomous than the older ones. In this paper, we present an innovative approach, based on formal verification, for establishing the correctness of conflict detection systems. Fundamental to our approach is the concept of *trajectory*, which is a continuous path in the x-y plane constrained by physical laws and operational requirements. From the model of trajectories, we extract, and formally prove, high level properties that can serve as a framework to analyze conflict scenarios. We use the AILS alerting algorithm as a case study of our approach.

Key Words. trajectory modeling, conflict detection, collision alerting, formal methods, theorem proving

Subject. Computer Science

*This work was supported by the National Aeronautics and Space Administration under NASA Contract No. NAS1-97046 while the first and fourth author were in residence at ICASE, NASA Langley Research Center, Hampton, VA 23681-2199, USA.

†ICASE, Mail Stop 132C, NASA Langley Research Center, Hampton, VA 23681-2199, USA, e-mail: munoz@icase.edu.

‡Assessment Technology Branch, Mail Stop 130, NASA Langley Research Center, Hampton, VA 23681-2199, USA, e-mail: r.w.butler@larc.nasa.gov.

§Assessment Technology Branch, Mail Stop 130, NASA Langley Research Center, Hampton, VA 23681-2199, USA, e-mail: v.a.carreno@larc.nasa.gov.

¶INRIA, Domaine de Voluceau - Rocquencourt - B.P. 105, 78153 Le Chesnay Cedex, France, e-mail: gilles.dowek@inria.fr.

1 Introduction

In the current aerospace system, commercial flights are controlled by Air Traffic Control (ATC) from gate-to-gate. Before a flight can take place, the complete route plan must be sent to the ATC authorities in charge of the geographical sectors crossed by the aircraft. During the flight, even minor changes to the plan require a clearance from ATC before they can be performed. New distributed air-ground traffic management concepts [1] are being developed to address the inefficiencies of the current system. For example, the *free-flight* concept allows direct flight routes without ATC intervention[16], and the *Airborne Information for Lateral Spacing (AILS)* concept allows simultaneous and independent landing on closely spaced runways [17].

A key aspect of these new concepts is that they shift responsibility for aircraft separation from air traffic controllers to pilots and automation. This change is theoretically possible because recent technology such as D-GPS (Differential Global Position System) and ADS-B (Automatic Dependent Surveillance Broadcast) can provide very accurate data-flight information to pilots and on board computers. Computer systems can warn pilots when other aircraft are dangerously intruding into their own airspace. Despite the technology advances, a major concern of civil aviation authorities is that this approach may compromise the overall system safety.

In this paper, we address the issue of safety assessment for conflict detection systems. In the avionics community, testing and simulations are the standard methods for certifying safety of digital systems. The AILS project, for instance, has conducted extensive simulation and testing of the alerting algorithm. So far, no major flaws in its logic have been detected. However, neither testing nor simulation can give a definitive answer to questions such as: *"What is the lookahead time for an alarm prior to a conflict?"* or *"Does there exist a trajectory leading to an undetected conflict?"* These questions can only be solved by using mathematical analysis. Given the nature of the problem[1], we also believe that such analysis should be mechanically checked via a theorem proving system, such as PVS, or other automated proving techniques, e.g., model-checking.

In general, avionics systems, such as AILS, are hybrid systems. That is, they consist of simultaneous discrete and continuous behavior. The continuous behavior arises from the kinematics of the aircraft. The discrete behavior is an inherent aspect of any embedded digital system. Several approaches have been used in the literature to model hybrid systems (see [18]). Most of these approaches use extensions of finite state automata theory to handle state variables ranging over real numbers. Properties are then formalized as a reachability problem and proven by using model-checking and theorem proving techniques. These techniques have been shown to be effective for handling systems where control logic modes

[1] As the appendix reveals, the analysis required to establish the safety properties involves a complex mixture of long deductions, algebraic manipulations, calculations of formulas with specific values, and inequality reasoning. Performing this analysis by hand is tedious and error prone. For example, some of the proofs required case splits that on the surface looked symmetrical, but were later found to be slightly different during mechanical checking.

trigger continuous and dynamic changes of the state. For instance, the TCAS alerting system for preventing midair collision was modeled using a hybrid automata approach [14]. In other collision alerting systems, such as AILS, the discrete aspects do not arise from control variables but from the discretization of temporal and spatial domains. For instance, the AILS algorithm checks every half a second whether future aircraft locations (calculated by projection of the current locations) violates a distance threshold in a given lookahead time.

In our approach, instead of relying on state automata models, we construct a continuous model of aircraft trajectories, where we can prove properties using standard calculus and mathematics. Then, we verify the correctness of the algorithm with respect to this continuous model. In this paper, and for readability, we have used standard mathematics and traditional logic reasoning. Nevertheless, our development has been formally checked in the general verification system PVS [15]. All the theories and proofs are available through the URL `http://shemesh.larc.nasa.gov/fm/ftp/ails/`.

The remainder of this paper is organized as follows. In Section 2, we develop a mathematical framework for the analysis of conflict scenarios. This framework is formalized and verified in Section 3. In Section 4, we use our mathematical model to study the correctness of the AILS alerting algorithm. The last section summarizes our work and contains concluding remarks. As an appendices, we include the technical lemmas referenced in the paper, a table of translations for the conventions used in the paper into the equivalent PVS language, and the AILS alerting algorithm in PVS.

2 Conflict Avoidance Framework

Conflict detection algorithms are designed to predict *conflict* situations between the own aircraft and another aircraft within some lookahead time T in the future, i.e., $T > 0$. In our framework, two aircraft have a *(potential) conflict* at time T, if there exists a trajectory leading to a distance between the aircraft less than a given value `ConflictRange` at time T. The value of `ConflictRange` largely depends on the concept that is being implemented. For a landing concept such as AILS, the `ConflictRange` is in the order of feet, but for a general mid-air conflict detection algorithm it could be in the order of nautical miles.

Predictions of aircraft trajectories are made to determine if a conflict exists in a given lookahead time. Two types of information can be used for prediction: (1) intent information for medium to long lookahead times; and (2) state information for short to medium lookahead times. Intent information refers to information in flight plans, destination, in route way points, etc. State information uses the airplane heading, speed and location to predict future aircraft states. In this paper, we are only concerned with trajectory prediction based on state information.

Assuming that aircraft have reliable access to accurate data flight information, two key properties that must be established for a conflict detection algorithm are (1) any predicted conflict within time T issues an alarm at time 0, and (2) an alarm at time 0 reflects a potential conflict at time T. The first property is called *correctness* and the latter one is *certainty*. Notice that certainty means that the alerting system does not issue *false alarms*.

Since possible conflict that are not alerted may lead to future collisions, correctness is a much more critical feature, from a safety point of view, than certainty. However, false alarms will have a negative effect in the overall performance of the airspace system [12].

Given the hybrid nature of the conflict detection systems, formal verification of correctness and certainty is a complex task, highly dependent on the particular subtleties of each algorithm. In this section, we develop a general framework to study that kind of systems. It consists of (1) a nominal model of trajectories, (2) intruder and evader aircraft trajectory assumptions, (3) convergence and divergence trajectory criteria, and (4) a set of general conditions for conflict avoidance. This framework, which is formalized and verified in Section 3, is used in Section 4 for studying the correctness and certainty properties of the AILS alerting algorithm.

2.1 Aircraft Trajectories

At the basis of our verification approach is the concept of *aircraft trajectory*. In [13], Kuchar and Yang present a survey on conflict detection and resolution modeling methods. In that survey, three kinds of trajectory models are characterized: *nominal*, *worst-case*, and *probabilistic*. In the nominal approach, the future aircraft state, i.e., position, speed, heading, bank angle, is projected from the current state according to physics laws. In the worst-case approach, the future state is projected by following a policy of extreme values for specific state variables. In a probabilistic model, uncertainties such as weather conditions or extrapolation errors are taken into account to calculate the most probable aircraft trajectories. For the case of parallel landing, an algorithm based on a probabilistic model was proposed in [6].

In general, nominal models are more conservative than probabilistic and worst-case ones. However, they also generate a greater number of false alarms. In contrast, probabilistic models produce a lower number of false alarms [11, 6], but they may disregard some rare conflicting situations. To formally answer a question such as *"Does there exist a conflict without an alarm being issued?"*, we need a model that includes the set of *all possible* trajectories from given aircraft initial states. This is precisely the information provided by our nominal model.

In our model, a *trajectory* is defined to be a continuous path in the x-y plane subject to constraints imposed by the aircraft dynamics.[2] Formally, the kinematics of an aircraft moving at constant ground speed v in a two-dimensional plane is given by the equations

$$x'(t) = v\cos(\theta(t)) \quad (1)$$

$$y'(t) = v\sin(\theta(t)) \quad (2)$$

$$\theta'(t) = (g/v)\tan(\phi(t)) \quad (3)$$

where x, y, θ, ϕ are differentiable functions mapping time to location coordinates, heading, and bank angle, respectively. Equations 1 and 2 state that the derivative of the position functions gives the velocity vector of the aircraft. Equation 3 relates the bank angle with

[2]The vertical separation is typically handled separately. This will be studied in future work.

the heading of the aircraft. That equation states that the rate of direction change of an aircraft is proportional to the tangent of the bank angle by a factor of g/v, where g is the gravitational force. We assume a minimal ground speed of 210 feet per second.

In addition to the above physical constraints, we impose a maximum bank angle operational constraint for commercial aircraft to be 35^o, i.e.,

$$|\phi(t)| \leq 35\pi/180. \tag{4}$$

Henceforth, we use the constant `MaxBank` $= 35\pi/180$.

From the equations defining the motion of the aircraft, we can deduce minimum and maximum distances traveled by an aircraft in a given time. In particular, vt is the farthest distance, i.e., via straight line, that can be reached by an aircraft moving at constant speed v in t seconds. That property is called `YCNGFTYS`, which stands for *You Cannot Go Faster Than Your Speed*, and can be stated as follows

Theorem 1 (`YCNGFTYS`).

$$0 \leq t \supset \sqrt{(x(t) - x(0))^2 + (y(t) - y(0))^2} \leq vt.$$

The above theorem has been formally proven in PVS. The proof, however, is much more complex than the intuition behind it, which is illustrated in Figure 1.

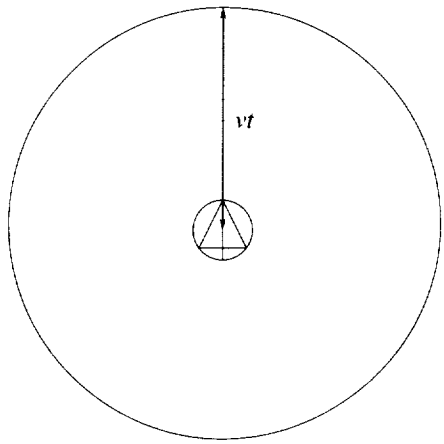


Figure 1: You Cannot Go Faster Than Your Speed

According to Figure 2, for an aircraft moving at constant speed v and with a constant bank angle ϕ, the distance from the position at time 0 to the position at time t is given by the formula

$$m(v, \phi, t) \quad = \quad 2r(v, \phi) \sin(vt/2r(v, \phi)) \tag{5}$$

where $r(v, \phi)$ is the turn radius of the aircraft.

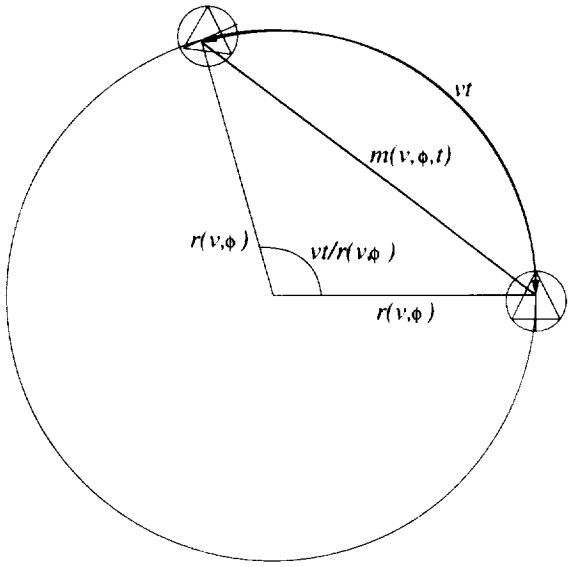


Figure 2: Distance Traveled in Curved Trajectory

The turn radius $r(v, \phi)$ can be calculated as follows.

$$\begin{aligned} vt/r(\phi, v) &= (g/v)\tan(\phi)t \quad \text{(From Equation 3)} \\ v/r(v, \phi) &= (g/v)\tan(\phi) \quad \text{(Simplifying } t\text{)}. \end{aligned}$$

Thus,

$$r(v, \phi) = v^2/(g\tan(\phi)). \tag{6}$$

According to Formula 4, the maximum change of heading per second of an aircraft moving at constant speed v is given by

$$\rho(v) = (g/v)\tan(\texttt{MaxBank}). \tag{7}$$

From Equation 6 and Equation 7:

$$r(v, \texttt{MaxBank}) = v/\rho(v) \tag{8}$$

and from Equation 5 and Equation 8:

$$m(v, \texttt{MaxBank}, t) = 2r(v, \texttt{MaxBank})\sin(\rho(v)t/2). \tag{9}$$

When $0 \leq \rho(v)t \leq 2$, we have formally proven in PVS that $m(v, \texttt{MaxBank}, t)$ is the minimum distance traveled by an aircraft moving at constant speed v in t seconds[3]. The property is called `YCNGSTYS`, which stands for *You Cannot Go Slower Than Your Speed*

[3]We conjecture that the property still holds for $0 \leq \rho(v)t \leq 2\pi$; but, we could not find a formal proof of this proposition.

Theorem 2 (YCNGSTYS).

$$0 \leq \rho(v)t \leq 2 \supset m(v, \texttt{MaxBank}, t) \leq \sqrt{(x(t) - x(0))^2 + (y(t) - y(0))^2}.$$

According to theorems YCNGFTYS (Theorem 1) and YCNGSTYS (Theorem 2), for an aircraft moving at constant ground speed v, the inner circle of radius $m(v, \texttt{MaxBank}, t)$ and the outer circle of radius vt, both centered at the current position of the aircraft, delimit the area that could be reached by the aircraft at time t. See Figure 3.

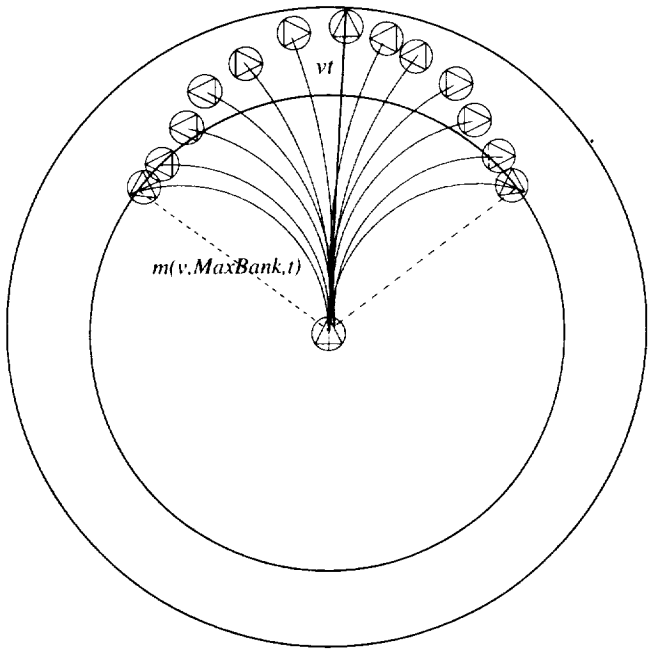


Figure 3: Reachable Area of an Aircraft at Time t

2.2 Intruder and Evader Aircraft

We consider an airspace sector with only two aircraft. We also assume that one of the aircraft, called *evader*, follows an straight path at its current heading, i.e., the bank angle of the evader is considered to be always 0. The other aircraft is called *intruder* and no particular assumptions are made for its trajectory. Without lost of generality, we take a coordinate system where the x-axis coincides with the evader trajectory. In that case, the heading angle of the evader aircraft is always 0.

Multiple-aircraft scenarios can be modeled as sequential composition of pair-wised aircraft conflict detection algorithms. Notice however, that when solving conflicts in a multiple-aircraft system, solutions to a pair of aircraft could create new conflicts in previously solved aircraft. Although, *conflict resolution algorithms* are out of the scope of this report, we would like to mention at least three kinds of techniques for conflict resolution that are relevant to

our model of trajectories (see for example, [2, 4, 5, 10]): *geometric optimization*, *modified potential-field*, and *predefined escape maneuver*. The first one tries to minimize the velocity vector change required for each conflict for solving the conflict. The modified potential-field approach exploits an analogy between air traffic and repulsion-attraction features of charged particles in a potential field. Predefined escape maneuvers are used in specific flight situations where stronger assumptions on aircraft trajectories can be made. Landing is one of these situations. Indeed, the AILS concept uses a predefined escape maneuver which instructs the pilot of the evader aircraft to climb and turn away 45^o from the intruder aircraft when a traffic warning alarm issued.

Henceforth, state functions representing the state of the evader aircraft are subscripted with a lowercase e; similarly, intruder state functions are subscripted with a lowercase i. We use θ_t as an abbreviation for $\theta_i(t)$. Applying the above restrictions on the evader trajectory to equations 1, 2, and 3, we get

$$x_e(t) = X_e + v_e t \tag{10}$$
$$y_e(t) = Y_e \tag{11}$$
$$\theta_e(t) = 0 \tag{12}$$
$$\phi_e(t) = 0 \tag{13}$$

where X_e and Y_e are the coordinates of the initial evader position.

The evader represents an aircraft flying on normal conditions while the intruder represents a blundering aircraft. Constraints on the evader trajectory are justifiable since even under free-flight rules, aircraft normally fly on straight lines. In the particular case of the AILS concept, and for safety reasons, the alerting algorithm runs twice on each airplane. In the first execution, the algorithm treats the local aircraft as the evader an the foreign aircraft as the intruder. In the second execution, the roles of intruder and evader aircraft are interchanged.

2.3 Convergence and Divergence of Trajectories

Fundamental to a conflict detection algorithm is the ability to determine whether the trajectories of two aircraft are diverging or converging and to find the point of closest separation of the projected trajectories. This amounts to finding the minimum of the distance between two straight lines. If the evader aircraft is assumed to have heading 0 and the intruder aircraft has heading θ, then the equations defining the *projected trajectories* are

$$\begin{aligned} x_e^*(t) &= x_e(0) + v_e t \\ y_e^*(t) &= y_e(0) \\ x_i^*(t) &= x_i(0) + v_i t \cos(\theta) \\ y_i^*(t) &= y_i(0) + v_i t \sin(\theta) \end{aligned}$$

and the distance between the *projected trajectories* at time t, $R(t)$, can be computed as follows:

$$\begin{aligned} \Delta_x(t) &= x_i^*(t) - x_e^*(t) \\ \Delta_y(t) &= y_i^*(t) - y_e^*(t) \end{aligned}$$

$$R(t) = \sqrt{\Delta_x(t)^2 + \Delta_y(t)^2} \tag{14}$$

To find the minimum of $R(t)$, first the derivative of $R(t)$ is computed:

$$R'(t) = \frac{\Delta_x(t)\Delta_x' + \Delta_y(t)\Delta_y'}{R(t)}$$

where

$$\begin{aligned} \Delta_x' &= v_i \cos(\theta) - v_e \\ \Delta_y' &= v_i \sin(\theta) \end{aligned}$$

When $R'(t+\tau) = 0$, we get the time τ, relative to t, of the point of closest separation between the aircraft. The solution to this equation is:

$$\tau(t) = -\frac{\Delta_x(t)\Delta_x' + \Delta_y(t)\Delta_y'}{\Delta_x'^2 + \Delta_y'^2} \tag{15}$$

These equations were formally derived using the computer algebra tool MuPAD [9]. It is important to note that τ is undefined, i.e., denominator is zero, when the aircraft are parallel and the ground speeds are equal.

For any time t, if $\tau(t)$ is negative or zero, the tracks are diverging or parallel, respectively. If $\tau(t)$ is greater than zero, the tracks are converging and $\tau(t)$ is the time of closest separation relative to t. See Figure 4.

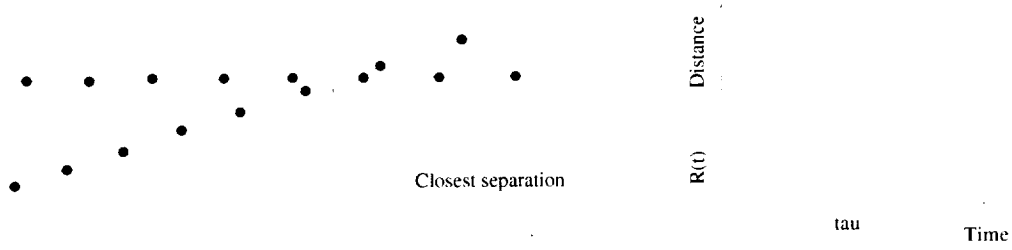


Figure 4: Converging tracks

We have formally proven that τ satisfies the following properties.

Lemma 1 (derivative_eq_zero_min).

$$R(t_1 + \tau(t_1)) \leq R(t_1 + t_2).$$

Lemma 2 (asymptotic_decrease_tau).

$$t_1 \leq t_2 \leq \tau(t) \supset R(t + t_1) \geq R(t + t_2).$$

Lemma 3 (asymptotic_increase_tau).

$$\tau(t) \leq t_1 \leq t_2 \supset R(t + t_1) \leq R(t + t_2).$$

2.4 General Conditions for Conflict Avoidance

In this section, we present a set of sufficient conditions for conflict avoidance between intruder and evader aircraft. The basic geometry of our approach is illustrated in Figure 5. The initial position of the intruder is $(x_i(0), y_i(0))$ and the position of the evader at time T is $(x_e(T), y_e(T))$. We name the angle between the path of the evader and the line passing by these two points as β. The distance from $(x_i(0), y_i(0))$ to $(x_e(T), y_e(T))$ is named l. Label d denotes the distance between the initial evader position and the initial intruder position. Given a time t, $0 \leq t \leq T$, the expressions $e(t)$ and $i(t)$ denote the distance between the intruder at time t and the evader at time T, and the distance between the intruder at time 0 and the intruder at time t, respectively. We also use r,ρ, and $m(t)$ as abbreviations for $r(v_i, \texttt{MaxBank})$, $\rho(v_i)$, and $m(v_i, \texttt{MaxBank}, t)$, respectively.

Formally, the distance from the position of the intruder aircraft at time t_i to the position of evader aircraft at time t_e, denoted $D_{ie}(t_i, t_e)$, is defined as follows

$$D_{ie}(t_i, t_e) = \sqrt{(x_i(t_i) - x_e(t_e))^2 + (y_i(t_i) - y_e(t_e))^2}.$$

Therefore,

$$\begin{aligned} l &= D_{ie}(0, T) \\ d &= D_{ie}(0, 0) \\ e(t) &= D_{ie}(t, T) \\ i(t) &= \sqrt{(x_i(0) - x_i(t))^2 + (y_i(0) - y_i(t))^2} \\ \theta_0 &= \theta_i(0) \end{aligned}$$

and β is an angle such that

$$x_e(T) = l\cos(\beta) + x_i(0) \tag{16}$$

$$y_e(T) = y_i(0) - l\sin(\beta) \tag{17}$$

Formally, we say that two aircraft *are in a conflict* at time t, when the following predicate holds

$$\texttt{conflict}_{ie}(t) \equiv D_{ie}(t, t) \leq \texttt{ConflictRange}. \tag{18}$$

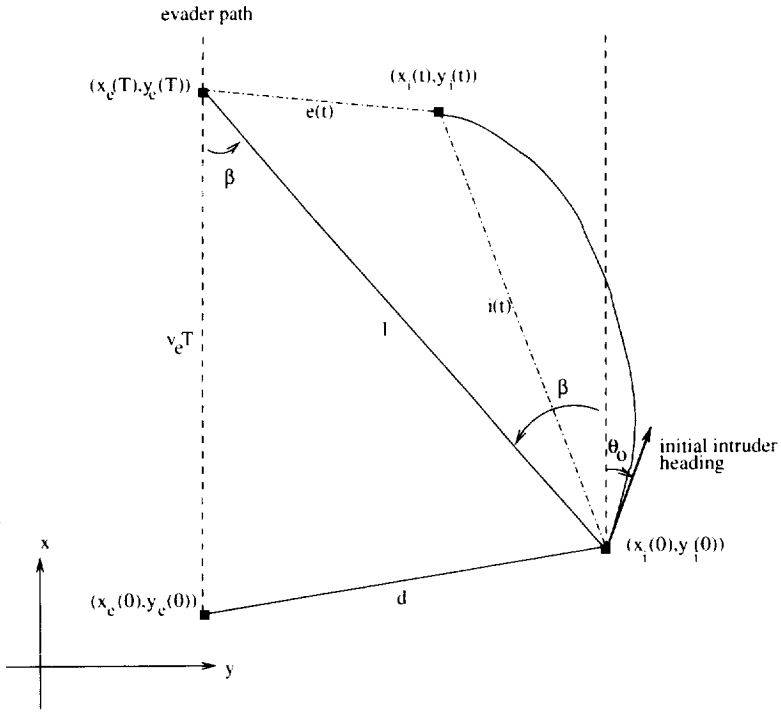


Figure 5: Basic Geometry

Note that $\neg\texttt{conflict}_{ie}(T)$ is equivalent to $e(T) > \texttt{ConflictRange}$. Furthermore, $\texttt{conflict}_{ie}(T)$ does not include conflicts at time less than T. However, since we assume a continuous time, a conflict at time $t < T$ can be analyzed by taken a new reference time system, where time 0 is $t - T$. In the new time of reference, the conflict happens at time T.

We now state a set of sufficient conditions to avoid conflict scenarios between intruder and evader aircraft. All these conditions are suggested by the geometry of the problem. Conditions (1) and (2) are consequences of the reachable area of an aircraft explained on Section 2.1. Condition (3) states that given some initial conditions, if intruder and evader aircraft are heading to opposite directions, then a conflict scenario is impossible. Last condition, characterizes non conflict scenarios when both aircraft are heading to the same direction.

Given $T \geq 0$, $\forall i, e.\ \neg\texttt{conflict}_{ie}(T)$, when

1. **no_conflict_gt_max**:

$$l > \texttt{MaxDistance},$$

   where $\texttt{MaxDistance} = v_i T + \texttt{ConflictRange}$, or

2. **no_conflict_lt_min**:

$$l < \texttt{MinDistance} \ \wedge \ 0 \leq \rho T \leq 2\pi,$$

   where $\texttt{MinDistance} = m(T) - \texttt{ConflictRange}$, or

3. **no_conflict_Omega**:

$$l > \texttt{ConflictRange} + v_i \ \wedge \ \rho T \leq \pi - \rho \ \wedge \ \texttt{Omega}(\beta + \theta_0),$$

where $\texttt{Omega}(\sigma) \ \equiv \ \pi/2 \leq \sigma \leq 3\pi/2$.

If $v_i = v_e = 250$ feet/s, and $\texttt{AlertRange}$=1400 feet, as it is in the AILS concept, we also have $\neg\texttt{conflict}_{ie}(T)$ when

4. **ails_no_conflict_tau_le0**:

$$\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \ 9.5 \leq T \leq 10 \ \wedge$$
$$\neg\texttt{Omega}(\beta + \theta_0) \ \wedge \ d > \texttt{AlertRange} \ \wedge \ \tau(0) \leq 0.$$

Next section is devoted to the formal verification of the conflict avoidance properties.

3 Formal Verification of the Conflict Avoidance Framework

The specification language of the PVS system is based on a higher-order logic extended with a very rich type theory. This language gives us all the necessary power to express our model of nominal trajectories in a simple way. For instance, trajectories are defined via the PVS sub-typing mechanism in such a manner that the equations characterizing the motion of an aircraft are just type-correctness conditions. Most of these conditions are discharged automatically by the PVS type checker.

PVS includes several decision procedures to cope with well-known decidable theories. However, like most theorem provers, it has little automated support for non-linear arithmetic and real analysis. We have extended the pre-defined theory of real numbers and the theory of differential functions developed in [3] with theories dealing with trigonometric and other transcendental functions.

Non effective real functions are declared in PVS as uninterpreted constants of a given type. Their behavior is then constrained via axioms. For example, cos and sin are functions from reals to the real interval $[-1 \ldots 1]$ satisfying, among other properties, $\sin(a)^2 + \cos(b)^2 = 1$. In a similar way, $\sqrt{}$ is a function from non negative reals to non negative reals such that $\sqrt{a}^2 = a$ for $a \geq 0$. From this axiom, we can prove, for instance, that $\sqrt{a^2} = a$ for $a \geq 0$.

The remainder of this section is devoted to the proof of the General Conditions for Conflict Avoidance presented in Section 2. First, we discuss some technical details on the formal proofs of inequalities and on integrating geometrical reasoning in a theorem prover such as PVS. Then, we introduce a new system of coordinates. Finally, we detail the proofs of the conditions in Section 2.4.

3.1 Dealing With Inequalities

Most of the properties that must be proven involve inequalities. First notice that as a consequence of the axiomatic definition of $\sqrt{.}$, a property like $0 \leq a \leq \sqrt{b}$, must be proven by first establishing $a^2 \leq b$ and then using a property of monotony over the squared function. To deal with general inequalities, we assume the following calculus theorem

Theorem 3 (`monotonic_anti_deriv`).

$$\begin{array}{l} \forall f, g : \mathbb{R} \to \mathbb{R}.\ \forall a, b : \mathbb{R}.\ a \leq b\ \supset \\ \quad (\forall c : \mathbb{R}.\ a \leq c \leq b\ \supset\ f'(c) \leq g'(c)) \\ \quad \supset \\ \quad f(b) - f(a) \leq g(b) - g(a). \end{array}$$

In the verification process is sometimes inevitable to perform calculations on expressions containing non effective functions such as the trigonometric functions. It is tempting to use approximation series to define, for instance, sin and cos. However, mixing approximation series and axiomatic definitions of trigonometric functions may be source of paradoxes. Say for example that sin and cos compute approximate values of the real ones. It will be very unlike that $\sin(a)^2 + \cos(a)^2$ evaluates to 1 for any value of a. In order to avoid that kind of inconsistencies, we mix approximations and uninterpreted functions in a very rigorous way. Assume we want to prove that $e_1[\sin(a)]^+ \leq e_2[\cos(b)]^+$, i.e., e_1 contains a distinguished positive occurrence of $\sin(a)$ and e_2 contains a distinguished positive occurrence of $\cos(b)$. Then, we find a computable upper bound of $\sin(a)$, say $\sin_{ub}(a)$, and a computable lower bound of $\cos(b)$, say $\cos_{lb}(b)$. Finally, we prove $e_1[\sin(a)]^+ \leq e_2[\cos(b)]^+$ as follows

$$e_1[\sin(a)]^+ \leq e_1[\sin_{ub}(a)]^+ \tag{19}$$
$$e_1[\sin_{ub}(a)]^+ \leq e_2[\cos_{lb}(b)]^+ \tag{20}$$
$$e_2[\cos_{lb}(b)]^+ \leq e_2[\cos(b)]^+ \tag{21}$$

Most of the times, Formulas 19 and 21 are simple to discharge. If $e_1[\sin(a)_{ub}]^+$ and $e_2[\cos(b)_{lb}]^+$ are computable then we prove Formula 20 by evaluating the expressions. Otherwise, we use the same technique to remove other non computable values. Eventually, we will get two expressions that we can evaluate. This technique is so used and simple that we have developed PVS strategies to automate the work.

In particular, we use the following definitions

$$\sin_{lb}(a) = \sum_{i=1}^{4}(-1)^{i-1}\frac{a^{2i-1}}{(2i-1)!} \qquad \sin_{ub}(a) = \sum_{i=1}^{5}(-1)^{i-1}\frac{a^{2i-1}}{(2i-1)!}$$

$$\cos_{lb}(a) = 1 + \sum_{i=1}^{3}(-1)^{i}\frac{a^{2i}}{(2i)!} \qquad \cos_{ub}(a) = 1 + \sum_{i=1}^{4}(-1)^{i}\frac{a^{2i}}{(2i)!}$$

and the axioms[4]

[4]In PVS, real numbers are written as rational numbers.

Axiom 1 (PI).

$$314/100 \leq \pi \leq 315/100.$$

Axiom 2 (SIN).

$$0 \leq a \leq \pi \ \supset \ \sin_{lb}(a) \leq \sin(a) \leq \sin_{ub}(a).$$

Axiom 3 (COS).

$$-\pi/2 \leq a \leq \pi/2 \ \supset \ \cos_{lb}(a) \leq \cos(a) \leq \cos_{ub}(a).$$

3.2 New System of Coordinates

The first major step in our formal development is to take as reference a new system of coordinates where the origin is the position of the evader aircraft at time T, i.e., $(x_e(T), y_e(T))$, and the x-y plane has been rotate by θ_0 degrees. The new $\hat{x}$-$\hat{y}$ plane, which is illustrated in Figure 6, is defined as follows

$$\hat{x}(t) = \cos(\theta_0)[x(t) - x_e(T)] + \sin(\theta_0)[y(t) - y_e(T)] \quad (22)$$

$$\hat{y}(t) = \cos(\theta_0)[y(t) - y_e(T)] - \sin(\theta_0)[x(t) - x_e(T)] \quad (23)$$

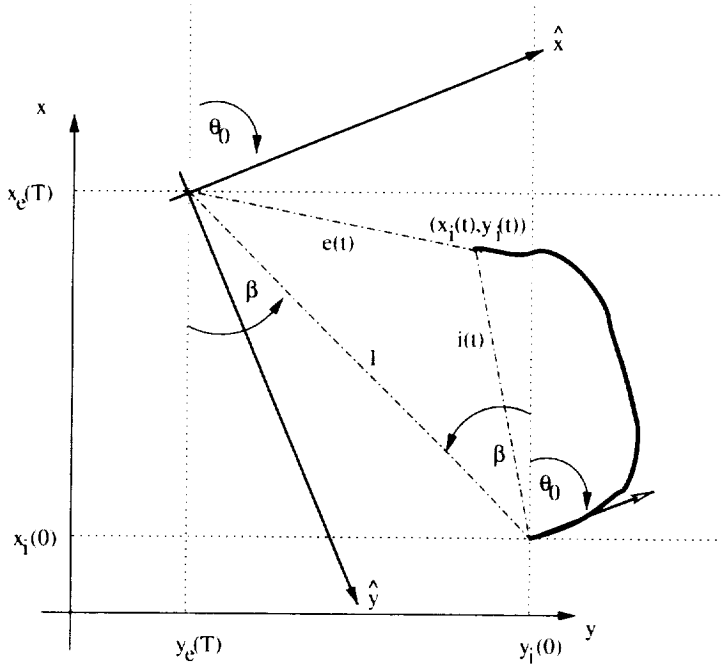


Figure 6: New Coordinate System

We have formally proven that distances are invariant under rotation and translation of the coordinate system. In particular, we have proven

Lemma 4 (`isometric`).

$$(x_1(t) - x_2(t))^2 + (y_1(t) - y_2(t))^2 = (\hat{x_1}(t) - \hat{x_2}(t))^2 + (\hat{y_1}(t) - \hat{y_2}(t))^2.$$

From lemma `isometric`, we can easily derive

Lemma 5 (`isometric_evader`).

$$e(t)^2 = \hat{x_i}(t)^2 + \hat{y_i}(t)^2.$$

Lemma 6 (`isometric_intruder`).

$$i(t)^2 = (\hat{x_i}(t) - \hat{x_i}(0))^2 + (\hat{y_i}(t) - \hat{y_i}(0))^2.$$

As a corollary of lemma `isometric_evader`, we have

Lemma 7 (`majoration`).

$$e(t)^2 \geq \hat{x_i}(t)^2 \ \wedge \ e(t)^2 \geq \hat{y_i}(t)^2.$$

3.3 Geometric Reasoning

Since conflict detection systems solve a physical problem, visualization plays an important role in the verification process. We have extensively used tools such as MuPAD [9] and GNUPLOT [19] to find geometrical relations between different components of the mathematical framework before attempting a direct proof in the theorem prover.

In this section, we show proofs of two geometrical properties: `Alpha_d_AlertRange` and `R_T`. Lemma `Alpha_d_AlertRange` exploits the Law of Cosines to bound the angle β and R_T provides a formula for computing $R(T)$ from l and the angles β and θ. We recall that $R(T)$ is the projected distance between the evader and the intruder assuming that the intruder continues in a straight line on his present course. Since geometrical reasoning is usually easier to illustrate than to formalize, we base our reasoning on Figure 7. However, the PVS proofs are filled with details concerning the coordinate geometry version of the diagram.

Lemma 8 (`Alpha_d_AlertRange`).

$$\texttt{Alpha}(\beta) \iff d \leq \texttt{AlertRange},$$

where $\texttt{Alpha}(\sigma) \equiv \cos(\beta) \geq ((vT)^2 + l^2 - \texttt{AlertRange}^2)/2vTl.$

Proof. From the Law of Cosines,

$$\cos(\beta) = ((vT)^2 + l^2 - d^2)/2vTl.$$

Therefore,

$$\begin{array}{rcll} \cos(\beta) & \geq & ((vT)^2 + l^2 - \texttt{AlertRange}^2)/2vTl & \iff \\ ((vT)^2 + l^2 - d^2)/2vTl & \geq & ((vT)^2 + l^2 - \texttt{AlertRange}^2)/2vTl & \iff \\ -d^2 & \geq & -\texttt{AlertRange}^2 & \iff \\ d^2 & \leq & \texttt{AlertRange}^2 & \iff \\ d & \leq & \texttt{AlertRange}. & \end{array}$$

□

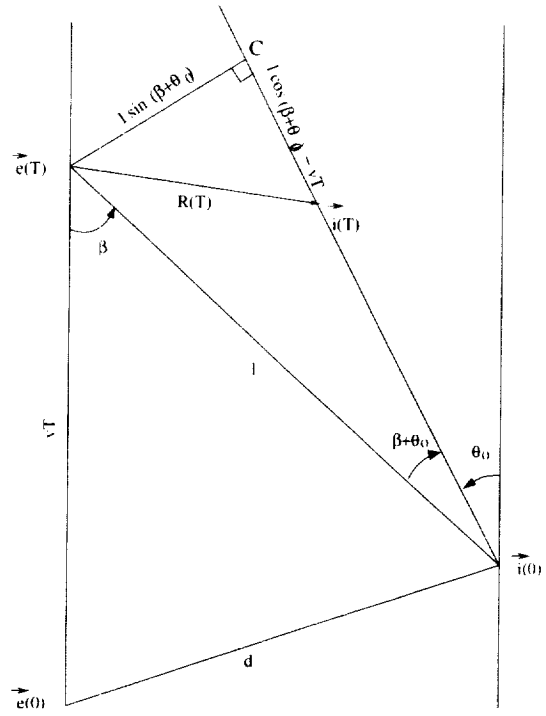


Figure 7: $R(T)$: Projected Distance at Time T

Lemma 9 (R_T).

$$R^2(T) = (l\cos(\beta+\theta_0) - (vT))^2 + (l\sin(\beta+\theta_0))^2.$$

Proof. At time T the evader will be at $\vec{e}(T) = (x_e(T), y_e(T))$ and the projected location of the intruder at time T is $\vec{i}(T) = (x_i(T), y_i(T))$. Dropping a perpendicular line from $\vec{e}(T)$ to the intruder line defines point C. The distance from $\vec{e}(T)$ to C is $l\sin(\beta+\theta_0)$. The distance from $\vec{i}(0) = (x_i(0), y_i(0))$ to C is $l\cos(\beta+\theta_0)$. The distance from $\vec{i}(0)$ to $\vec{i}(T)$ is vT since the intruder is travelling at a constant rate of v. Thus, the distance from point $\vec{i}(T)$ to C will be $l\cos(\beta+\theta_0) - vT$. By the Pythagorean theorem, we have:

$$R^2(T) = (l\cos(\beta+\theta_0) - (vT))^2 + (l\sin(\beta+\theta_0))^2.$$

□

Finally, we address the proof of the sufficient conditions presented in Section 2.4. Note that some proofs may refer to technical lemmas included in the Appendix A.

3.4 Theorem no_conflict_gt_max

We must establish that the distance between the intruder and the evader at time T is greater than `ConflictRange` for all possible trajectories of the intruder:

$$T \geq 0 \ \wedge \ l > \texttt{MaxDistance} \\ \supset \\ \neg\texttt{conflict}_{ie}(T).$$

Proof. To establish $\neg\texttt{conflict}_{ie}(T)$, it suffices to show $e(T) > \texttt{ConflictRange}$. From Lemma `YCNGFTYS_evader` (see Appendix A), and instantiating t with T, we have

$$T \geq 0 \ \wedge \ l \geq v_i T \ \supset \ l - v_i T \leq e(T).$$

The premise $l \geq v_i$ is discharged from hypothesis $l > \texttt{MaxDistance}$ and definition of `MaxDistance`. To conclude the proof, we show that $l - v_i T > \texttt{ConflictRange}$ as follows

$$\begin{array}{rcll} l - v_i T & > & \texttt{MaxDistance} - v_i T & \text{By hypothesis } l > \texttt{MaxDistance} \\ & > & \texttt{ConflictRange} + v_i T - v_i T & \text{By definition of MaxDistance} \\ & > & \texttt{ConflictRange} & \text{Simplifying.} \end{array}$$

□

3.5 Theorem no_conflict_le_min

This theorem states that all possible trajectories of the intruder stay outside of the conflict region, if the initial distance from the evader, l, is less than `MinDistance`:

$$T \geq 0 \ \wedge \ l < \texttt{MinDistance} \ \wedge \ 0 \leq \rho T \leq 2 \\ \supset \\ \neg\texttt{conflict}_{ie}(T).$$

Proof. To establish $\neg\texttt{conflict}_{ie}(T)$, we show that $\texttt{ConflictRange} < e(T)$. From definition of `MinDistance` and Equation 9,

$$\texttt{ConflictRange} \ = \ 2r\sin(\rho T/2) - \texttt{MinDistance}. \tag{24}$$

Since $l < \texttt{MinDistance}$, we have

$$\texttt{ConflictRange} < 2r\sin(\rho T/2) - l.$$

From the `YCNGSTYS_evader` lemma (see Appendix A), we have

$$l \leq 2r\sin(\rho T/2) \ \wedge \ 0 \leq \rho T \leq 2 \ \supset \ 2r\sin(\rho T/2) - l \leq e(T)$$

from which the desired result follows by transitivity. The premise of lemma `YCNGSTYS_evader` is discharged by establishing

$$\texttt{MinDistance} \leq 2r\sin(\rho T/2)$$

via Equation 24, and applying the assumption that $l < \texttt{MinDistance}$. □

3.6 Theorem no_conflict_Omega

$$T \geq 0 \ \wedge \ l > \texttt{ConflictRange} + v_i \ \wedge \ \rho T \leq \pi - \rho \ \wedge \ \texttt{Omega}(\beta + \theta_0)$$
$$\supset$$
$$\neg\texttt{conflict}_{ie}(T).$$

Proof. To establish $\neg\texttt{conflict}_{ie}(T)$, we show $e(T) > \texttt{ConflictRange}$. We split into two cases: $0 \leq T < 1$ and $1 \leq T$.

1. Case $0 \leq T < 1$. From lemma `YCNGFTYS_evader`, instantiating t with T, we have

$$l \geq v_i T \ \supset \ l - v_i T \leq e(t) \tag{25}$$

   Since $1 > T$, $l - v_i T > l - v_i$. But $l - v_i > \texttt{ConflictRange} > 0$, then the premise $l \geq v_i T$ of Formula 25 holds. Hence, $e(t) \geq l - v_i T > l - v_i > \texttt{ConflictRange}$.

2. Case $1 \leq T$. By the `majoration` lemma (Lemma 7)):

$$e(T)^2 \geq \hat{x_i}(T)^2.$$

   Appliying squared root to both sides results in

$$e(T) > \hat{x_i}(T).$$

   By the `no_conflict_xp_1_Omega` lemma (see Appendix):

$$1 \leq T \ \wedge \ \rho T \leq \pi - \rho \ \wedge \ \texttt{Omega}(\beta + \theta_0) \ \supset \ \hat{x_i}(T) > \texttt{ConflictRange}.$$

   Transitivity yields the desired result.

□

3.7 Theorem ails_no_conflict_tau_le0

$$v = v_i = v_e = 250 \ \wedge \ \texttt{AlertRange} = 1400 \ \wedge \ 9.5 \leq T \leq 10 \ \wedge$$
$$\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \ \neg\texttt{Omega}(\beta + \theta_0) \ \wedge$$
$$d > \texttt{AlertRange} \ \wedge \ \tau(0) \leq 0$$
$$\supset$$
$$\neg\texttt{conflict}_{ie}.$$

Proof. This theorem follows immediately from Lemma `Alpha_d_AlertRange` (Lemma 8) and the following three lemmas:

- Lemma `cos_no_conflict`:

$$v = v_i = v_e = 250 \ \wedge \ \texttt{AlertRange} = 1400 \ \wedge \ 9.5 \leq T \leq 10 \ \wedge$$
$$\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge$$
$$\neg\texttt{Alpha}(\beta) \ \wedge \ \neg\texttt{Omega}(\beta + \theta_0) \ \wedge \ \cos(\theta_0 + \beta) \leq \cos(\beta)$$
$$\supset$$
$$\neg\texttt{conflict}_{ie}(T).$$

- Lemma `R_T_d_diff`:

$$l > 0 \ \supset \ R(T) \geq d \iff \cos(\theta_0 + \beta) \leq \cos(\beta).$$

- Lemma `tau_le_0_diverg`:

$$\tau(0) \leq 0 \ \supset \ R(T) \geq d.$$

□

Proof of lemma `cos_no_conflict`. We establish $e(T)^2 > \texttt{ConflictRange}^2$. From the `isometric_evader` lemma (Lemma 5), substituting T for t, we have:

$$e(T)^2 = \hat{x}_i(T)^2 + \hat{y}_i(T)^2.$$

From lemmas `xpt` and `ypt` (see Appendix A), instantiating t with T, we get:

$$\begin{aligned} \hat{x}_i(T) &\geq r\sin(\rho T) - l\cos(\beta + \theta_0) \\ \hat{y}_i(T) &\geq l\sin(\beta + \theta_0) + r(\cos(\rho T) - 1). \end{aligned}$$

We now split on the two cases that come from the ¬`Omega` premise:

1. Case $0 \leq \beta + \theta_0 \leq \pi/2$. Lemma `Math_prop_no_conflict_1` (see Appendix A):

$$v = 250 \ \wedge \ 9.5 \leq T \leq 10 \ \wedge$$
$$\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge$$
$$\texttt{MinBeta} \leq a \leq \pi/2 \ \wedge$$
$$y \geq l\sin(a) + r[\cos(\rho T) - 1] \ \wedge \ x \geq r\sin(\rho T) - l\cos(a)$$
$$\supset$$
$$x^2 + y^2 > \texttt{ConflictRange}^2,$$

   where $\texttt{MinBeta} = 539/1000$, gives us the following after substituting $\theta_0 + \beta$ for a, $\hat{x}_i(T)$ for x, and $\hat{y}_i(T)$ for y:

$$\hat{x}_i(T)^2 + \hat{y}_i(T)^2 > \texttt{ConflictRange}^2.$$

   We discharge the assumption:

$$\texttt{MinBeta} \leq \theta_0 + \beta$$

by using the premise

$$\cos(\theta_0 + \beta) \leq \cos(\beta)$$

and Lemma `cos_beta_NOT_Alpha` (see Appendix A):

$$\begin{gathered} v = 250 \ \wedge \ 9.5 \leq T \leq 10 \ \wedge \ \texttt{AlertRange} = 1400 \wedge \\ \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \ \neg \texttt{Alpha}(\beta) \\ \supset \\ \cos(\beta) \leq \cos(\texttt{MinBeta}). \end{gathered}$$

Notice that cos is decreasing in the interval $[0 \ldots \pi]$.

2. This case is symmetric to the previous one. We use Lemma `Math_prop_no_conflict_2` (see Appendix A):

$$\begin{gathered} v = 250 \ \wedge \ 9.5 \leq T \leq 10 \wedge \\ \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \wedge \\ 3\pi/2 \leq a \leq 2\pi - \texttt{MinBeta} \wedge \\ y \geq l \sin(a) + r(\cos(\rho T) - 1) \ \wedge \ x \geq r \sin(\rho T) - l \cos(a) \\ \supset \\ x^2 + y^2 > \texttt{ConflictRange}^2, \end{gathered}$$

and the fact that cos is increasing in the interval $[3\pi/2 \ldots 2\pi]$.

□

Proof of lemma `R_T_d_diff`. We must establish

$$l > 0 \ \supset \ R(T) \geq d \iff \cos(\theta_0 + \beta) \leq \cos(\beta).$$

From lemma `R_T` (Lemma 9), we have

$$R(T)^2 = (l \cos(\beta + \theta_0) - vT)^2 + (l \sin(\beta + \theta_0))^2.$$

Simplifying the right side we have

$$\begin{aligned} R(T)^2 &= l^2 \cos(\beta + \theta_0)^2 - 2vTl \cos(\beta + \theta_0) + (vT)^2 + l^2 \sin(\beta + \theta_0)^2 \\ &= l^2 - 2vTl \cos(\beta + \theta_0) + (vT)^2. \end{aligned}$$

From Figure 7, we get

$$d^2 = (vT)^2 + l^2 - 2vTl \cos(\beta).$$

Subtracting d^2 from $R(T)^2$ yields:

$$R(T)^2 - d^2 = 2vTl(\cos(\beta) - \cos(\beta + \theta_0)).$$

Since $2vTl > 0$, we have

$$R(T)^2 - d^2 \geq 0 \iff \cos(\beta) - \cos(\beta + \theta_0) \geq 0.$$

The desired result follows from the fact that $R(T) \geq d \iff R(T)^2 - d^2 \geq 0$.

□

Proof of lemma `tau_le_0_diverg`. To establish

$$\tau(0) \leq 0 \supset R(T) \geq d,$$

we begin with lemma `asymptotic_increase_tau` (Lemma 3):

$$\tau(t) \leq t_1 \leq t_2 \supset R(t + t_2) \geq R(t + t_1).$$

Substituting T for t_2 and 0 for t and t_1, we have:

$$R(T) \geq R(0).$$

But by definition, $R(0) = d$, from which the desired result follows trivially. □

4 Verification of the AILS Alerting Algorithm

The AILS alerting algorithm determines when an alarm will be triggered by generating possible collision trajectories for the aircraft involved in the parallel landing.[5] Distance and times of minimum approaches for the generated trajectories are compared against distance and time alert thresholds. Collision trajectories are calculated based on projection of aircraft states, which consist of current position, heading, speed, and bank angle. Operational requirements for the AILS concept state that both aircraft are in the same horizontal plane and that the ground speeds of the aircraft are constant. We choose a conservative value of $v = 250$ feet per second for the aircraft ground speed.

At the beginning of the algorithm, one aircraft is considered to be the intruder and the other is considered to be the evader. The evader aircraft is assumed to fly on a straight line following the center line of its runway, which is usually called *localizer*. The algorithm is designed around the idea that the intruder aircraft is flying a circular path, based on a constant turn rate, from which it can escape on a straight line following tangential tracks separated 1.5 to 3 degrees. For all the possible trajectories, the algorithm computes time and distance at the minimum separation. If they fall in time and distance alert thresholds, then an alarm is issued. The algorithm runs on time-steps of 0.5 seconds.

The original AILS algorithm was written in FORTRAN at Langley Research Center. It has been revised several times and the latest version, flown in the Boeing 757 experimental aircraft, was written by Honeywell. That algorithm provides several levels of alarms ranging from advisory cautions to warnings according to the severity of the blundering of the intruder aircraft. A traffic warning must be followed by an escape maneuver. For the work presented in this paper, we use a higher level abstract model of the alerting algorithm, described in [7], where only traffic warning alarms are considered. That model was written in PVS (see Appendix C).

[5] In this section, we use indistinctly the words *collision* and *conflict*.

4.1 Curved Trajectories

The logic of the AILS algorithm assumes that the evader aircraft stays on the localizer during the final approach. However, the algorithm is designed to issue an alarm for any intruder trajectory that threatens the evader aircraft. An original design target for the algorithm was that an alarm should be issued at least 19 seconds before the potential collision. For a large class of trajectories, which we will call *curved*, the algorithm can be easily shown to meet this goal. A *curved* trajectory is a trajectory where the aircraft follows a circular path (at the current turn radius) until it exits the circle in a straight tangential track. Unfortunately, curved trajectories do not provide the worst-case scenario. Indeed, in our more general model of trajectories, i.e., paths only constrained by the dynamics of the aircraft, we have seen that two aircraft can approach to within 10.5 seconds of a collision *without* an alarm being triggered by the AILS algorithm. That is, for lookahead times of 10.5 seconds or greater, there exists trajectories leading to a potential collision for which an alarm will not be issued. Using a simulation tool that we have implemented in Java [7], these trajectories were first discovered. Later these trajectories were analyzed in PVS. In these trajectories the intruder gradually approaches the evader at the beginning of the final approach, but then attacks the evader in a very aggressive maneuver after approaching to within 1400 feet. The situation is illustrated in Figure 8.

In the next section we will see that for lookahead times of 10 seconds or less the AILS algorithm is *correct*, i.e., it will issue an alarm for any trajectory for which there is a potential collision.

4.2 Correctness and Certainty

The AILS algorithm has been used as a case-study for formal safety assessment via the framework proposed in Section 2. In particular, in this section, we address the formal statement of correctness and certainty properties of the AILS alerting algorithm.

In PVS, the algorithm is specified by the predicate

`ails_alert`(i, e) : `State` : `bool`

that takes the initial states of an intruder aircraft i and an evader aircraft e, and returns `true` or `false` depending on whether the alarm is issued or not. The two arguments i, e of type `State` contain the state variables that the algorithm operates on. `State` is defined as a record with fields `x,y,heading,bank` that represent the *measured* values of an aircraft's position, heading, and bank angle. In this paper, we have assumed that these measurements are made without error. The measurement process was formalized in PVS using a function `measure2state`. The net result of a measurement without error is that if tr is a trajectory consisting of functions x, θ, ϕ, and θ, the following equalities hold[6]:

$$\texttt{x}(\texttt{measure2state}(tr, t)) \;=\; x(t) \tag{26}$$

[6]Access to records is written in PVS as function calls, i.e., if s is a `State`, `x`(s) refers to the field `x` of the state s.

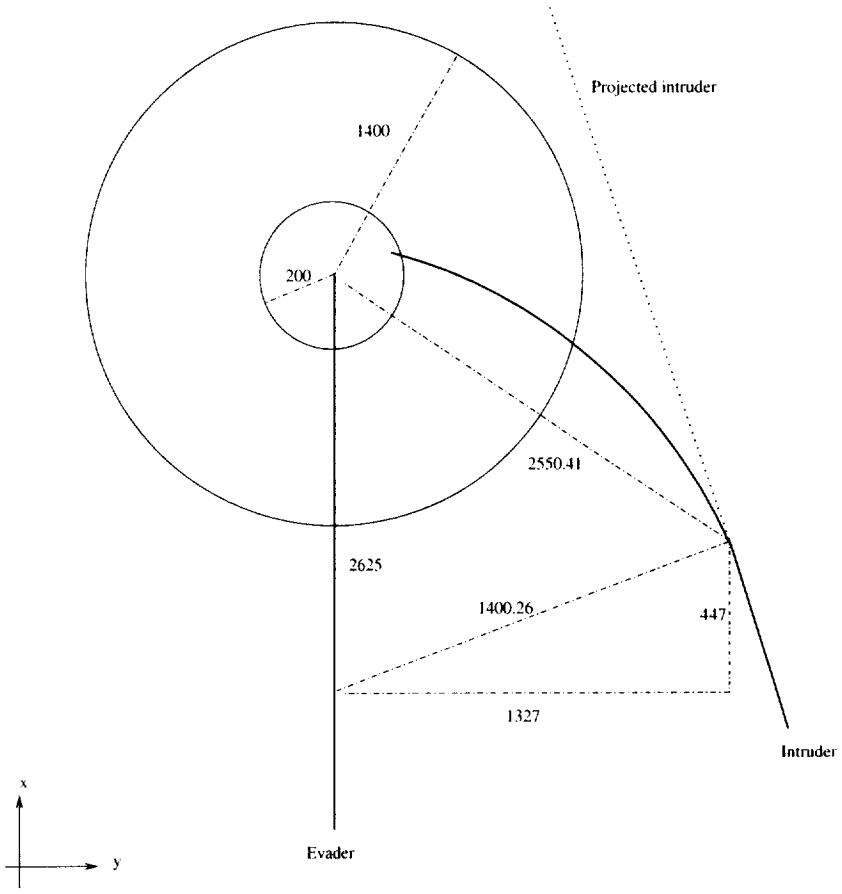


Figure 8: AILS Worst-case Scenario for T=10.5 seconds

$$\texttt{y}(\texttt{measure2state}(tr,t)) = y(t) \quad (27)$$
$$\texttt{heading}(\texttt{measure2state}(tr,t)) = \theta(t) \quad (28)$$
$$\texttt{bank}(\texttt{measure2state}(tr,t)) = \phi(t). \quad (29)$$

Although with ADS-B exchange of information the errors can be made very small, it should be included. In future work will look at incorporating measurement error into the analysis, e.g.

$$\begin{aligned}
\texttt{x}(\texttt{measure2state}(tr,t)) &= x(t)+\epsilon_x \\
\texttt{y}(\texttt{measure2state}(tr,t)) &= y(t)+\epsilon_y \\
\texttt{heading}(\texttt{measure2state}(tr,t)) &= \theta(t)+\epsilon_\theta \\
\texttt{bank}(\texttt{measure2state}(tr,t)) &= \phi(t)+\epsilon_\phi.
\end{aligned}$$

where the ϵ's are bounded according to the error inaccuracies of measurement devices.

The *correctness property* of the AILS algorithm states that if there exists an intruder trajectory that brings the two aircraft within `CollisionRange` of each other, then the algorithm will issue an alarm on the evader aircraft T seconds before of that potential collision. By using the framework developed in Section 2, we have formally proven the `ails_correctness` theorem for $9.5 \leq T \leq 10$. Therefore, each execution of the algorithm completely covers all potential collisions in a lookahead time between 9.5 to 10 seconds. Since the time step of the AILS concept, i.e., the time gap between two consecutive executions of the alerting algorithm, is 0.5 seconds, potential collisions at time less than 9.5 are covered by earlier executions of the algorithm. Due to operational constraints, when the AILS system is engaged during a final approach, there is a safe window of at least 9.5 seconds when no collision can occur.

In the following, and due to AILS operational requirements, we assume $v = v_i = v_e = 250$ and `AlertRange` $= 1400$. We also take `ConflictRange` equal to 200 feet, which is roughly the wing span of a Boeing 747.

Theorem 4 (`ails_correctness`).

$$\forall i, e.\ 9.5 \leq T \leq 10 \ \wedge\ \texttt{conflict}_{ie}(T)$$
$$\supset$$
$$\texttt{ails\_alert}(\texttt{measure2state}(i,0), \texttt{measure2state}(e,0)).$$

Proof. We split the proof in two cases, depending on whether $d \leq$ `AlertRange` or not (d is the distance from intruder to evader at time 0). The conclusion follows immediately from lemmas `ails_alarm_at_alerting_distance` and `ails_alarm_when_collision`, whose proofs are detailed in Section 4.3.

1. **`ails_alarm_at_alerting_distance`**:

$$d \leq \texttt{AlertRange} \ \supset\ \texttt{ails\_alert}(\texttt{measure2state}(i,0), \texttt{measure2state}(e,0)).$$

2. **ails_alarm_when_conflict**:

$$d > \texttt{AlertRange} \ \wedge \ 9.5 \leq T \leq 10 \ \wedge \ \texttt{conflict}_{ie}(T)$$
$$\supset$$
$$\texttt{ails\_alert}(\texttt{measure2state}(i,0), \texttt{measure2state}(e,0)).$$

□

On the other hand, the AILS algorithm is *uncertain* for $T \leq 10$ seconds, i.e., there exist scenarios where an alarm is issued but there are no possible collision trajectories within 10 seconds. In other words, false alarms may be issued.

Theorem 5 (ails_uncertainty).

$$\exists s_i, s_e : \texttt{State}.\ \forall i, e.\ s_i = \texttt{measure2state}(i,0) \ \wedge \ s_e = \texttt{measure2state}(e,0) \ \wedge \ 0 < T \leq 10$$
$$\supset$$
$$\texttt{ails\_alert}(s_i, s_e) \ \wedge \ \neg\texttt{conflict}_{ie}(T).$$

Proof. Take s_e and s_i the states such that $\texttt{x}(s_e) = 0$, $\texttt{y}(s_e) = 0$, $\texttt{heading}(s_e) = 0$, $\texttt{bank}(s_e) = 0$, $\texttt{x}(s_i) = 1400$, $\texttt{y}(s_i) = 0$, $\texttt{heading}(s_i) = 0$, and $\texttt{bank}(s_i) = 0$. We show:

1. $\texttt{ails\_alert}(s_i, s_e)$. It follows from lemma **ails_alarm_at_alerting_distance** and calculation of $d \leq \texttt{AlertRange}$ for the values of s_i and s_e.

2. $\neg\texttt{conflict}_{ie}(t)$. It follows from lemma **no_conflict_gt_max** and calculation of $l > \texttt{MaxDistance}$ for the values of s_i and s_e.

□

4.3 AILS Verification

This section is devoted to the formal proofs of lemmas **ails_alarm_at_alerting_distance** and **ails_alarm_when_conflict**. The proof of the later lemma extensively uses the conditions for conflict avoidance of Section 2.4. We refer to Appendix C for the PVS specification of the AILS alerting algorithm.

Lemma 10 (ails_alarm_at_alerting_distance).

$$d \leq \textit{AlertRange} \ \supset \ \textit{ails\_alert}(\textit{measure2state}(i,0), \textit{measure2state}(e,0)).$$

Proof. Expanding the definition of **ails_alert** (see Appendix C) yields:

```
IF ρ(bank(i)) = 0
   THEN chktrack(i, e, 0)
ELSE arc_loop(...)
ENDIF
```

We split into two cases.

1. Case $\rho(\texttt{bank}(i)) = 0$. In this case, we must prove:

$$d \leq \texttt{AlertRange} \supset \texttt{chktrack}(i, e, 0).$$

But `chktrack` expands into:

```
IF τ(i,e,0) ≤ 0
   THEN chkrange(R(i,e,0), 0)
 ELSIF τ(i,e,0) > AlertTime
       THEN R(i,e,AlertTime) ≤ AlertRange
       ELSE R(i,e,τ(i,e,0)) ≤ AlertRange
 ENDIF
```

where `AlertTime` = 19 seconds. We split into two cases.

(a) Case $\tau(i, e, 0) \leq 0$. In this case we must prove:

$$d \leq \texttt{AlertRange} \supset \texttt{chkrange}(R(i, e, 0), 0).$$

Expanding `chkrange`, we have

$$d \leq \texttt{AlertRange} \supset R(i, e, 0) \leq \texttt{AlertRange} \wedge 0 \leq \texttt{AlertTime}.$$

But $R(i, e, 0) = d$ by definition so this is clearly true.

(b) Case $\tau(i, e, 0) > \texttt{AlertTime}$. In this case we must prove:

$$d \leq \texttt{AlertRange} \supset R(i, e, \texttt{AlertTime}) \leq \texttt{AlertRange}.$$

Using lemma `asymptotic_decrease_tau` (lemma 2), we have

$$0 \leq \texttt{AlertTime} \leq \tau(0) \supset R(i, e, 0) \geq R(i, e, \texttt{AlertTime}).$$

Since $R(i, e, 0) = d$, we have by transitivity the desired result.

(c) $0 < \tau(i, e, 0) \leq \texttt{AlertTime}$. In this case we must prove:

$$d \leq \texttt{AlertRange} \supset R(\tau(i, e, 0)) \leq \texttt{AlertRange}.$$

From lemma `derivative_eq_zero_min` (lemma 2, this lemma characterizes the property of τ that $R(\tau(i, e, t))$ is a minimum):

$$R(\tau(i, e, 0)) \leq R(i, e, 0).$$

Once again since $R(i, e, 0) = d$, we reach the needed result.

2. Case $\rho(\texttt{bank}(i)) \neq 0$. Expanding `ails_alert` and `arc_loop`, and using the fact that $\texttt{mod}(0, m) = 0$, for $m \neq 0$, we end up with the goal

$$d \leq \texttt{AlertRange} \supset \texttt{chktrack}(i, e, 0),$$

which is identical to the result proved in the previous case. This means that only one tangential projection is necessary to issue an alarm.

□

Lemma 11 (`alarm_when_conflict`).

$$\begin{array}{c} d > \mathit{AlertRange} \ \wedge \ 9.5 \leq T \leq 10 \ \wedge \ \mathit{conflict}_{ie}(T) \\ \supset \\ \mathit{ails\_alert}(\mathit{measure2state}(i, 0), \mathit{measure2state}(e, 0)). \end{array}$$

Proof. First, by simple calculations we get

$$10\rho \leq 2 \tag{30}$$
$$10\rho \leq \pi - \rho. \tag{31}$$

Now, we use hypothesis $\texttt{conflict}_{ie}(T)$, to derive:

1. $l \leq \texttt{MaxDistance}$, from Theorem `no_conflict_gt_max` (see Section 3.4),
2. $l \geq \texttt{MinDistance}$, from Theorem `no_conflict_lt_min` (see Section 3.5) and Formula 30,
3. $\neg\texttt{Omega}(\beta + \theta_0)$, from Theorem `no_conflict_Omega` (see Section 3.6), Formula 31, and $l > \texttt{ConflictRange}$ (since $l \geq \texttt{MinDistance}$), and
4. $\tau(0) > 0$, from Theorem `ails_no_conflict_tau_le0` (see Section 3.7), (1), (2), (3), and hypothesis $d > \texttt{AlertRange}$.

Lemma `ails_alarm_tau_gt0` (see Appendix C):

$$\begin{array}{c} \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \ 9.5 \leq T \leq 10 \ \wedge \\ \neg\texttt{Omega}(\beta + \theta_0) \ \wedge \ \tau(0) > 0 \ \wedge \ \texttt{conflict}_{ie}(T) \\ \supset \\ \texttt{ails\_alert}(\texttt{measure2state}(i, 0), \texttt{measure2state}(e, 0)) \end{array}$$

yields the result. □

The proofs of lemma `alarm_at_alerting_distance` and and lemma `alarm_tau_gt0` only use a small part of the *potential* capability of `ails_alert`. The `chktrack` function is called recursively within `ails_alert` when the intruder's bank angle is not 0. The net effect is that `chktrack` is executed against a sequence of tangents (about 1 to 3 seconds apart) from

the projected curved path of the intruder. Interestingly, the correctness property (i.e., alarm property) only depends on the existence of the first `chktrack` execution. In other words, the `ails_alert` function could be reduced to a single `chktrack` and the theorem would still hold. However, the presence of these other `chktrack` executions enables the algorithm to often issue an alarm earlier than the worst-case time. We have shown that in the worst case, even with these extra `chktrack` executions present, there exists a trajectory where the alarm is not issued until 10.5 seconds prior to a potential conflict (Figure 8). Thus, the simplified algorithm has exactly the same worst-case performance as `ails_alert` but may have an inferior average performance[7]. However, this is offset by the fact that the simpler algorithm is far less susceptible to *false alarms*. In this context, we say that a false alarm occurs when the algorithm issues an alarm and there are *no* feasible trajectories that carry the intruder within the conflict region[8]. We have demonstrated that there are scenarios where `ails_alert` will issue an alarm even though there are *no* feasible trajectories that lead to a conflict. Thus, `ails_alert` does not satisfy the certainty property. We have not yet explored whether the simpler algorithm or a variation of it satisfies the certainty property.

5 Conclusion

In this paper, we have presented the foundation for a new approach to verifying the safety of conflict detection algorithms that may one day be deployed in the national airspace. Such algorithms are an enabling technology for free flight, where pilots are allowed to fly their own preferred trajectories. The introduction of these algorithms in a free-flight context raises significant safety issues. Historically the trajectories of aircraft have been managed by ground controllers through use of aircraft position data obtained from radar. The primary responsibility for maintaining aircraft separation has been borne by the air traffic controller. But under a free-flight approach, much of the responsibility for maintaining separation will be transfered to the pilots *and the software which provides them aircraft positions and warnings of potential conflicts*. We believe that current methods for gaining assurance about the safety of ground-based decision-aid software *are inadequate* for many of the software systems that will be deployed in the future in support of free flight. The current approach is based on human-factors based experimentation using high fidelity simulations. When the responsibility for safety resides in the human controller, this is clearly an appropriate approach. The primary question to be answered is whether the software provides the controllers with useful information that aids them in their decision making. But as software takes on more and more of the responsibility for generating aircraft trajectories and detecting potential conflicts and perhaps even producing (and executing?) the evasive maneuvers, we will need additional tools to guarantee safety. It is our view that the correctness of the algorithm must be established for *all possible* situations. Simulation and testing cannot accomplish

[7]How one might formally capture the notion of *average performance* is an interesting question.

[8]No algorithm can issue an alarm *only* when an actual conflict will occur, since doing so requires an accurate prediction of the actual future path of the other aircraft.

this. Although simulation and controlled experimentation are clearly necessary, they are not sufficient to guarantee safety This can only be done by analytical means, i.e. formal methods. We should also note that it will also be necessary to demonstrate that the implementation of these algorithms is correct. This refinement verification, in our view, must also be accomplished using formal methods. We hope to explore this issue with our colleagues in future work.

The trajectory model used in this paper is the final result after experimenting with several other alternatives. Earlier work looked at discrete versions with the expectation that this would lead to a more tractable verification task. Unfortunately the discretization of the trajectories led to significant (and accumulating) modeling error that led to erroneous conclusions. In the end, we settled on modeling trajectories as differentiable functions over real numbers. These trajectories are constrained by the dynamics of an aircraft. These constraints enable one to establish high-level properties that delineate when a conflict is possible. In this paper we have developed a formal theory about trajectories that can serve as the basis for the formal analysis of conflict detection and resolution (CD&R) algorithms. Several limitations to this formal theory will be addressed in future work: (1) the theory only deals with 2 aircraft, (2) the vertical dimension is not modeled, and (3) aircraft data measurement errors are not modeled.

Because the trajectories of the aircraft are modeled by differentiable functions over real numbers and the discrete algorithms are periodically executed on a digital computer, the problem domain falls into the domain of hybrid models. The hybrid nature of the domain makes the verification problem especially difficult. Automatic methods such as model checking cannot directly handle the continuous trajectories, and discretization leads to unacceptable errors. We are forced to reason about such systems in the context of a fully general theorem prover designed to handle a rich logic such as higher-order logic, type theory, or ZFC set theory. We have used SRI International's PVS theorem prover in our work and found it to be sufficient to handle the problem but our work was often impeded by PVS's baroque method for dealing with nonlinear arithmetic. Although PVS provides a splendid suite of decision procedures that can automate much of the tedium of theorem proving, in this arena, they are not adequate. Simple properties of the reals must be manually extracted from the PVS prelude, manually instantiated, and directly invoked during the proof. Also it is often necessary to perform case splits to get a formula into a form that can be handled by the prover. Current work at SRI funded by NASA Langley is seeking to improve the PVS capability for reasoning about formulas containing nonlinear arithmetic.

Future work will concentrate on applying this modeling framework to specific CD&R algorithms and perhaps to self-spacing and merging algorithms designed to increase capacity in the terminal area. We would also like to develop formal methods for analyzing conflict resolution schemes and the safety of algorithmically-generated evasive maneuvers [8]. The CD&R methods must be generalized to cover sets of aircraft constrained by formally specified notions of aircraft density (static or dynamic). Finally, we would like to generalize the methods to encompass measurement error and data errors. This is a necessary step towards developing formal methods useful for the design and implementation phases of realistic

avionics.

Acknowledgment

The authors would like to thank Alfons Geser and Michael Holloway for their helpful comments on preliminary versions of this manuscript.

References

[1] Advanced Air Transportation Technologies (AATT) Project. Concept definition for Distributed Air/Ground Traffic Management (DAG-TM), version 1.0. NASA Ames Research Center - NASA Langley Research Center. URL: `http://www.asc.nasa.gov/aatt/dag.html`, 1999.

[2] J.-M. Alliot and N. Durand. FACES: A free flight autonomous and coordinated embarked solver. 2nd USA/EUROPE ATM R&D seminar, 1998.

[3] B. Dutertre. Elements of mathematical analysis in PVS. In J. Von Wright, J. Grundy, and J. Harrison, editors, *Ninth international Conference on Theorem Proving in Higher Order Logics TPHOL*, volume 1125 of *Lecture Notes in Computer Science*, pages 141–156, Turku, Finland, August 1996. Springer Verlag.

[4] K. Bilimoria. A geometric optimization approach to aircraft conflict resolution. In *Guidance, Navigation, and Control Conference*, volume AIAA 2000-4265, Denver, CO, August 2000.

[5] K. Bilimoria, K. Sheth, H. Lee, and S. Grabbe. Performance evaluation of airborne separation assurance for free flight. In *Guidance, Navigation, and Control Conference*, volume AIAA 2000-4269, Denver, CO, August 2000.

[6] B. Carpenter and J. Kuchar. Probability-based collision alerting logic for closely-spaced parallel approach. In *AIAA 35th Aerospace Sciences Meeting*, volume AIAA-97-0222, Reno, NV, January 1997.

[7] V. Carreño and C. Muñoz. Aircraft trajectory modeling and alerting algorithm verification. In J. Harrison and M. Aagaard, editors, *Theorem Proving in Higher Order Logics: 13th International Conference, TPHOLs 2000*, volume 1869 of *Lecture Notes in Computer Science*, pages 90–105. Springer-Verlag, 2000. An earlier version appears as report NASA/CR-2000-210097 ICASE No. 2000-16.

[8] G. Dowek, C. Muñoz, and A. Geser. Tactical conflict detection and resolution in a 3-D airspace. Technical Report NASA/CR-2001-210853 ICASE Report No. 2001-7, ICASE-NASA Langley, ICASE Mail Stop 132C, NASA Langley Research Center, Hampton VA 23681-2199, USA, April 2001.

[9] B. Fuchssteiner. *MuPAD User's Manual.* John Wiley and Sons, Chichester, New York, first edition, March 1996. Includes a CD for Apple Macintosh and UNIX.

[10] J. Hoekstra, R. Ruigrok, R. van Gent, J. Visser, B. Gijsbers, M. Valenti, W. Heesbeen, B. Hilburn, J. Groeneweg, and F. Bussink. Overview of NLR free flight project 1997-1999. Technical Report NLR-CR-2000-227, National Aerospace Laboratory NLR, May 2000.

[11] H. Kremer, B. Bakker, and H. Blom. Probabilistic versus geometric conflict probing. Air Traffic Management Research and Development Seminar. Saclay, France, 1997.

[12] J. Kuchar and Jr. R. Hansman. A unified methodology for the evaluation of hazard alerting systems. Technical Report ASL-95-1, ASL MIT Aeronautical System Laboratory, January 1995.

[13] J. Kuchar and L. Yang. Survey of conflict detection and resolution modeling methods. In *AIAA Guidance, Navigation, and Control Conference*, volume AIAA-97-3732, pages 1388–1397, New Orleans, LA, August 1997.

[14] J. Lygeros and N. Lynch. On the formal verification of the TCAS conflict resolution algorithms. In *Proceedings 36th IEEE Conference on Decision and Control*, San Diego, CA, pages 1829–1834, December 1997. Extended abstract.

[15] S. Owre, J. M. Rushby, and N. Shankar. PVS: A prototype verification system. In Deepak Kapur, editor, *11th International Conference on Automated Deduction (CADE)*, volume 607 of *Lecture Notes in Artificial Intelligence*, pages 748–752, Saratoga, NY, June 1992. Springer-Verlag.

[16] Radio Technical Commission for Aeronautics. Final report of the RTCA board of directors' select committee on free flight. Technical Report Issued 1-18-95, RTCA, Washington, DC, 1995.

[17] L. Rine, T. Abbott, G. Lohr, D. Elliott, M. Waller, and R. Perry. The flight deck perspective of the NASA Langley AILS concept. Technical Report NASA/TM-2000-209841, NASA, January 2000.

[18] C. Tomlin, G. Pappas, and S. Sastry. Conflict resolution for air traffic management: A study in multi-agent hybrid systems. *IEEE Transactions on Automatic Control*, 43(4), April 1998.

[19] T. Williams, C. Kelley, J. Campbell, D. Kotz, and R. Lang. *GNUPLOT An Interactive Plotting Program.* Free Software Foundation, 675 Mass Ave, Cambridge, MA 02139, USA, Tel: (617) 876-3296, USA, August 1990.

Appendix A: Technical Lemmas

Lemma 12 (Dxp).

$$\hat{x}'(t) = v\cos(\theta_t - \theta_0).$$

Proof. We begin with the definition of $\hat{x}$ (the x-coordinate after rotating the axes by θ_0. See Formula 22):

$$\hat{x}(t) = \cos(\theta_0)[x_i(t) - x_e(T)] + \sin(\theta_0)[y_i(t) - y_e(T)].$$

Differentiating we have:

$$\hat{x}'(t) = \cos(\theta_0)\, x_i'(t) + \sin(\theta_0)\, y_i'(t).$$

From the aircraft dynamics (see formulas 1 and 2) we have $x_i'(t) = v\cos(\theta_t)$ and $y_i'(t) = v\sin(\theta_t)$ which leads us to:

$$\hat{x}'(t) = \cos(\theta_0)\, v\cos(\theta_t) + \sin(\theta_0)\, v\sin(\theta_t).$$

Applying the cosine of the difference of two angles trigonometric identity, we have

$$\hat{x}'(t) = v\cos(\theta_t - \theta_0).$$

□

Lemma 13 (Dyp).

$$\hat{y}'(t) = v\sin(\theta_t - \theta_0).$$

Proof. We begin with the definition of $\hat{y}$ (the y-coordinate after rotating the axes by θ_0. See Formula 23):

$$\hat{y}(t) = -\sin(\theta_0)[x_i(t) - x_e(T)] + \cos(\theta_0)[y_i(t) - y_e(T)].$$

Differentiating we have:

$$\hat{y}'(t) = -\sin(\theta_0)\, x_i'(t) + \cos(\theta_0)\, y_i'(t).$$

From the aircraft dynamics (see formulas 1 and 2) we have $x_i'(t) = v\cos(\theta_t)$ and $y_i'(t) = v\sin(\theta_t)$ which leads us to:

$$\hat{y}'(t) = -\sin(\theta_0)\, v\cos(\theta_t) + \cos(\theta_0)\, v\sin(\theta_t).$$

Applying the sine of the difference of two angles trigonometric identity, we have

$$\hat{y}'(t) = v\sin(\theta_t - \theta_0).$$

□

Lemma 14 (YCNGFTYS_evader).

$$t \geq 0 \ \wedge \ l \geq vt \ \supset \ l - vt \leq e(t).$$

Proof. From Figure 5, we see that

$$l \leq e(t) + i(t).$$

From Theorem YCNGFTYS(Theorem 1), and definition of $i(t)$, we have

$$i(t) \leq vt.$$

Thus, $l \leq e(t) + vt$ and hence

$$l - vt \leq e(t).$$

□

Lemma 15 (YCNGSTYS_evader).

$$t \geq 0 \ \wedge \ \rho t \leq 2 \ \wedge \ l \leq 2r \sin(\rho t/2) \ \supset \ 2r \sin(\rho t/2) - l \leq e(t).$$

Proof. Appyling the triangle inequality to Figure 5, we have

$$i(t) \leq l + e(t).$$

Rearranging

$$e(t) \geq i(t) - l.$$

From Theorem 2 and definition of $i(t)$, we have $i(t) \geq 2r \sin(\rho t/2)$, which give us:

$$e(t) \ \geq \ 2r \sin(\rho t/2) - l.$$

□

Lemma 16 (theta_inv).

$$-\rho t \leq \theta_t - \theta_0 \leq \rho t.$$

Proof. By Formula 4 we have

$$|\phi(t)| \leq \texttt{MaxBank}.$$

Monotonic increasing property of tangent function over interval $[-\pi/4, \pi/4]$ yields:

$$\tan(-\texttt{MaxBank}) \leq \tan(\phi(t)) \leq \tan(\texttt{MaxBank}).$$

Multiplying by g/v yields:

$$\frac{g \tan(-\texttt{MaxBank})}{v} \leq \frac{g \tan(\phi(t))}{v} \leq \frac{g \tan(\texttt{MaxBank})}{v}.$$

By aircraft dynamics (see Equation 3), we have:

$$\frac{g\tan(-\texttt{MaxBank})}{v} \leq \theta_t{}' \leq \frac{g\tan(\texttt{MaxBank})}{v}.$$

But, $\tan(-\texttt{MaxBank}) = -\tan(\texttt{MaxBank})$ and by definition $\rho = g\tan(\texttt{MaxBank})/v$, giving us our desired result:

$$-\rho \leq \theta_t{}' \leq \rho.$$

Integrating from 0 to t yields (Theorem 3):

$$-\rho t \leq \theta_t - \theta_0 \leq \rho t.$$

□

Lemma 17 (Dxp0_PI).

$$0 \leq t \ \wedge \ \rho t \leq \pi \ \supset \ \hat{x}'(t) \geq v\cos(\rho t).$$

Proof. From lemma **theta_inv** we have:

$$-\rho t \leq \theta_t - \theta_0 \leq \rho t.$$

We consider two cases.

1. Case $\theta_t - \theta_0 \geq 0$. Since the cos function is monotonically decreasing over $[0, \pi]$, we have

   $$\cos(\theta_t - \theta_0) \geq \cos(\rho t).$$

   From lemma **Dxp**, we know $\hat{x}'(t) = v\cos(\theta_t - \theta_0)$, we conclude

   $$\hat{x}'(t) = v\cos(\theta_t - \theta_0) \geq v\cos(\rho t).$$

2. Case $\theta_t - \theta_0 \leq 0$. From lemma **theta_inv** we have:

   $$\theta_t - \theta_0 \geq -\rho t.$$

   Since the cos function is monotonically increasing over $[-\pi, 0]$, we have

   $$\cos(\theta_t - \theta_0) \geq \cos(-\rho t).$$

   Since $\cos(-\rho t) = \cos(\rho t)$, we conclude

   $$\hat{x}'(t) = v\cos(\theta_t - \theta_0) \geq v\cos(\rho t).$$

□

Lemma 18 (xpt).

$$0 \leq t \ \wedge \ \rho t \leq \pi \ \supset \ \hat{x}(t) - \hat{x}(0) \geq h_x(t),$$

where $h_x(t) = r \sin(\rho t)$.

Proof. From lemma DxpO_PI we have:

$$0 \leq t \ \wedge \ \rho t \leq \pi \ \supset \ \hat{x}'(t) \geq v \cos(\rho t).$$

By differentiation:

$$\hat{x}'(t) \geq \frac{d}{dt}[\frac{v}{\rho} \sin(\rho t)].$$

Integrating both sides:

$$\int_0^t \hat{x}'(t)dt \geq \int_0^t \frac{d}{dt}[\frac{v}{\rho} \sin(\rho t)]dt.$$

This yields

$$\hat{x}(t)\Big|_0^t \geq \frac{v}{\rho} \sin(\rho t)\Big|_0^t.$$

Simplifying and using definition of r $(r = v/\rho)$, we conclude

$$\hat{x}(t) - \hat{x}(0) \geq v \sin(\rho t) = h_x(t).$$

□

Lemma 19 (DypO_PI2).

$$0 \leq t \ \wedge \ \rho t \leq \pi/2 \ \supset \ -v \sin(\rho t) \leq \hat{y}'(t) \leq v \sin(\rho t).$$

Proof. From lemma theta_inv we have:

$$-\rho t \leq \theta_t - \theta_0 \leq \rho t.$$

Since $|\rho t| \leq \pi/2$ and $|\theta_t - \theta_0| \leq \pi/2$ and sin is monotonically increasing over this region, we have

$$-\sin(\rho t) \leq \sin(\theta_t - \theta_0) \leq \sin(\rho t).$$

Multiplying through by v and using lemma Dyp: $\hat{y}'(t) = v \sin(\theta_t - \theta_0)$, yields:

$$-v \sin(\rho t) \leq \hat{y}'(t) \leq v \sin(\rho t).$$

□

Lemma 20 (ypt).

$$0 \leq t \ \wedge \ \rho t \leq \pi/2 \ \supset \ h_y(t) - h_y(0) \leq \hat{y}(t) - \hat{y}(0) \leq h_y(0) - h_y(t),$$

where $h_y(t) = r \cos(\rho t)$.

Proof. From lemma `Dyp0_PI2`, we get:

$$-v\sin(\rho t) \leq \hat{y}'(t) \leq v\sin(\rho t).$$

By definition $h_y(t) = r\cos(\rho t)$, so ${h_y}'(t) = -r\,\rho\sin(\rho t) = -v\sin(\rho t)$, so we obtain

$${h_y}'(t) \leq \hat{y}'(t) \leq -{h_y}'(t).$$

Integrating yields (see Theorem 3):

$$\int_0^t {h_y}'(t)dt \leq \int_0^t \hat{y}'(t)dt \leq \int_0^t -{h_y}'(t)dt$$

and evaluating gives us:

$$h_y(t) - h_y(0) \leq \hat{y}(t) - \hat{y}(0) \leq h_y(0) - h_y(t).$$

□

Lemma 21 (`cos_beta_NOT_Alpha`).

$$\begin{gathered} v = 250 \ \wedge\ 9.5 \leq T \leq 10 \ \wedge\ \texttt{AlertRange} = 1400 \ \wedge \\ \neg\texttt{Alpha}(\beta) \ \wedge\ \texttt{MinDistance} \leq l \ \wedge\ l \leq \texttt{MaxDistance} \\ \supset \\ \cos(\beta) \leq \cos(\texttt{MinBeta}). \end{gathered}$$

Proof. We begin by restating the formula using $\neg A \ \wedge\ B \ \supset\ C \iff \neg C \ \wedge\ B \ \supset\ A$, a tautology:

$$\begin{gathered} \cos(\beta) > \cos(\texttt{MinBeta}) \ \wedge\ \texttt{MinDistance} \leq l \ \wedge\ l \leq \texttt{MaxDistance} \ \wedge \\ v = 250 \ \wedge\ 9.5 \leq T \leq 10 \ \wedge\ \texttt{AlertRange} = 1400 \\ \supset \\ \texttt{Alpha}(\beta). \end{gathered}$$

We must establish `Alpha`(β), which is defined as below:

$$l\cos(\beta) \geq \frac{(v\,T)^2 + l^2 - \texttt{AlertRange}^2}{2Tv}.$$

Clearly it suffices to show that

$$l\cos(\texttt{MinBeta}) \geq \frac{(v\,T)^2 + l^2 - \texttt{AlertRange}^2}{2Tv}.$$

Multiplying both sides by $2Tv$ yields

$$2Tvl\cos(\texttt{MinBeta}) \geq (v\,T)^2 + l^2 - \texttt{AlertRange}^2,$$

which is true for $9.5 \leq T \leq 10$, $v = 250$, and $\texttt{AlertRange} = 1400$. □

Lemma 22 (Alpha_d_alertRange).

$$\texttt{Alpha}(\beta) \iff d \leq \texttt{AlertRange}.$$

Proof. By definition of Alpha, we have:

$$l\cos(\beta) \geq \frac{(vT)^2 - l^2 - \texttt{AlertRange}^2}{2vT} \iff d \leq \texttt{AlertRange}.$$

Simplifying the left-hand side, we have:

$$\texttt{AlertRange}^2 \geq (vT)^2 - l^2 - 2vTl\cos(\beta) \geq \iff d \leq \texttt{AlertRange}.$$

Now using the Law of Cosines (see Figure 7), we get $d^2 = (vT)^2 - l^2 - 2vTl\cos(\beta)$, and substituting, we have

$$d^2 \leq \texttt{AlertRange}^2 \geq \iff d \leq \texttt{AlertRange}$$

which is trivially true because d and AlertRange are distances and hence non-negative. □

Lemma 23 (xp0).

$$\hat{x}(0) = -l\cos(\theta_0 + \beta).$$

Proof. We begin with the definition of $\hat{x}$ (the x-coordinate after rotating the axes by θ_0. See Formula 22):

$$\hat{x}(t) = \cos(\theta_0)[x_i(t) - x_e(T)] + \sin(\theta_0)[y_i(t) - y_e(T)].$$

From formulas 16 and 17, we have $x_e(T) = l\cos(\beta) + x_i(0)$ and $y_e(T) = y_i(0) - l\sin(\beta)$. Substituting we have:

$$\begin{aligned}
\hat{x}(0) &= \cos(\theta_0)[-l\cos(\beta)] + \sin(\theta_0)[l\sin(\beta)] \\
&= -l\left[\cos(\theta_0)\cos(\beta) - \sin(\theta_0)\sin(\beta)\right] \\
&= -l\cos(\theta_0 + \beta).
\end{aligned}$$

The last step following from the trigonmetric identity for the cosine of the sum of two angles. □

Lemma 24 (no_conflict_xp_1_Omega).

$$1 \leq t \ \wedge \ \rho t \leq \pi - \rho \ \wedge \ \texttt{Omega}(\beta + \theta_0) \ \supset \ \hat{x}(t) > \texttt{ConflictRange}.$$

Proof. We begin with Lemma xpt:

$$0 \leq t \ \wedge \ \rho t \leq \pi \supset \hat{x}(t) - \hat{x}(0) \geq h_x(t).$$

By definition $h_x(t) = r\sin(\rho t)$, so we have

$$\hat{x}(t) - \hat{x}(0) \geq r\sin(\rho t)$$

dropping the premises. From the Lemma `xp0`:

$$\hat{x}(0) = -l\cos(\theta_0 + \beta).$$

Under the `Omega` assumption, $\cos(\theta_0 + \beta) \leq 0$, therefore $\hat{x}(0)$ is non-negative, giving us

$$\hat{x}(t) \geq r\sin(\rho t).$$

Then since $t \geq 1$, we have $\rho t \geq \rho$, and since $\rho t \leq \pi/2$, we have $\sin(\rho t) \geq \sin(\rho)$. This leads to

$$\hat{x}(t) \geq r\sin(\rho).$$

From the following `increasing_r_sin_rho` axiom:

$$r\sin(\rho) > \texttt{ConflictRange}$$

(which has been checked in MuPAD whenever $v > 210$), we have the desired result. □

Lemma 25 (`alarm_NOT_Omega_T`).

$$\begin{aligned}\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \neg\texttt{Omega}(\beta + \theta_0) \ \wedge \ \texttt{conflict}_{ie}(T) \\ \supset \ R_*(T) \leq \texttt{AlertRange}.\end{aligned}$$

Proof. We begin with Lemma `R_T`, which gives us:

$$R_*^2(T) = [l\cos(\beta + \theta_0) - vT]^2 + [l\sin(\beta + \theta_0)]^2. \tag{32}$$

From Lemma `conflict_beta_theta`, we have

$$\begin{aligned}\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \neg\texttt{Omega}(\beta + \theta_0) \ \wedge \ \texttt{conflict}_{ie}(T) \\ \supset \ ((\beta + \theta_0 \leq \texttt{MinBeta}) \vee (\beta + \theta_0 > 2\pi - \texttt{MinBeta})).\end{aligned}$$

This gives us two cases to consider:

1. Case $\beta + \theta_0 \leq \texttt{MinBeta}$. From Lemma `Math_prop_alarm_1`, after substituting $\beta + \theta_0$ for a, we have

   $$\begin{aligned}\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \ 0 \leq \beta + \theta_0 \ \wedge \ \beta + \theta_0 \leq \texttt{MinBeta} \\ \supset \ [l\cos(\beta + \theta_0) - vT]^2 + [l\sin(\beta + \theta_0)]^2 \leq \texttt{AlertRange}^2.\end{aligned}$$

   Using this and Equation 32, we have[9]

   $$R_*^2(T) \leq \texttt{AlertRange}^2$$

   from which the desired result $R_*(T) \leq \texttt{AlertRange}$ immediately follows since `AlertRange` is positive.

[9]The angle β is defined such that $0 \leq \beta + \theta_0 < 2\pi$.

2. Case $\beta + \theta_0 \geq 2\pi - \texttt{MinBeta}$. From Lemma `Math_prop_alarm_2`, after substituting $\beta + \theta_0$ for a, we have

$$\begin{aligned} &\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ &2\pi - \texttt{MinBeta} \leq \beta + \theta_0 \ \wedge \ \beta + \theta_0 \leq 2\pi \\ \supset \ &[l\cos(\beta + \theta_0) - vT]^2 + [l\sin(\beta + \theta_0)]^2 \leq \texttt{AlertRange}^2. \end{aligned}$$

Using this and Equation 32, we have

$$R_*^2(T) \leq \texttt{AlertRange}^2$$

from which the desired result $R_*(T) \leq \texttt{AlertRange}$ immediately follows since `AlertRange` is positive.

□

Lemma 26 (`alarm_NOT_Omega_tau`).

$$\begin{aligned} &\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ &\neg\texttt{Omega}(\theta_0 + \beta) \ \wedge \\ &\texttt{conflict}_{ie}(T) \ \wedge \ \tau(0) > 0 \\ \supset \ &R_*(\tau(0)) \leq \texttt{AlertRange}. \end{aligned}$$

Proof. This proof follows easily from the analysis of two cases:

1. Case $R_*(T) \leq \texttt{AlertRange}$. If $T \leq \tau(0)$, then from Lemma `asymptotic_decrease_tau` we have $R_*(\tau(0)) < R_*(T)$ which gives us the desired result immediately by transitivity. Otherwise (i.e., $T > \tau(0)$), we use Lemma `asymptotic_increase_tau` which gives us $R_*(\tau(0)) \leq R_*(T)$ from which the desired result immediately follows by transitivity.

2. Case $R_*(T) > \texttt{AlertRange}$. From Lemma `alarm_NOT_Omega_T`:

$$\begin{aligned} &\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ &\neg\texttt{Omega}(\beta + \theta_0) \ \wedge \ \texttt{conflict}_{ie}(T) \\ \supset \ &R_*(T) \leq \texttt{AlertRange} \end{aligned}$$

from which the desired result immediately follows by transitivity.

□

Lemma 27 (`alarm_NOT_Omega_AlertTime`).

$$\begin{aligned} &\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ &\neg\texttt{Omega}(\beta + \theta_0) \ \wedge \ \texttt{conflict}_{ie}(T) \ \wedge \\ &\tau(0) > \texttt{AlertTime} \\ \supset \ &R_*(\texttt{AlertTime}) \leq \texttt{AlertRange}. \end{aligned}$$

Proof. From Lemma `alarm_NOT_Omega_T`, we have

$$\begin{aligned}\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \neg\texttt{Omega}(\beta+\theta_0) \ \wedge \ \texttt{conflict}_{ie}(T) \\ \supset \ R_*(T) \leq \texttt{AlertRange}\end{aligned}$$

and from Lemma `asymptotic_decrease_tau`, we have $R_*(\texttt{AlertTime}) < R_*(T)$. Combining these two results gives us the desired result immediately by transitivity. □

Lemma 28 (`conflict_beta_theta`).

$$\begin{aligned}\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \neg\texttt{Omega}(\beta+\theta_0) \ \wedge \ \texttt{conflict}_{ie}(T) \\ \supset \ ((\beta+\theta_0 \leq \texttt{MinBeta}) \vee (\beta+\theta_0 > 2\pi - \texttt{MinBeta})).\end{aligned}$$

Proof. From the definition of $\texttt{conflict}_{ie}$, we have

$$\sqrt{(x_i(T)-x_e(T))^2+(y_i(T)-y_e(T))^2} \leq \texttt{ConflictRange}.$$

Squaring both sides:

$$(x_i(T)-x_e(T))^2+(y_i(T)-y_e(T))^2 \leq \texttt{ConflictRange}^2.$$

From Lemma `isometric_evader`, we have

$$e(T)^2 = \hat{x}_i(T)^2 + \hat{y}_i(T)^2.$$

By definition of $e(T)$, we have:

$$(x_i(T)-x_e(T))^2+(y_i(T)-y_e(T))^2 = \hat{x}_i(T)^2 + \hat{y}_i(T)^2.$$

By substitution, we obtain:

$$\hat{x}_i(T)^2 + \hat{y}_i(T)^2 \leq \texttt{ConflictRange}.$$

From Lemma `xpt_PI`, we obtain

$$\begin{aligned}0 \leq T \ \wedge \ \rho T \leq \pi \\ \supset \ \hat{x}(T) \geq r\sin(\rho T) - l\cos(\beta+\theta_0)\end{aligned}$$

and from Lemma `yp_PI2`, we obtain

$$\begin{aligned}0 \leq T \ \wedge \ \rho T \leq \pi/2 \ \supset \\ \hat{y}(T) \geq l\sin(\beta+\theta_0) + r\,(\cos(\rho T)-1) \ \wedge \\ \hat{y}(T) \leq l\sin(\beta+\theta_0) - r\,(\cos(\rho T)-1).\end{aligned}$$

Direct calculation provides $\rho T < \pi/2$ which discharges the premises of these two lemmas. Then from Lemma `Math_prop_no_conflict_1` (substituting $\beta + \theta_0$ for a, $\hat{x}(T)$ for x, and $\hat{y}(T)$ for y), we get

$$\begin{aligned} \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \texttt{MinBeta} \leq \beta + \theta_0 \ \wedge \ \beta + \theta_0 \leq \pi/2 \ \wedge \\ \hat{y}(T) \geq l \sin(\beta + \theta_0) + r\,(\cos(\rho T) - 1) \ \wedge \\ \hat{x}(T) \geq r \sin(\rho T) - l \cos(\beta + \theta_0) \\ \supset \ \hat{x}^2(T) + \hat{y}^2(T) > \texttt{ConflictRange}^2. \end{aligned}$$

From Lemma `Math_prop_no_conflict_2` (substituting $\beta + \theta_0$ for a, $\hat{x}(T)$ for x, and $\hat{y}(T)$ for y), we get

$$\begin{aligned} \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ 3\pi/2 \leq \beta + \theta_0 \ \wedge \ \beta + \theta_0 \leq 2\pi - \texttt{MinBeta} \ \wedge \\ \hat{y}(T) \leq l \sin(\beta + \theta_0) - r\,(\cos(\rho T) - 1) \ \wedge \\ \hat{x}(T) \geq -l \cos(\beta + \theta_0) + r \sin(\rho T) \\ \supset \ \hat{x}^2(T) + \hat{y}^2(T) > \texttt{ConflictRange}^2. \end{aligned}$$

Discharging the premises of these lemmas from the main premises and derived results we obtain:

$$\begin{aligned} \texttt{MinBeta} \leq \beta + \theta_0 \ \wedge \ \beta + \theta_0 \leq \pi/2 \ \wedge \\ \supset \ \hat{x}^2(T) + \hat{y}^2(T) > \texttt{ConflictRange}^2 \end{aligned}$$

and

$$\begin{aligned} 3\pi/2 \leq \beta + \theta_0 \ \wedge \ \beta + \theta_0 \leq 2\pi - \texttt{MinBeta} \ \wedge \\ \supset \ \hat{x}^2(T) + \hat{y}^2(T) > \texttt{ConflictRange}^2. \end{aligned}$$

The contrapositive of these are:

$$\begin{aligned} \hat{x}^2(T) + \hat{y}^2(T) \leq \texttt{ConflictRange}^2 \\ \supset \ \texttt{MinBeta} > \beta + \theta_0 \vee \beta + \theta_0 > \pi/2 \end{aligned}$$

and

$$\begin{aligned} \hat{x}^2(T) + \hat{y}^2(T) \leq \texttt{ConflictRange}^2 \\ \supset \ 3\pi/2 > \beta + \theta_0 \vee \beta + \theta_0 > 2\pi - \texttt{MinBeta}. \end{aligned}$$

Combining these results we end up with

$$\begin{aligned} (\texttt{MinBeta} > \beta + \theta_0 \vee \beta + \theta_0 > \pi/2) \ \wedge \\ (3\pi/2 > \beta + \theta_0 \vee \beta + \theta_0 > 2\pi - \texttt{MinBeta}). \end{aligned}$$

But by definition of the premise ¬Omega we have:

$$\beta + \theta_0 < \pi/2 \vee \beta + \theta_0 > 3\pi/2.$$

Combining these last two results yields

$$\texttt{MinBeta} > \beta + \theta_0 \vee \beta + \theta_0 > 2\pi - \texttt{MinBeta}$$

the desired result. □

Lemma 29 (xpt_PI).

$$0 \leq t \ \wedge \ \rho t \leq \pi \ \supset \ \hat{x}(t) \geq r \sin(\rho t) - l \cos(\beta + \theta_0).$$

Proof. From Lemma xpt, we have

$$0 \leq t \ \wedge \ \rho t \leq \pi \ \supset \ \hat{x}(t) - \hat{x}(0) \geq h_x(t).$$

From Lemma xp0, we have

$$\hat{x}(0) = -l \cos(\theta_0 + \beta).$$

By definition, $h_x = r \sin(\rho t)$, so we have

$$\hat{x}(t) \geq r \sin(\rho t) - l \cos(\theta_0 + \beta)$$

the desired result. □

Lemma 30 (yp_PI2).

$$\begin{aligned} 0 \leq T \ \wedge \ \rho T \leq \pi/2 \ \supset \\ \hat{y}(T) \geq l \sin(\beta + \theta_0) + r\,(\cos(\rho T) - 1) \ \wedge \\ \hat{y}(T) \leq l \sin(\beta + \theta_0) - r\,(\cos(\rho T) - 1). \end{aligned}$$

Proof. From Lemma ypt, we have

$$\begin{aligned} 0 \leq t \ \wedge \ \rho t \leq \pi/2 \ \supset \\ \hat{y}(t) - \hat{y}(0) \geq h_y(t) - h_y(0) \ \wedge \\ \hat{y}(t) - \hat{y}(0) \leq h_y(0) - h_y(t). \end{aligned}$$

From Lemma yp0, we have

$$\hat{y}(0) = l \sin(\theta_0 + \beta).$$

By definition of $h_y = r \cos(\rho t)$ and substituting for $\hat{y}(0)$, we have

$$\begin{aligned} \hat{y}(t) - l \sin(\theta_0 + \beta) \geq r \cos(\rho t) - r \cos(0) \ \wedge \\ \hat{y}(t) - l \sin(\theta_0 + \beta) \leq r \cos(0) - r \cos(\rho t) \end{aligned}$$

which simplifies to

$$\begin{aligned} \hat{y}(t) - l \sin(\theta_0 + \beta) \geq r\,(\cos(\rho t) - 1) \ \wedge \\ \hat{y}(t) - l \sin(\theta_0 + \beta) \leq r\,(1 - \cos(\rho t)) \end{aligned}$$

from which the desired result immediately follows. □

Lemma 31 (yp0).

$$\hat{y}(0) = l\sin(\theta_0 + \beta).$$

Proof. We begin with the definition of $\hat{y}$ (the y-coordinate after rotating the axes by θ_0. Formula 23, after substituting 0 for y, becomes

$$\hat{y}(0) = \cos(\theta_0)[y_i(0) - y_e(T)] - \sin(\theta_0)[x_i(0) - x_e(T)].$$

From formulas 16 and 17, we have $x_e(T) = l\cos(\beta) + x_i(0)$ and $y_e(T) = y_i(0) - l\sin(\beta)$. Substituting we have:

$$\begin{aligned}\hat{x}(0) &= \cos(\theta_0)[l\sin(\beta)] - \sin(\theta_0)[l\cos(\beta)] \\ &= l\left[\cos(\theta_0)\sin(\beta) - \sin(\theta_0)\cos(\beta)\right] \\ &= l\sin(\theta_0 + \beta).\end{aligned}$$

The last step following from the trigonmetric identity for the sine of the sum of two angles. □

Lemma 32 (ails_alarm_tau_gt0).

$$\begin{gathered}\texttt{MinDistance} \le l \le \texttt{MaxDistance} \wedge \\ \neg\texttt{Omega}(\beta + \theta_0) \wedge \\ \tau(0) > 0 \wedge \\ \texttt{conflict}_{ie}(T) \\ \supset \\ \texttt{ails\_alert}(\texttt{measure2state}(i,0), \texttt{measure2state}(e,0)).\end{gathered}$$

Proof. We split into two cases.

1. Case $\rho(\texttt{bank(i)}) = 0$. In this case `ails_alert` simplifies to $\texttt{chktrack}(i, e, 0)$. Expanding `chktrack` we have,

   ```
   IF τ(0) > AlertTime
     THEN R*(AlertTime) ≤ AlertRange
   ELSE R*(τ(0)) ≤ AlertRange
   ENDIF
   ```

   where $R_*(t)$ is an abbreviation for $R(\texttt{measure2state}(i,0), \texttt{measure2state}(e,0), t)$, which is the R function (i.e., Equation 14) evaluated on the measured state variables at time t.

   (a) Case $\tau(0) > \texttt{AlertTime}$. We need to establish that

   $$R_*(\texttt{AlertTime}) \le \texttt{AlertRange}.$$

From Lemma asymptotic_decrease_tau, we have:

$$R_*(T) \geq R_*(\texttt{AlertTime}).$$

From Lemma alarm_NOT_Omega_T we have :

$$\begin{aligned}\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \neg\texttt{Omega}(\beta + \theta_0) \ \wedge \ \texttt{conflict}_{ie}(T) \\ \supset \ R_*(T) \leq \texttt{AlertRange}\end{aligned}$$

and we immediately get the desired result by transitivity.

(b) Case $\tau(0) \leq \texttt{AlertTime}$. We need to establish that

$$R_*(\tau(0)) \leq \texttt{AlertRange}.$$

From Lemma alarm_NOT_Omega_tau, we have

$$\begin{aligned}\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \neg\texttt{Omega}(\theta_0 + \beta) \ \wedge \\ \texttt{conflict}_{ie}(T) \ \wedge \ \tau(0) > 0 \\ \supset \ R_*(\tau(0)) \leq \texttt{AlertRange}\end{aligned}$$

which discharges this case.

2. Case $\rho(\texttt{bank}(i)) \neq 0$. Expanding ails_alert and arc_loop, and using the fact that $\texttt{mod}(0, m) = 0$, for $m \neq 0$, we end up with an identical goal to the result proved in the previous case. This means that only one tangential projection is necessary to issue an alarm.

□

The following lemmas are more general than the other lemmas in the appendix in that they only involve standard mathematical functions and not the specific functions of the collision avoidance framework[10]. It is noteworthy that they were discovered with the aid of a plotting tool (GNUPLOT) and a computer algebra program named MuPAD. At first these were introduced into the PVS theories as axioms. After all of the main theorems of this paper were completed, proofs of these lemmas were constructed in PVS. Whether this last step is necessary is a philosophical one. Nevertheless, this two step process was essential to the discovery of several of the proofs in this paper.

[10]Although the lemmas reference terms such as MinDistance and ConflictRange, these are just constants that can be replaced by their values.

Lemma 33 (Math_prop_no_conflict_1).

$$\begin{gathered} v = 250 \ \wedge \ 9.5 \leq T \leq 10 \ \wedge \\ \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \texttt{MinBeta} \leq a \leq \pi/2 \ \wedge \\ y \geq l\sin(a) + r[\cos(\rho T) - 1] \ \wedge \ x \geq r\sin(\rho T) - l\cos(a) \\ \supset \\ x^2 + y^2 > \texttt{ConflictRange}^2, \end{gathered}$$

where $\texttt{MinBeta} = 539/1000$.

Proof. The key to proving this theorem was finding the minimum of

$$r\sin(\rho T) - l\cos(\texttt{MinBeta})]^2 + [r\left(\cos(\rho T) - 1\right) + l\sin(\texttt{MinBeta})]^2 \tag{33}$$

and splitting the proof into the two cases for each side of this minimum. The mimimum occurred around $L = 2442$ as illustrated in Figure 9.

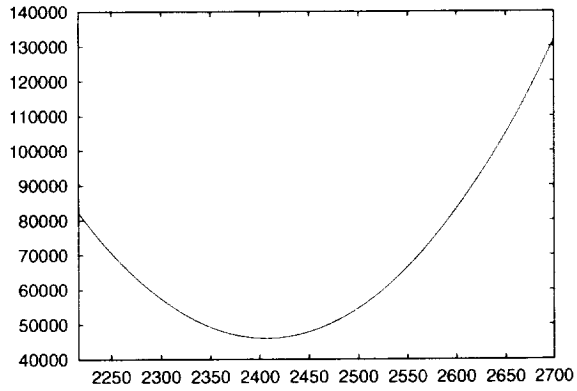


Figure 9: Plot of formula 33 as a function of l.

1. Case $l \leq L$. Because sin is monotonically increasing over the the range $[0, \pi/2]$ and $\texttt{MinBeta} \leq a$, we have:

   $$l\sin(a) + r[\cos(\rho T) - 1] \geq l\sin(\texttt{MinBeta}) + r[\cos(\rho T) - 1].$$

   Applying transitivity to this formula and the y premise of the theorem, we have

   $$y \geq l\sin(\texttt{MinBeta}) + r[\cos(\rho T) - 1].$$

   Squaring both sides:

   $$y^2 \geq [r\left(\cos(\rho T) - 1\right) + l\sin(\texttt{MinBeta})]^2. \tag{34}$$

Because cos is monotonically decreasing over the range $[0, \pi/2]$ and $\texttt{MinBeta} \leq a$, we have:

$$r\sin(\rho T) - l\cos(a) \geq r\sin(\rho T) - l\cos(\texttt{MinBeta}).$$

Applying transitivity to this formula and the x premise of the theorem, we have

$$x \geq r\sin(\rho T) - l\cos(\texttt{MinBeta}).$$

Squaring both sides:

$$x^2 \geq [r\sin(\rho T) - l\cos(\texttt{MinBeta})]^2. \tag{35}$$

Combining formulas 35 and 34 yields:

$$x^2 + y^2 \geq [r\sin(\rho T) - l\cos(\texttt{MinBeta})]^2 + [r\,(\cos(\rho T) - 1) + l\sin(\texttt{MinBeta})]^2.$$

Expanding the squares:

$$\begin{aligned} x^2 + y^2 \;\geq\; & r^2\sin^2(\rho T) + l^2\cos^2(\texttt{MinBeta}) - 2lr\sin(\rho T)\cos(\texttt{MinBeta}) \\ & + r^2\,(\cos(\rho T) - 1)^2 + l^2\sin^2(\texttt{MinBeta}) \\ & + 2lr\,(\cos(\rho T) - 1)\sin(\texttt{MinBeta}). \end{aligned}$$

Using $\sin^2(\alpha) + \cos^2(\alpha) = 1$, we have

$$\begin{aligned} x^2 + y^2 \;\geq\; & r^2\sin^2(\rho T) + l^2 - 2lr\sin(\rho T)\cos(\texttt{MinBeta}) \\ & + r^2\,(\cos(\rho T) - 1)^2 + 2lr\,[\cos(\rho T) - 1]\sin(\texttt{MinBeta}). \end{aligned}$$

Further expansion and simplification yields:

$$\begin{aligned} x^2 + y^2 \;\geq\; & l^2 - 2lr\sin(\rho T)\cos(\texttt{MinBeta}) \\ & + r^2[\sin^2(\rho T) + \cos^2(\rho T) - 2\,\cos(\rho T) + 1] + 2lr\,[\cos(\rho T) - 1]\sin(\texttt{MinBeta}). \end{aligned}$$

Using $\sin^2(\alpha) + \cos^2(\alpha) = 1$ again, we have

$$\begin{aligned} x^2 + y^2 \;\geq\; & l^2 - 2lr\sin(\rho T)\cos(\texttt{MinBeta}) \\ & + 2r^2[1 - \cos(\rho T)] + 2lr\,[\cos(\rho T) - 1]\sin(\texttt{MinBeta}). \end{aligned}$$

Rearranging terms and simplifying:

$$\begin{aligned} x^2 + y^2 \;\geq\; & l^2 + 2r^2[1 - \cos(\rho T)] \\ & + 2lr[(\cos(\rho T) - 1)\sin(\texttt{MinBeta}) - \sin(\rho T)\cos(\texttt{MinBeta})]]. \end{aligned}$$

Further manipulation yields:

$$\begin{aligned} x^2 + y^2 \;\geq\; & l^2 + 2r^2[1 - \cos(\rho T)] - 2lr\sin(\texttt{MinBeta}) \\ & + 2lr[\cos(\rho T)\sin(\texttt{MinBeta}) - \sin(\rho T)\cos(\texttt{MinBeta})]]. \end{aligned}$$

Using the difference of two angles trigonmetric identity for sin yields:

$$\begin{aligned} x^2 + y^2 \;\; &\geq \;\; l^2 + 2r^2[1 - \cos(\rho T)] - 2lr\sin(\texttt{MinBeta}) \\ &\quad +2lr\sin(\texttt{MinBeta} - \rho T). \end{aligned}$$

Using $\sin(-\alpha) = -\sin(\alpha)$, we have:

$$\begin{aligned} x^2 + y^2 \;\; &\geq \;\; l^2 + 2r^2[1 - \cos(\rho T)] - 2lr\sin(\texttt{MinBeta}) \\ &\quad -2lr\sin(\rho T - \texttt{MinBeta}). \end{aligned}$$

Rearranging terms and simplifying:

$$\begin{aligned} x^2 + y^2 \;\; &\geq \;\; l^2 + 2r^2 - 2r^2\cos(\rho T) - 2lr\sin(\texttt{MinBeta}) \\ &\quad -2lr\sin(\rho T - \texttt{MinBeta}). \end{aligned}$$

Now, axiom `Ax2`, which has been checked in MuPAD, yields:

$$\begin{aligned} &v = 250 \;\wedge\; 9.5 \leq T \leq 10 \;\wedge \\ &\quad \texttt{MinDistance} \leq l \leq L \qquad \supset \quad l^2 + 2\,r^2 - \texttt{ConflictRange}^2 - 2r^2\cos(\rho T) \\ &\qquad\qquad\qquad\qquad\qquad\qquad -2lr\sin(\texttt{MinBeta}) - 2lr\sin(\rho T - \texttt{MinBeta}) > 0. \end{aligned}$$

Rearranging terms of this axiom gives us:

$$l^2 + 2\,r^2 - 2r^2\cos(\rho T) - 2lr\sin(\texttt{MinBeta}) - 2lr\sin(\rho T - \texttt{MinBeta}) > \texttt{ConflictRange}^2.$$

Transitivity yields

$$x^2 + y^2 \;\; \geq \;\; \texttt{ConflictRange}^2$$

the desired result.

2. Case $l > L$. Using the techniques described in Section 3, the lemma `Math_prop_no_conflict_y_L_PI2`:

$$\begin{gathered} v = 250 \;\wedge\; 9.5 \leq T \leq 10 \;\wedge \\ L \leq l \leq \texttt{MaxDistance} \;\wedge \\ \texttt{MinBeta} \leq a \leq \pi/2 \;\wedge \\ y \geq l\sin(a) + r[\cos(\rho T) - 1] \\ \supset \\ y > \texttt{ConflictRange} \end{gathered}$$

is easily established. The premises of this lemma follow from the premises of the theorem, so we have

$$y > \texttt{ConflictRange}.$$

Squaring both sides yields

$$y^2 > \texttt{ConflictRange}^2.$$

From which the desired result:

$$x^2 + y^2 > \texttt{ConflictRange}^2$$

immediately follows.

□

Lemma 34 (Math_prop_no_conflict_2).

$$\begin{array}{c} v = 250 \ \wedge \ 9.5 \leq T \leq 10 \ \wedge \\ \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ 3\pi/2 \leq a \leq 2\pi - \texttt{MinBeta} \ \wedge \\ y \geq l\sin(a) + r(\cos(\rho T) - 1) \ \wedge \ x \geq r\sin(\rho T) - l\cos(a) \\ \supset \\ x^2 + y^2 > \texttt{ConflictRange}^2. \end{array}$$

Proof. Using lemma Math_prop_no_conflict_1, substituting $2\pi - a$ for a and $-y$ for y yields:

$$\begin{array}{c} v = 250 \ \wedge \ 9.5 \leq T \leq 10 \ \wedge \\ \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \texttt{MinBeta} \leq 2\pi - a \leq \pi/2 \ \wedge \\ -y \geq l\sin(2\pi - a) + r[\cos(\rho T) - 1] \ \wedge \\ x \geq r\sin(\rho T) - l\cos(2\pi - a) \\ \supset \\ x^2 + (-y)^2 > \texttt{ConflictRange}^2. \end{array}$$

Since $(-y)^2 = y^2$, $\sin(2\pi - a) = -sin(a)$ and $\cos(2\pi - a) = \cos(a)$, we have

$$\begin{array}{c} v = 250 \ \wedge \ 9.5 \leq T \leq 10 \ \wedge \\ \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ \texttt{MinBeta} \leq 2\pi - a \leq \pi/2 \ \wedge \\ -y \geq -l\sin(a) + r[\cos(\rho T) - 1] \ \wedge \\ x \geq r\sin(\rho T) - l\cos(a) \\ \supset \\ x^2 + y^2 > \texttt{ConflictRange}^2. \end{array}$$

Multiplying both sides of the y premise by -1 and writing $\texttt{MinBeta} \leq 2\pi - a \leq \pi/2$ as $3\pi/2 \leq a \leq 2\pi - \texttt{MinBeta}$ yield the desired result. □

Lemma 35 (Math_prop_alarm_1).

$$\begin{array}{c} \texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ 0 \leq a \ \wedge \ a \leq \texttt{MinBeta} \\ \supset \ [l\cos(a) - vT]^2 + [l\sin(a)]^2 \leq \texttt{AlertRange}^2. \end{array}$$

Proof. Using algebraic manipulation we get

$$[l\cos(a) - vT]^2 + [l\sin(a)]^2 \ = \ v^2T^2 + l^2 - 2vTl\cos(a).$$

Using the techniques described in Section 3, we get

$$v^2T^2 + l^2 - 2vTl\cos(a) \ \leq \ v^2T^2 + l^2 - 2vTl\cos(\texttt{MinBeta}).$$

Finally, we have checked in MuPAD, and assumed it as an axiom in PVS (Axiom **Ax3**), that

$$v^2T^2 + l^2 - 2vTl\cos(\texttt{MinBeta}) \leq \texttt{AlertRange}^2$$

under the given hypothesis. Transitivity yields the result. □

Lemma 36 (Math_prop_alarm_2).

$$\begin{aligned}\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge\ 2\pi - \texttt{MinBeta} \leq a \ \wedge\ a \leq 2\pi \\ \supset\ [l\cos(a) - vT]^2 + [l\sin(a)]^2 \leq \texttt{AlertRange}^2.\end{aligned}$$

Proof. Using lemma **Math_prop_alarm_1**, substituting $2\pi - a$ for a yields:

$$\begin{aligned}\texttt{MinDistance} \leq l \leq \texttt{MaxDistance} \ \wedge \\ 0 \leq 2\pi - a \ \wedge\ 2\pi - a \leq \texttt{MinBeta} \\ \supset\ [l\cos(2\pi - a) - vT]^2 + [l\sin(2\pi - a)]^2 \leq \texttt{AlertRange}^2.\end{aligned}$$

We conclude using the equalities $\cos(2\pi - a) = \cos(a)$, $\sin(2\pi - a) = -\sin(a)$, and $[-l\sin(a)]^2 = [l\sin(a)]^2$. □

Appendix B: Table of Translations

Paper	PVS
i	intr
e	evad
θ_t	theta(t)
θ_0	theta(0)
ρ	rho(v)
β	beta
ϕ	phi
$R(t)$	R(intruder,evader,t)
$\tau(t)$	tau(intruder,evader,t)
x_i	x(intruder)
x_e	x(evader)
$\hat{x}(t)$	xp(t)
$\hat{y}(t)$	yp(t)
$\hat{x}'(t)$	D(xp(t))
$\hat{y}'(t)$	D(yp(t))
$x*$	xtrk
$y*$	ytrk
$\sin_{lb}$	sin_lb
$\cos_{lb}$	cos_lb
$\sin_{ub}$	sin_ub
$\cos_{ub}$	cos_ub
r_{ub}	r_ub(V)
ρ_{ub}	rho_ub(V)
r_{lb}	r_lb(V)
ρ_{lb}	rho_lb(V)
$D_{ie}(t_i, t_e)$	Die(ti,te)

Appendix C: AILS Alerting Algorithm in PVS

```
ails: THEORY
 BEGIN

  Bank: TYPE = {r: real | −MaxBank ≤ r ≤ MaxBank}

  State: TYPE = [# x: real, y: real, heading: real, bank: Bank #]

  i, e: VAR State

  range, t: VAR real

  φ: VAR Bank

  r, ρ: VAR real

  k: VAR [0... MaxStep]

  idtrk: VAR posnat

  ρ(φ): real = g tan(φ)/v

  chkrange(range, t): bool = range ≤ AlertRange ∧ t ≤ AlertTime

  chktrack(i, e, t): bool =
      LET τ = τ(i, e, 0) IN
        IF τ ≤ 0
          THEN chkrange(R(i, e, 0), t)
        ELSIF t + τ > AlertTime
              THEN R(i, e, AlertTime) ≤ AlertRange
              ELSE R(i, e, τ) ≤ AlertRange
        ENDIF

  arc_loop(i, e, r, ρ, idtrk, k): RECURSIVE bool =
    IF k = MaxStep
      THEN FALSE
    ELSE LET t = k 1/2 IN
           LET xloc = x(e) + v t IN
             LET yloc = y(e) IN
               LET (x*, y*)
                     = IF ρ > 0
```

```
                    THEN (x(i) + r (sin(heading(i) + ρt) − sin(heading(i))),
                          y(i) + r (cos(heading(i)) − cos(heading(i) + ρt)))
                  ELSE (x(i) + r (sin(heading(i)) − sin(heading(i)) + ρt),
                        y(i) + r (cos(heading(i) + ρt) − cos(heading(i))))
                  ENDIF
          IN
          IF ¬ mod(k, idtrk) = 0
            THEN LET range = √((x* − xloc)² + (y* − yloc²)) IN
                   IF chkrange(range, t)
                     THEN TRUE
                   ELSE arc_loop(i, e, r, ρ, idtrk, k + 1)
                   ENDIF
          ELSE LET tantrk = heading(i) + tρ IN
                 LET int = i WITH [x := x*, y := y*, heading := tantrk] IN
                   LET eva = e WITH [x := xloc, y := yloc] IN
                     IF chktrack(int, eva, t)
                       THEN TRUE
                     ELSE arc_loop(i, e, r, ρ, idtrk, k + 1)
                     ENDIF
          ENDIF
  ENDIF
   MEASURE (MaxStep − k)

ails_alert(i, e): bool =
    LET φ = bank(i) IN
      LET ρ = ρ(φ) IN
        IF ρ = 0
          THEN chktrack(i, e, 0)
        ELSE LET r = v²/(g tan(φ)) IN
               LET idtrk
                 = IF ρ ≥ 3
                     THEN 1
                   ELSIF ρ ≥ 1 + 1/2
                         THEN 2
                         ELSE IF ρ ≥ 3/4 THEN 4 ELSE 8 ENDIF
                   ENDIF
                IN arc_loop(i, e, r, ρ, idtrk, 0)
        ENDIF

END ails
```

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	May 2001	Technical Memorandum

4. TITLE AND SUBTITLE
On the Formal Verification of Conflict Detection Algorithms

5. FUNDING NUMBERS
727-01-22-01

6. AUTHOR(S)
César Muñoz, Ricky W. Butler, Víctor A. Carreño, and Gilles Dowek

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
NASA Langley Research Center
Hampton, VA 23681-2199

8. PERFORMING ORGANIZATION REPORT NUMBER
L-18083

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA/TM-2001-210864

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified-Unlimited
Subject Category 61 Distribution: Standard
Availability: NASA CASI (301) 621-0390

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)
Safety assessment of new air traffic management systems is a main issue for civil aviation authorities. Standard techniques such as testing and simulation have serious limitations in new systems that are significantly more autonomous than the older ones. In this paper, we present an innovative approach, based on formal verification, for establishing the correctness of conflict detection systems. Fundamental to our approach is the concept of trajectory, which is a continuous path in the x-y plane constrained by physical laws and operational requirements. From the Model of trajectories, we extract, and formally prove, high level properties that can serve as a framework to analyze conflict scenarios. We use the AILS alerting algorithm as a case study of our approach.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Trajectory modeling; Conflict detection; Collision alerting; Formal methods; Theorem proving	57
	16. PRICE CODE A04

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL