Formal Verification of Air Traffic Conflict Prevention Bands Algorithms

Anthony J. Narkawicz and César A. Muñoz
Langley Research Center, Hampton, Virginia

Gilles Dowek
Ecole polytechnique, France

June 2010
NASA STI Program . . . in Profile

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

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

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.

- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

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

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

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

Specialized services also include creating custom thesauri, building customized databases, and organizing and publishing research results.

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

- Access the NASA STI program home page at [http://www.sti.nasa.gov](http://www.sti.nasa.gov)

- E-mail your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA STI Help Desk at 443-757-5803

- Phone the NASA STI Help Desk at 443-757-5802

- Write to:
  NASA STI Help Desk
  NASA Center for AeroSpace Information
  7115 Standard Drive
  Hanover, MD 21076-1320
Formal Verification of Air Traffic Conflict Prevention Bands Algorithms

Anthony J. Narkawicz and César A. Muñoz
Langley Research Center, Hampton, Virginia

Gilles Dowek
Ecole polytechnique, France

National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia 23681-2199

June 2010
The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.
Abstract

In air traffic management, a pairwise conflict is a predicted loss of separation between two aircraft, referred to as the ownship and the intruder. A conflict prevention bands system computes ranges of maneuvers for the ownship that characterize regions in the airspace that are either conflict-free or “don’t go” zones that the ownship has to avoid. Conflict prevention bands are surprisingly difficult to define and analyze. Errors in the calculation of prevention bands may result in incorrect separation assurance information being displayed to pilots or air traffic controllers. This paper presents provably correct 3-dimensional prevention bands algorithms for ranges of track angle, ground speed, and vertical speed maneuvers. The algorithms have been mechanically verified in the Prototype Verification System (PVS). The verification presented in this paper extends in a non-trivial way that of previously published 2-dimensional algorithms.
## Contents

1 Introduction ........................................ 1

2 Statement of the Problem ............................ 3
   2.1 Conflicts .................................... 3
   2.2 Track Angle, Ground Speed, and Vertical Speed Maneuvers .... 4
   2.3 Conflict Detection Algorithm ....................... 5
   2.4 Prevention Bands Algorithms ..................... 6
   2.5 Proving Correctness of a Prevention Bands Algorithm ........... 7

3 The Function \( \Omega \) ................................. 8
   3.1 Cylindrical Distance ............................ 9
   3.2 The Definition of \( \Omega \) ......................... 9
   3.3 Continuity of \( \Omega \) ............................. 11

4 Classification of Critical Vectors .................. 11
   4.1 Vertical Case .................................. 14
   4.2 Circle Case 2D .................................. 14
   4.3 Circle Case 3D .................................. 14
   4.4 Line Case ...................................... 15
   4.5 The Classification Theorem ....................... 16
   4.6 Entry and Exit Times ............................ 16

5 Track Angle Prevention Bands ....................... 18
   5.1 A Special Version of \( \Omega_{\text{strk}} \) .................. 19
   5.2 Line Solutions For Track Angle Maneuvers ............... 21
   5.3 2D Circle Solutions For Track Angle Maneuvers ............ 23
   5.4 3D Circle Solutions For Track Angle Maneuvers ............ 25
   5.5 A Prevention Bands Algorithm For Track Angle Maneuvers ...... 26

6 Ground Speed Prevention Bands ..................... 29
   6.1 Line Solutions For Ground Speed Maneuvers ............... 30
   6.2 2D Circle Solutions For Ground Speed Maneuvers ............ 32
   6.3 3D Circle Solutions For Ground Speed Maneuvers ............ 33
   6.4 A Prevention Bands Algorithm For Ground Speed Maneuvers ...... 34

7 Vertical Speed Prevention Bands .................... 36
   7.1 3D Circle and Vertical Solutions For Vertical Speed Maneuvers .... 36
   7.2 A Prevention Bands Algorithm For Vertical Speed Maneuvers ...... 39

8 Conclusion ........................................... 40
1 Introduction

In air traffic management, a (pairwise) conflict is a predicted loss of separation between two aircraft within a lookahead time. One of the aircraft is called the ownship and the other aircraft, which represents an arbitrary traffic aircraft, is called the intruder.

A conflict prevention system consists of algorithms that sense traffic aircraft and characterize ranges of maneuvers for the ownship that are either conflict-free or that lead to conflict. The maneuvers are typically constrained to those where only one parameter of the ownship’s velocity is varied at a time: track angle, vertical speed, or ground speed.

More precisely, a (pairwise) prevention bands algorithm, for a given parameter such as track angle, ground speed, or vertical speed, has as input the state information of the ownship and intruder aircraft, i.e., their 3-dimensional position and velocity vectors. It returns a list of regions, called bands, consisting of values for the specified parameter. There is a natural way to associate a color, either red or green, to each band. Red bands specify “don’t go” zones, i.e., parameter values that the ownship has to avoid because they lead to conflict. Conversely, the green bands specify parameter values for the ownship that yield conflict-free maneuvers.

Figure 1 illustrates in a graphical display prevention bands for the ownship for track angle, ground speed, and vertical speed maneuvers. Given the current position and velocity vectors of the aircraft, the displayed bands in Figure 1 indicate that the aircraft will be in conflict if, for instance, the ownship maneuvers to a track angle of 45°, to a ground speed of 300 knots, or to a vertical speed of 0 feet per min. On the other hand, if the ownship maneuvers to any value in the green regions the aircraft will be conflict-free.

A pairwise prevention bands algorithm is correct if every possible value for the chosen parameter is either contained in a band or is a boundary point of one of the bands, and if the colors of the bands characterize conflict as follows. For all bands $B$ and parameter values $x \in B$, the ownship’s maneuver corresponding to the value $x$
is in conflict with the traffic aircraft if and only if the color of B is red. Equivalently, the ownship's maneuver corresponding to x is not in conflict if and only if the color of B is green.

Conflict prevention bands are surprisingly difficult to define and analyze [1]. The formal verification of a prevention bands algorithm for horizontal conflicts was described in [2]. Three-dimensional prevention bands algorithms were presented, without correctness proofs, in [3]. The 3-dimensional algorithms presented in that paper compute incorrect bands for some special cases. This paper presents correct versions of the prevention bands algorithms originally proposed in [3]. The correctness properties of these new algorithms have been formally verified in the Prototype Verification Systems (PVS) [4].

This paper focuses on pairwise algorithms, i.e., it considers only one traffic aircraft: the intruder. Prevention bands algorithms for an arbitrary number of traffic aircraft can be obtained from a pairwise algorithm by simply letting the red region for n-aircraft be the union of the red regions computed for the ownship and each individual traffic aircraft. The green regions can be computed as the complement of the red ones. The correctness of the algorithms for n-aircraft can be easily derived from the correctness of the pairwise prevention bands algorithms.

**Notation**

The mathematical development presented in this paper has been fully formalized in PVS.\(^1\) However, for readability, this paper uses standard mathematical notation instead of PVS syntax.

Vector variables are written in **boldface** letters and can denoted by their components. For example, if \( w \in \mathbb{R}^3 \) and \( u \in \mathbb{R}^2 \), then \( w = (w_x, w_y, w_z) \) and \( u = (u_x, u_y) \). The notation \( w_{(x,y)} \) denotes the projection of \( w \) in the horizontal plane, i.e.,\(^2\)

\[
w_{(x,y)} \equiv (w_x, w_y),
\]

and the notation \( u \text{ with } [z \leftarrow r] \) denotes the 3-dimensional vector whose projection to \( \mathbb{R}^2 \) is \( u \) and whose z-coefficient is \( r \in \mathbb{R} \), i.e.,

\[u \text{ with } [z \leftarrow r] \equiv (u_x, u_y, r).
\]

As usual, the notation \( \|w\| \) refers to the norm of the vector \( w \) and the notation \( w \cdot w' \) refers to the dot product of the vectors \( w \) and \( w' \). The expression \( 0 \) represents the zero vector, e.g., the vector whose components are 0.

If \( u \in \mathbb{R}^2 \), then \( u^\perp \) denotes the (right) perpendicular vector:

\[u^\perp \equiv (u_y, -u_x).
\]

From this definition, it can be easily proven that \( u \cdot u^\perp = 0 \). Furthermore, if \( u \) is nonzero, then the vector \( w \in \mathbb{R}^2 \) can be written as a linear combination of \( u \) and \( u^\perp \) in the following way:

\[
w = \frac{1}{\|u\|^2} ((u \cdot w) u + (u^\perp \cdot w) u^\perp).
\] \(^{(1)}\)

\(^1\)Electronically available from http://shemesh.larc.nasa.gov/people/cam/ACCoRD.

\(^2\)The symbol \( \equiv \) is used in this paper to introduce mathematical definitions.
The function sign: \( \mathbb{R} \to \{-1, 1\} \) is defined such that \( \text{sign}(x) = 1 \) if \( x \geq 0 \) and \( \text{sign}(x) = -1 \) otherwise. As usual in mathematics, \( \iota = \pm 1 \) denotes the fact that an integer \( \iota \) belongs to the set \( \{-1, 1\} \). Moreover, \( \neg \), \( \implies \), \( \iff \) denote logical negation, implication, and equivalence, respectively.

Finally, by convention, names of predicates and functions used in the specification of the problem are written in italics. Functions that represent algorithms to be implemented in a programming language are written in typewriter font.

## 2 Statement of the Problem

The prevention bands algorithms discussed here only use state-based information for the two aircraft, i.e., constant position and velocity vectors that are elements of the 3-dimensional Euclidean space \( \mathbb{R}^3 \). Aircraft dynamics are represented by a point moving at constant linear speed. These approximations of real aircraft behavior are valid for short lookahead times (typically less than 5 minutes). The current state of the ownship and traffic aircraft are denoted by the following vectors.

| \( \mathbf{s}_o \in \mathbb{R}^3 \) | Initial position of the ownship aircraft |
| \( \mathbf{v}_o \in \mathbb{R}^3 \) | Initial velocity of the ownship aircraft |
| \( \mathbf{s}_i \in \mathbb{R}^3 \) | Initial position of the traffic aircraft |
| \( \mathbf{v}_i \in \mathbb{R}^3 \) | Initial velocity of the traffic aircraft |

In the airspace system, the separation criterion for two aircraft is specified as a minimum horizontal separation \( D \) and a minimum vertical separation \( H \). A conflict between the ownship and the intruder occurs when there is a time in the future, within a lookahead time \( T \), such that the horizontal distance between the aircraft is less than \( D \), and the vertical distance is less than \( H \). Typically, \( D \) is 5 nautical miles, \( H \) is 1000 feet, and \( T \) is 5 minutes.

For the remainder of the paper, it is assumed that the ground speeds of the ownship and intruder aircraft are not zero, i.e., both \( \| \mathbf{v}_o(x,y) \| \neq 0 \) and \( \| \mathbf{v}_i(x,y) \| \neq 0 \) hold, and that the aircraft are not in loss of separation, i.e., either \( \| \mathbf{s}_o(x,y) - \mathbf{s}_i(x,y) \| \geq D \) or \( |\mathbf{s}_{oz} - \mathbf{s}_{iz}| \geq H \) hold. Therefore,

\[
\begin{align*}
\mathbf{v}_o(x,y) &\neq \mathbf{0}, \\
\mathbf{v}_i(x,y) &\neq \mathbf{0}, \\
\mathbf{s}_o - \mathbf{s}_i &\neq \mathbf{0}.
\end{align*}
\]

As noted in the introduction, the possible maneuvers considered for the ownship are constrained to those where only one parameter of the ownship’s velocity vector is varied, e.g., track angle, ground speed, or vertical speed.

### 2.1 Conflicts

The ownship and the intruder aircraft are in conflict if there exists \( t \in [0,T] \) such that, at time \( t \), vertical separation is lost, i.e,

\[
|((\mathbf{s}_o + t \mathbf{v}_o) - (\mathbf{s}_i + t \mathbf{v}_i))_z| < H,
\]
and horizontal separation is lost, i.e.,

\[ \|(s_0 + t v_o)(x,y) - (s_i + t v_i)(x,y)\| < D. \]

Since \((s_0 + t v_o) - (s_i + t v_i) = (s_0 - s_i) + t(v_o - v_i)\), the predicate that characterizes conflict can be defined on \(s = s_0 - s_i\) and \(v = v_o - v_i\), the relative position and velocity vector, respectively, of the ownship with respect to the intruder.

That is, conflict can be viewed as a predicate of two vectors \(s\) and \(v\) rather than a predicate of four vectors \(s_0, v_o, s_i,\) and \(v_i\), a result that greatly simplifies the notation. Thus, the predicate \(\text{conflict}\) can be formally defined as follows.

\[
\text{conflict?}(s, v) \equiv \exists t \in [0, T] : ||(s + t v)\| < H \text{ and } \|s(x,y) + tv(x,y)\| < D.
\]

For the remainder of this paper, the relative position and velocity vectors, \(s\) and \(v\), will be used in place of \(s_0 - s_i\) and \(v_o - v_i\), respectively.

The separation criterion can be understood as an imaginary cylinder of height \(H\) and diameter \(D\) around each aircraft and a conflict between two aircraft as a future overlapping of these cylinders. In this paper, an alternative but equivalent view is considered where the intruder is surrounded by a cylinder, called \(\text{protected zone}\), of half-height \(H\) and radius \(D\). From this perspective, a conflict between these two aircraft is equivalent to the existence of a time \(t \in [0, T]\) at which the ownship is in the interior of the intruder’s protected zone.

### 2.2 Track Angle, Ground Speed, and Vertical Speed Maneuvers

A \textit{maneuver} for the ownship is a new velocity vector \(v'_o\) that is implemented by the aircraft in zero time. Track angle, ground speed, and vertical speed maneuvers are formally defined as follows.

- A \textit{track angle maneuver} for the ownship is a velocity vector \(v'_o\) such that \(\|v'_o(x,y)\| = \|v_o(x,y)\|\) and \(v'_{o z} = v_{o z}\). In this case, there exists a function \(\text{track} : \mathbb{R}^3 \rightarrow \mathbb{R}\) that computes a real number \(\alpha = \text{track}(v'_o)\), called the \textit{track angle} of \(v'_o\), such that
  \[
  v'_o(x,y) = (\|v_o(x,y)\| \sin \alpha, \|v_o(x,y)\| \cos \alpha).
  \]

The function \(\text{track}\) is easily defined using the arc tangent function and the signs of \(v'_{ox}\) and \(v'_{oy}\).

- A \textit{ground speed maneuver} for the ownship is a velocity vector \(v'_o\) such that \(v'_o(x,y)\) and \(v_o(x,y)\) are parallel (have the same track angle) and \(v'_{o z} = v_{o z}\). In this case, there exists a real number \(p\) with the property that
  \[
  v'_o = (\frac{p}{\|v_o(x,y)\|} v_{ox}, \frac{p}{\|v_o(x,y)\|} v_{oy}, v_{o z}).
  \]

The number \(p\) is the \textit{ground speed} of \(v_o\), i.e., \(\|v'_o(x,y)\| = p\).
• A *vertical speed maneuver* for the ownship is a velocity vector \( \mathbf{v}_o' \) such that \( \mathbf{v}_o'(x,y) = \mathbf{v}_o(x,y) \), i.e., the horizontal velocity vectors are equal. In this case, there exists a real number \( r \), called the *vertical speed* of \( \mathbf{v}_o' \) such that

\[
\mathbf{v}_o' = (v_{ox}, v_{oy}, r).
\]

The functions \( \nu_{trk}, \nu_{gs}, \nu_{vs} : \mathbb{R} \rightarrow \mathbb{R}^3 \), implicitly parametrized by \( \mathbf{v}_o \), are defined as follows.

\[
\nu_{trk}(\alpha) = (\|\mathbf{v}_o(x,y)\| \sin \alpha, \|\mathbf{v}_o(x,y)\| \cos \alpha, v_{oz}),
\]

\[
\nu_{gs}(p) = \left( \frac{p}{\|\mathbf{v}_o(x,y)\|} v_{ox}, \frac{p}{\|\mathbf{v}_o(x,y)\|} v_{oy}, v_{oz} \right),
\]

\[
\nu_{vs}(r) = (v_{ox}, v_{oy}, r),
\]

These functions assign to each track angle \( \alpha \in \mathbb{R} \), ground speed \( p \in \mathbb{R} \), and vertical speed \( r \in \mathbb{R} \), respectively, the corresponding velocity vector for the ownship. Important properties of the functions \( \nu_{trk}, \nu_{gs}, \) and \( \nu_{vs} \) are:

\[
\|\nu_{trk}(\alpha)(x,y)\| = \|\mathbf{v}_o(x,y)\|,
\]

\[
\|\nu_{gs}(p)(x,y)\| = p,
\]

\[
\nu_{vs}(r)_z = r.
\]

The constructions in this paper will restrict ground speed maneuvers to those where the ground speed \( p \) is positive.

### 2.3 Conflict Detection Algorithm

A *conflict detection* algorithm \( cd \) is a function that takes as parameters the relative position of the aircraft \( s \) and the velocity vectors \( \mathbf{v}_o, \mathbf{v}_i \), and returns a Boolean value, i.e., True or False.

**Definition 1.** The algorithm \( cd \) is **correct** if it holds that

\[
\text{conflict?}(s, \mathbf{v}_o - \mathbf{v}_i) \implies cd(s, \mathbf{v}_o, \mathbf{v}_i).
\]

It is **complete** if it holds that

\[
cd(s, \mathbf{v}_o, \mathbf{v}_i) \implies \text{conflict?}(s, \mathbf{v}_o - \mathbf{v}_i).
\]

In other words, a conflict detection algorithm is correct if it does not have missed alerts, i.e., it detects all conflicts, and it is complete if it does not have false alerts, i.e., it only detects actual conflicts. Note that a conflict detection algorithm \( cd_T \) that always returns True is correct and an algorithm \( cd_F \) that always returns False is complete. However, \( cd_T \) is not complete and \( cd_F \) is not correct. An example of a correct *and* complete conflict detection algorithm is \( cd3d \) (see Appendix in [3]).
2.4 Prevention Bands Algorithms

Given a function \( \nu : \mathbb{R} \rightarrow \mathbb{R}^3 \) and a closed interval \( I = [I_1, I_2] \), a prevention bands algorithm for \( \nu \) over \( I \) is a function with parameters \( s, \nu_{o}, \) and \( \nu_{i} \) that returns a finite, ordered sequence \( L_\nu \) of elements of \( I \), such that \( I_1 \in L_\nu \) and \( I_2 \in L_\nu \). Each consecutive pair \( A \) and \( B \) of entries in \( L_\nu \) determines an open interval \( (A, B) \), which is called a band (for the parameter represented by \( \nu \)).

By abuse of notation, the syntax \((A, B) \in L_\nu \) will denote that \((A, B)\) is a band in \( L_\nu \), i.e., \( A \) and \( B \) are consecutive entries in \( L_\nu \).

**Definition 2.** Given a function \( \nu : \mathbb{R} \rightarrow \mathbb{R}^3 \) and a closed interval \( I \subset \mathbb{R} \), a prevention bands algorithm for \( \nu \) over \( I \) is correct if for any band \((A, B)\) in \( L_\nu \), and real numbers \( x, y \in (A, B) \), it holds that

\[
\text{conflict}?(s, \nu(x) - \nu_{i}) \leftrightarrow \text{conflict}?(s, \nu(y) - \nu_{i}).
\]

The definition above states that all the points in a band computed by a correct prevention bands algorithm have the same conflict property, e.g., either all the points yield conflict-free maneuvers or all the points yield maneuvers that lead to conflict. Typically, \( \nu \) will be one of the functions \( \nu_{\text{trk}}, \nu_{\text{gs}}, \) or \( \nu_{\text{vs}} \) defined in formulas (3), (4), and (5). The boundaries \( I_1 \) and \( I_2 \), of the interval \( I \), are minimum and maximum values for the argument of \( \nu \). For \( \nu = \nu_{\text{trk}} \), the standard values are \( I_1 = 0 \) and \( I_2 = 2\pi \). For \( \nu = \nu_{\text{gs}} \) and \( \nu = \nu_{\text{vs}} \), \( I_1 \) and \( I_2 \) are typically the minimum and maximum ground or vertical speeds for the ownship, respectively.

To each band \((A, B)\) in \( L_\nu \), a color is associated as follows:

\[
\text{color}(s, \nu_{i}, A, B) =
\begin{cases}
\text{Red} & \text{if } \text{cd}(s, \nu\left(\frac{A + B}{2}\right), \nu_{i}) \\
\text{Green} & \text{else}
\end{cases}
\]

where \( \text{cd} \) is any correct conflict detection algorithm, such as \( \text{cd3d} \).

The following theorem can be easily proven from Definition 2.

**Theorem 1.** Given a function \( \nu : \mathbb{R} \rightarrow \mathbb{R}^3 \) and a closed interval \( I \subset \mathbb{R} \), a prevention bands algorithm for \( \nu \) is correct if and only if for any band \((A, B)\) in \( L_\nu \),

\[
\begin{align*}
\text{color}(s, \nu_{i}, A, B) &= \text{Red} & \iff & \forall y \in (A, B) : \text{conflict}?(s, \nu(y) - \nu_{i}), & \text{and} \\
\text{color}(s, \nu_{i}, A, B) &= \text{Green} & \iff & \forall y \in (A, B) : \neg \text{conflict}?(s, \nu(y) - \nu_{i}).
\end{align*}
\]

The relation between a graphical display such as in Figure 1 and the output of a prevention bands algorithm can be illustrated by considering the track angle display, that is, where \( \nu = \nu_{\text{trk}} \) and \( I = [0, 2\pi] \). A prevention bands algorithm for track angle will return a finite, ordered sequence \( L_{\nu_{\text{trk}}} \) of track angles in the interval \([0, 2\pi]\). This sequence will contain both of the angles 0 and 2\pi. If the algorithm is correct, then each consecutive pair, \( \alpha \) and \( \beta \), of track angles in this sequence defines a band, i.e., an open interval \((\alpha, \beta)\), with the property that either
Points computed by algorithm

- Midpoint angle in conflict
- Midpoint angle not in conflict

Figure 2. Relation between track angle prevention bands algorithm and graphical display

1. all track angles between \( \alpha \) and \( \beta \) result in conflict, or

2. all track angles between \( \alpha \) and \( \beta \) do not result in conflict.

If the track angles between \( \alpha \) and \( \beta \) all result in conflict, the region between \( \alpha \) and \( \beta \) is colored red. Otherwise, this region is colored green. The color of each such region is determined by conflict information at the midpoint \( \frac{\alpha + \beta}{2} \). This is illustrated by Figure 2.

2.5 Proving Correctness of a Prevention Bands Algorithm

This section provides a general strategy that can be followed to formally verify that a given prevention bands algorithm is correct. Subsequent sections will describe the use of this strategy in the formal verification of prevention bands algorithms for track angle, ground speed, and vertical speed.

Recall that a prevention bands algorithm depends on a function \( \nu : \mathbb{R} \rightarrow \mathbb{R}^3 \), (e.g. \( \nu = \nu_{\text{track}} \)), and a closed interval \( I = [I_1, I_2] \). Thus, a real-valued argument \( x \) of the function \( \nu \) is understood as a parameter of the ownship’s velocity vector, and the value \( \nu(x) \) is the corresponding velocity vector for that parameter. The following theorem can be used to verify the correctness of a prevention bands algorithm for \( \nu \) over \( I \).

**Theorem 2.** Let \( L_\nu \) be a finite sequence computed by a prevention bands algorithm for \( \nu \) over an interval \( I \) and let \( \Omega_\nu : \mathbb{R} \rightarrow \mathbb{R} \) be a continuous function, implicitly parametrized by \( s \) and \( v_i \), such that

1. \( \Omega_\nu \) characterizes conflict? in the following way:

\[
\Omega_\nu(x) < 1 \iff \text{conflict?}(s, \nu(x) - v_i), \quad (12)
\]
2. $L_v$ is $\Omega_v$-complete: For all real value $x \in I$,

$$\Omega_v(x) = 1 \iff x \in L_v,$$

then the algorithm that computes $L_v$ is correct.

Proof. By Theorem 1, it suffices to prove that Formulas (10) and (11) hold. Let $(A, B)$ be a band in $L_v$.

- Suppose that $\text{color}(s, v_i, A, B) = \text{Red}$ and let $y$ be a real number in the open interval $(A, B)$. Suppose, by reduction to absurdity, that $\neg \text{conflict?}(s, \nu(y) - v_i)$. By Hypothesis 1, $\Omega_v(y) \geq 1$. However, by Hypothesis 2, since $(A, B) \in L_v$ and $y$ is equal to neither $A$ nor $B$, it follows that $\Omega_v(y) > 1$. By the definition of the function $\text{color}$ given in Equation (9), it holds that $\text{conflict?}(s, \nu(x) - v_i)$, where $x = \frac{A + B}{2}$. Again by Hypothesis 1, $\Omega_v(x) < 1$. Since $\Omega_v$ is continuous, the intermediate value theorem implies that there exists some $z$ between $x$ and $y$ such that $\Omega_v(z) = 1$. Since $z$ is therefore in the interval $(A, B)$, $A$ and $B$ are consecutive in $L_v$, and the algorithm computes all points where $\Omega_v$ realizes a value of 1, this is a contradiction.

- Similar reasoning can be used to show that if $\text{color}(s, v_i, A, B) = \text{Green}$, then any $y$ in $(A, B)$ satisfies $\neg \text{conflict?}(s, \nu(y) - v_i)$.

$\Box$

3 The Function $\Omega$

Using Theorem 2 to verify that a prevention bands algorithm is correct for track angle, ground speed, or vertical speed maneuvers, i.e., for the functions $\nu_{\text{trk}}$, $\nu_{\text{gs}}$, and $\nu_{\text{vs}}$, will require finding three separate instantiations of the function $\Omega_v$ that satisfies all the hypotheses of the theorem. This section proposes the definition of a function $\Omega$ that can be used to define $\Omega_v$ for any $\nu : \mathbb{R} \rightarrow \mathbb{R}^3$, where some of these hypotheses can be discharged once and for all.

Let $\Omega : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ be a continuous function, implicitly parametrized by $s (= s_o - s_i)$, that characterizes $\text{conflict?}$ in the following way:

$$\Omega(v) < 1 \iff \text{conflict?}(s, v).$$

(14)

For any continuous function $\nu$, a continuous function $\Omega_v : \mathbb{R} \rightarrow \mathbb{R}$ that satisfies Equation (12) can be defined as follows:

$$\Omega_v(x) = \Omega(\nu(x) - v_i).$$

(15)

Therefore, since functions $\nu_{\text{trk}}$, $\nu_{\text{gs}}$, and $\nu_{\text{vs}}$ are continuous, Formula (15) can be used to construct continuous functions $\Omega_{\nu_{\text{trk}}}$, $\Omega_{\nu_{\text{gs}}}$, and $\Omega_{\nu_{\text{vs}}}$ that satisfy Equation (12) in Theorem 2.

Given such a function $\Omega$, the verification of correctness of a track angle, ground speed, and vertical speed prevention bands algorithms over an interval $I$ can be
reduced to proving that \( L_v \), i.e., the sequence returned by each algorithm, is \( \Omega_v \)-complete, i.e., it contains all \( x \in I \) where the function \( \Omega_v \) attains a value of 1. Since each of the algorithms will compute a sequence of values in a distinct way, a special proof of \( \Omega_v \)-completeness will be required for each algorithm that computes \( L_v \). The function \( \Omega \) will be of use in this step as well. Indeed, the function \( \Omega \) will be defined such that vectors \( v \) where \( \Omega(v) = 1 \) have particular forms. The proof that \( L_v \) is \( \Omega_v \)-complete, for \( v \in \{ \nu_{\text{trk}}, \nu_{\text{gs}}, \nu_{\text{vs}} \} \), will be done by proving that \( x \in L_v \) if and only if the vector \( \nu(x) \) has one of these forms.

The rest of this section concerns the definition of such a function \( \Omega \).

### 3.1 Cylindrical Distance

Recall from Section 2.1 that the protected zone is a cylinder around the intruder aircraft that has half-height \( H \) and radius \( D \). In order to define the function \( \Omega \) that satisfies Equation (14), a notion of cylindrical distance is needed.

**Definition 3.** The cylindrical length of a vector \( w \in \mathbb{R}^3 \) is the quantity

\[
\|w\|_{\text{cyl}} = \max\left( \frac{\|w_{(x,y)}\|}{D}, \frac{|w_z|}{H} \right).
\]

**Definition 4.** The cylindrical distance between two vectors, \( w_1 \) and \( w_2 \), is the quantity \( \|w_1 - w_2\|_{\text{cyl}} \).

Cylindrical distance is a metric on \( \mathbb{R}^3 \), in the sense of real analysis [5], and \( \mathbb{R}^3 \) is a metric space with this metric. In particular, this means that the triangle inequality holds for any \( w_0, w_1, w_2 \in \mathbb{R}^3 \):

\[
\|w_0 - w_2\|_{\text{cyl}} \leq \|w_0 - w_1\|_{\text{cyl}} + \|w_1 - w_2\|_{\text{cyl}}. \tag{16}
\]

The key property of cylindrical distance, as it relates to loss of separation of aircraft, is stated in the following theorem.

**Theorem 3.** Two aircraft are in loss of separation if and only if \( \|s\|_{\text{cyl}} < 1 \), where, as in Section 1, \( s = s_o - s_i \) is the relative position vector of the aircraft.

### 3.2 The Definition of \( \Omega \)

By Theorem 3, the ownship and the intruder aircraft are in conflict if and only if there exists some \( t \in [0, T] \) such that \( \|s + t v\|_{\text{cyl}} < 1 \). Thus, for \( s \) such that \( \|s\|_{\text{cyl}} \neq 1 \), i.e., for \( s \) not on the boundary of the protected zone, the function \( \Omega(v) \) is defined as

\[
\Omega(v) = \min_{t \in [0,T]} \|s + t v\|_{\text{cyl}}. \tag{17}
\]

Two important remarks on the definition of the function \( \Omega \) given by Formula (17) are in order. First, the function \( \Omega \) is well-defined since the quantity \( \|s + t v\|_{\text{cyl}} \) actually attains a minimum as \( t \) ranges over the interval \( [0, T] \). That is, there exists some \( \tau \in [0, T] \) such that \( \|s + \tau v\|_{\text{cyl}} \leq \|s + t v\|_{\text{cyl}} \) for all \( t \in [0, T] \). Indeed, when the vectors \( s \) and \( v \) are fixed, the function \( d_{\text{cyl}} : [0,T] \mapsto \mathbb{R} \) defined by \( d_{\text{cyl}}(t) = \|s + t v\|_{\text{cyl}} \)
is continuous, and every continuous function on a closed interval attains a minimum on that interval. The function $d_{cyl}$ is continuous because it is the maximum of two functions, $d_{horiz}$ and $d_{vert}$, defined by

$$d_{horiz}(t) = \frac{||((s + tv)_{x,y})||}{D},$$

$$d_{vert}(t) = \frac{||((s + tv)_{z})||}{H},$$

each of which are continuous.

Second, Formula (17) does not define $\Omega$ when $||s||_{cyl} = 1$. If $||s||_{cyl} = 1$, in which case $s$ is on the boundary of the cylinder, then any $v$ which points outward from the cylinder will satisfy $\min_{t \in [0,T]} ||s + tv||_{cyl} = 1$. This is because the minimum is attained at $t = 0$ for any such $v$. This is illustrated in Figure 3 in the case where $||s_{(x,y)}|| = D$ and $|s_z| < H$. Therefore, if $||s||_{cyl} = 1$, there is an infinite number of vectors $v$ such that $\min_{t \in [0,T]} ||s + tv||_{cyl} = 1$. Defining $\Omega$ in this case using Formula (17) would make the proof that $L_\nu$ is $\Omega_\nu$-complete impossible, as by definition of a prevention bands algorithm the sequence $L_\nu$ is finite.

While this shows that some care is needed when defining $\Omega$ on the boundary of the cylinder, it is possible to define $\Omega$ so that

1. it satisfies Equation (12),

2. it is continuous, and

3. it is suitable for showing that a sequence $L_\nu$ is $\Omega_\nu$-complete.

$$\Omega(v) = \begin{cases} 
  s_{(x,y)} \cdot v_{(x,y)} & \text{if } ||s_{(x,y)}|| = D \text{ and } |s_z| < H \\
  s_z v_z & \text{if } ||s_{(x,y)}|| < D \text{ and } |s_z| = H \\
  \max(s_{(x,y)} \cdot v_{(x,y)}, s_z v_z) & \text{if } ||s_{(x,y)}|| = D \text{ and } |s_z| = H \\
  \min_{t \in [0,T]} ||s + tv||_{cyl} & \text{otherwise, i.e., if } ||s||_{cyl} \neq 1
\end{cases}$$

(18)

The following theorem is a basic exercise in vector algebra.

**Theorem 4.** $\text{conflict}(s, v) \iff \Omega(v) < 1.$
The formal proof that $\Omega$ is continuous requires more work and it is explained in the rest of this section. Section 4 provides a classification theorem for $\Omega$, which is used then used in sections 5-7 to show that the sequences $L_\nu$, for $\nu \in \{\nu_{trk}, \nu_{gs}, \nu_{vs}\}$, computed by the proposed prevention bands algorithms, are $\Omega_\nu$-complete.

### 3.3 Continuity of $\Omega$

Since the if-statements in the definition of $\Omega$ do not depend on $\nu$, $\Omega$ is continuous if and only if each of the quantities $s(x,y) \cdot v(x,y)$, $s_z v_z$, $\max(s(x,y) \cdot v(x,y), s_z v_z)$, and $\min_{t \in [0,T]} \|s + t v\|_{cyl}$ are continuous functions of $v$. Only one of these four statements is nontrivial, that the minimum $\min_{t \in [0,T]} \|s + t v\|_{cyl}$ is continuous in $v$. This can be proved with standard techniques from real analysis [5]. In fact, it follows from a generalization of the Heine-Cantor theorem, which says that a continuous function on a closed interval is uniformly continuous. In particular, the following theorem has been proved.

**Theorem 5.** If $A$ and $B$ are real numbers with $A < B$ and $f: [A, B] \times \mathbb{R}^n \mapsto \mathbb{R}$ is continuous, then the function $g: \mathbb{R}^n \mapsto \mathbb{R}$ defined by $g(v) = \min_{t \in [A, B]} f(t, v)$ is continuous.

The formal proof of this theorem required the development of a vector analysis library in PVS, which is now part of the PVS NASA Libraries.\(^3\)

The continuity of $\Omega$ is a direct consequence of Theorem 5, when $A = 0$, $B = T$, and $f(t, v) = \|s + t v\|_{cyl}$.

**Theorem 6.** The function $\Omega$ is continuous.

The purpose for constructing the function $\Omega$ was to provide a definition for $\Omega_\nu: \mathbb{R} \mapsto \mathbb{R}$ for every function $\nu: \mathbb{R} \mapsto \mathbb{R}^3$. The following corollaries follow directly from theorems 4 and 6.

**Corollary 7.** For any $\nu: \mathbb{R} \mapsto \mathbb{R}^3$, the function $\Omega_\nu$, defined in Equation (15), satisfies $\Omega_\nu(x) < 1$ if and only if $\text{conflict}(s, \nu(x) - v_i)$.

**Corollary 8.** If $\nu: \mathbb{R} \mapsto \mathbb{R}^3$ is continuous, then the function $\Omega_\nu$ is continuous.

Since functions $\nu_{trk}$, $\nu_{gs}$, and $\nu_{vs}$ are continuous, corollaries 7 and 8 hold for $\Omega_{\nu_{trk}}$, $\Omega_{\nu_{gs}}$, and $\Omega_{\nu_{vs}}$.

### 4 Classification of Critical Vectors

To verify the correctness of a prevention bands algorithm for $\nu$ over a closed interval $I$, it must be shown that the computed sequence $L_\nu$ is finite and includes all points $x \in I$ such that $\Omega(\nu(x) - v_i) = 1$. Vectors $v$ that satisfy $\Omega(v) = 1$ are called critical vectors. This section shows that critical vectors can be analytically classified in a finite way.

Consider a relative position vector \( \mathbf{s} \) that satisfies \( \| \mathbf{s} \|_{\text{cyl}} \neq 1 \) and a critical vector \( \mathbf{v} \). Since \( \Omega(\mathbf{v}) = 1 \), it holds that \( \min_{t \in [0,T]} \|\mathbf{s} + t\mathbf{v}\|_{\text{cyl}} = 1 \). This minimum is attained at a real number \( \tau \in [0,T] \). Since \( \| \mathbf{s} \|_{\text{cyl}} \neq 1 \), it follows that \( \tau \neq 0 \). Thus, either \( \tau = T \) or \( 0 < \tau < T \). If it holds that \( v_z \neq 0 \), \( 0 < \tau < T \), \( |s_z + \tau v_z| = H \), and \( \|(s + \tau v)(x,y)\| < D \), then it can be shown that \( \min_{t \in [0,T]} \|\mathbf{s} + t\mathbf{v}\|_{\text{cyl}} < 1 \). That is, there is a time near \( \tau \) where the aircraft will be in loss of separation. This is illustrated in Figure 4.

If the same conditions hold, but with \( v_z = 0 \), then \( \tau \) is not unique, and it can also be shown that a particular \( \tau \) can be chosen so that \( 0 < \tau < T \), \( |s_z + \tau v_z| = H \), and \( \|(s + \tau v)(x,y)\| = D \).

Since, \( 1 = \Omega(\mathbf{v}) = \|\mathbf{s} + \tau\mathbf{v}\|_{\text{cyl}} = \max(\frac{\|(s + \tau v)(x,y)\|}{D}, \frac{|s_z + \tau v_z|}{H}) \), this leaves the following cases.

1. Case \( \tau = T \), \( |s_z + T v_z| = H \), and \( \|(s + T\mathbf{v})(x,y)\| < D \).
2. Case \( \tau = T \), \( |s_z + T v_z| < H \), and \( \|(s + T\mathbf{v})(x,y)\| = D \).
3. Case \( |s_z + \tau v_z| = H \) and \( \|(s + \tau v)(x,y)\| = D \).
4. Case \( 0 < \tau < T \), \( |s_z + \tau v_z| < H \), and \( \|(s + \tau v)(x,y)\| = D \).

These four cases are illustrated in figures 5, 6, 7, and 8, respectively.

These cases will be formalized using four predicates: \texttt{vertical\_case}\? (Section 4.1), \texttt{circle\_case\_2D}\? (Section 4.2), \texttt{circle\_case\_3D}\? (Section 4.3), and \texttt{line\_case}\? (Section 4.4). It will be shown in Section 4.5 that these four predicates are sufficient to classify solutions to the equation \( \Omega(\mathbf{v}) = 1 \), even in the case where \( \|\mathbf{s}\|_{\text{cyl}} = 1 \).
Figure 6. Case $\tau = T$, $|s_z + T v_z| < H$, and $\|(s + T v)(x,y)\| = D$

Figure 7. Case $|s_z + \tau v_z| = H$, and $\|(s + \tau v)(x,y)\| = D$

Figure 8. Case $0 < \tau < T$, $|s_z + \tau v_z| < H$, and $\|(s + \tau v)(x,y)\| = D$
4.1 Vertical Case

Consider the case 1 where \( \tau = T \), \( |s_z + T \cdot v_z| = H \), and \( \|(s + T \cdot v)_{(x,y)}\| < D \), which is illustrated by Figure 5. In this case, if \( (s_z + T \cdot v_z) \cdot v_z > 0 \), it can be formally proven that there is some \( t \in (0, T) \) such that \( \|s + t \cdot v\|_{\text{cyl}} < 1 \), which is a contradiction. This motivates the definition of the following predicate on \( s_z \), \( v_z \), a real number \( t \), and an integer \( \iota = \pm 1 \).

\[
\text{vertical\_case}\,(s_z, v_z, t, \iota) \equiv |s_z + t \cdot v_z| = H \text{ and } \iota (s_z + t \cdot v_z) \cdot v_z \geq 0.
\] (19)

Intuitively, the number \( \iota \) can be thought of as direction, with \( \iota = -1 \) corresponding to entry into the protected zone at time \( t \), and \( \iota = 1 \) corresponding to exit.

Case 1 corresponds to \( \text{vertical\_case}\,(s_z, v_z, T, -1) \). The condition

\[
\|(s + T \cdot v)_{(x,y)}\| < D
\]

is explicitly not included in this predicate, because the more general form is useful when classifying other types of critical vectors. It is important to note that if \( |s_z + T \cdot v_z| = H \), then \( \text{vertical\_case}\,(s_z, v_z, T, \iota) \) holds for some \( \iota = \pm 1 \).

Vectors \( v \) that satisfy the predicate \( \text{vertical\_case}\) are called vertical solutions.

4.2 Circle Case 2D

Consider the case 2 where \( \tau = T \), \( |s_z + T \cdot v_z| < H \), and \( \|(s + T \cdot v)_{(x,y)}\| = D \), which is illustrated by Figure 6. If \( (s_{(x,y)} + T \cdot v_{(x,y)}) \cdot v_{(x,y)} > 0 \), then it can be formally proven that there is some \( t \in (0, T) \) such that \( \|s + t \cdot v\|_{\text{cyl}} < 1 \), which is a contradiction. This motivates the definition of the following predicate on \( s \), \( v \), a real number \( t \), and \( \iota = \pm 1 \).

\[
\text{circle\_case\_2D}\,(s, v, t, \iota) \equiv \|(s + t \cdot v)_{(x,y)}\| = D \text{ and } \iota (s_{(x,y)} + t \cdot v_{(x,y)}) \cdot v_{(x,y)} \geq 0.
\] (20)

Case 2 corresponds to \( \text{circle\_case\_2D}\,(s, v, T, -1) \). The condition

\[
|s_z + T \cdot v_z| < H
\]

is not included in this predicate, because it will be used, along with \( \text{vertical\_case}\), to classify other types of critical vectors. As for the predicate \( \text{vertical\_case}\) above, an important property of \( \text{circle\_case\_2D}\) is that \( \|(s + t \cdot v)_{(x,y)}\| = D \) implies that \( \text{circle\_case\_2D}\,(s, v, t, \iota) \) holds for some \( \iota = \pm 1 \).

Vectors \( v \) that satisfy the predicate \( \text{circle\_case\_2D}\) are called 2D circle solutions.

4.3 Circle Case 3D

Consider the case 3 where \( |s_z + \tau \cdot v_z| = H \) and \( \|(s + \tau \cdot v)_{(x,y)}\| = D \), which is illustrated by Figure 7. It follows from the definitions of \( \text{vertical\_case}\) and \( \text{circle\_case\_2D}\) that there exists \( \iota_1, \iota_2 \), each equal to \(-1 \) or \(1 \), such that \( \text{vertical\_case}\,(s_z, v_z, \tau, \iota_1) \) and \( \text{circle\_case\_2D}\,(s, v, \tau, \iota_2) \). If \( \tau \) is positive and \( \iota_1 = \iota_2 \), it can be proven that either
vertical_case?(s_z, v_z, T, -1) or \( \Omega(v) < 1 \). In classifying the solutions to the equation \( \Omega(v) = 1 \), the case where \( \text{vertical_case}?(s_z, v_z, T, -1) \) is true is handled separately. Since it holds that \( \Omega(v) = 1 \), a requirement for the case where \( \|s_z + \tau v_z\| = H \) and \( \|(s + \tau v)(x,y)\| = D \) is therefore that \( \tau_1 = -\tau_2 \). This motivates the definition of the following predicate. Similar to the predicate \( \text{circle_case}_2D? \), this predicate depends on \( s, v, \tau = \pm 1 \), and a real number \( t \).

\[
circle_case_3D?(s, v, t, \tau) = t > 0 \quad \text{and} \quad circle_case_2D?(s, v, t, \tau) \quad \text{and} \quad vertical_case?(s_z, v_z, t, -\tau).
\]

Vectors \( v \) that satisfy the predicate \( \text{circle_case}_3D? \) are called 3D circle solutions.

### 4.4 Line Case

Consider the case where \( 0 < \tau < T, \|s_z + \tau v_z\| < H \), and \( \|(s + \tau v)(x,y)\| = D \), which is illustrated by Figure 8. As Figure 9 indicates, the fact that \( \tau \) satisfies \( \min_{t \in [0,T]} \|s + t v\|_{\text{cyl}} = \|s + \tau v\|_{\text{cyl}} \) can be used to show that the trajectory from \( s(x,y) \) along \( v(x,y) \) is tangent to the circle of radius \( D \) around the origin. In this figure, the vector \( v^\perp \) is the vector \( (v_y, -v_x, v_z) \).

It is immediately clear from Figure 9 that the angle \( \alpha \) can be no greater than \( \pi/2 \). Since \( s(x,y) \cdot v(x,y) = \|s(x,y)\| \|v(x,y)\| \cos \alpha \geq 0 \), it follows that \( s(x,y) \cdot v(x,y) \leq 0 \). In addition, \( \cos \beta = \frac{D}{\|s(x,y)\|} \). Thus,

\[
s(x,y) \cdot v^\perp(x,y) = \|s(x,y)\| \|v(x,y)\| \cos \beta = D \|v(x,y)\|.
\]

This construction depends on a vector \( v(x,y) \) that is tangent to the right side of the circle. The analogous construction for a vector \( v(x,y) \) that is tangent to the left side of the circle would use \(-v^\perp \) in the place of the vector \( v^\perp \). This motivates
the definition of the following predicate, which depends on $s$, $v$, and a parameter $\varepsilon$, which is equal to either $-1$ for a right-tangent, or $1$ for a left-tangent.

\[
\text{line}\_\text{case}(s, v, \varepsilon) \equiv s(x, y) \cdot v(x, y) \leq 0 \quad \text{and} \quad
-\varepsilon (s(x, y) \cdot v^\perp(x, y)) = D||v(x, y)||.
\] (23)

Vectors $v$ that satisfy the predicate $\text{line}\_\text{case}$ are called line solutions.

### 4.5 The Classification Theorem

Critical vectors can be classified according to the following theorem.

**Theorem 9.** If $\Omega(v) = 1$, then one of the following conditions holds.

1. $||s(x, y)|| \geq D$ and $\text{line}\_\text{case}(s, v, \iota)$ holds for some $\iota = \pm 1$.
2. $|s_x + T v_z| < H$ and $\text{circle}\_\text{case}\_2D(s, v, T, -1)$
3. There exists a real number $t > 0$ such that $\text{circle}\_\text{case}\_3D(s, v, t, t)$ holds for some $t = \pm 1$.
4. $||s(x, y) + T v(x, y)|| \leq D$ and $\text{vertical}\_\text{case}(s_z, v_z, T, -1)$

This theorem can be used to show that a sequence $L_v$ computed by a prevention bands algorithm is $\Omega_v$-complete by proving that $L_v$ contains all the vectors that have one of the four forms. It follows from this that $L_v$ contains all points $x \in I$ such that $\Omega_v(x) = 1$. When applying this technique to the case of track angle, ground speed, and vertical speed bands, it is still possible to find a few special cases where there are infinitely many points in $I$ at which $\Omega_v$ attains a value of $1$. These cases are handled separately by defining special versions of $\Omega_v$ that avoid this problem.

Section 4.6 defines functions $\Theta_H$ and $\Theta_D$ that compute the times where the aircraft lose vertical separation and horizontal separation, respectively, and illustrates the relation between these times and the four cases in the classification theorem (Theorem 9). The functions $\Theta_D$ and $\Theta_H$ will be used to define prevention bands algorithms for track angle, ground speed, and vertical speed maneuvers in sections 5, 6, and 7, respectively.

### 4.6 Entry and Exit Times

In Figure 5, the time $t$ at which the trajectory from $s$ along $v$ enters the protected zone vertically, i.e., where $(s + t v)_z = \pm H$, is precisely $T$. In Figure 6, the trajectory first touches the circle of radius $D$ around the origin at time $T$. In Figure 7, the time at which this trajectory enters the circle is precisely the time where its $z$-component exits the interval $[-H, H]$. In Figure 8, the trajectory is tangent to the circle, so the time where the trajectory first touches the circle is equal to the time where the trajectory last touches the circle.

All this indicates that there are relationships between the predicates defined in sections 4.1 to 4.4 and the following quantities:
• the times where the z-component of the trajectory from \( s \) along \( v \) enters and exits the interval \([-H, H]\), and

• the times where the 2-dimensional trajectory from \( s_{(x,y)} \) along \( v_{(x,y)} \) enters and exits the circle of radius \( D \) around the origin.

This section gives precise definitions of mathematical functions that compute these times and gives a variant of Theorem 9 that uses them.

The times where the z-component of the trajectory from \( s \) along \( v \) enters and exits the interval \([-H, H]\) are real numbers \( t \) such that \( |s_z + tv_z| = H \). This motivates the definition of the following function.

\[
\Theta_H(s_z, v_z, \iota) \equiv \frac{\iota \operatorname{sign}(v_z) H - s_z}{v_z}, \quad \text{for } v_z \neq 0,
\]

where the number \( \iota \) is \( \pm 1 \). It is easy to check that \( |s_z + \Theta_H(s_z, v_z, \iota)v_z| = H \). In addition,

\[
\Theta_H(s_z, v_z, -1) < \Theta_H(s_z, v_z, 1).
\]  

Intuitively, the times \( \Theta_H(s_z, v_z, -1) \) and \( \Theta_H(s_z, v_z, 1) \) are the times at which the z-component of the trajectory from \( s \) along \( v \) enters and exits the interval \([-H, H]\), respectively. It can be proved from definitions that \( \iota (s_z + \Theta_H(s_z, v_z, \iota)v_z) v_z \geq 0 \) for \( v_z \neq 0 \) and \( \iota = \pm 1 \).

**Lemma 10.** If \( v_z \neq 0 \), then \( |(s + tv)|_z = H \) if and only if \( t = \Theta_H(s_z, v_z, -1) \) or \( t = \Theta_H(s_z, v_z, 1) \).

**Corollary 11.** If \( v_z \neq 0 \) and \( \iota = \pm 1 \), then vertical-case \( \Theta(s_z, v_z, t, \iota) \) if and only if \( t = \Theta_H(s_z, v_z, \iota) \).

**Lemma 12.** If \( v_z \neq 0 \), then \( |(s + tv)|_z < H \) if and only if \( \Theta_H(s_z, v_z, -1) < t < \Theta_H(s_z, v_z, 1) \).

A similar construction can be used to find the times at which the trajectory from \( s_{(x,y)} \) along \( v_{(x,y)} \) enters and exits the circle of radius \( D \) around the origin. These times are real numbers \( t \) such that \( |(s + tv)|_{(x,y)}| = D^2 \). This is a quadratic equation in \( t \):

\[
\|v_{(x,y)}\|^2 t^2 + 2(s_{(x,y)} \cdot v_{(x,y)}) t + (\|s_{(x,y)}\|^2 - D^2) = 0. \tag{26}
\]

The roots of this quadratic equation are therefore given by the following function, where \( \iota = \pm 1 \).

\[
\Theta_D(s, v, \iota) \equiv \frac{-s_{(x,y)} \cdot v_{(x,y)} + \iota \sqrt{(s_{(x,y)} \cdot v_{(x,y)})^2 - \|v_{(x,y)}\|^2(\|s_{(x,y)}\|^2 - D^2)}}{\|v_{(x,y)}\|^2}. \tag{27}
\]

For this function to return a real number, it is required that the 2-dimensional vector \( v_{(x,y)} \) be nonzero and that discriminant of the quadratic equation (26) is nonnegative. That is, \( \Delta(s, v) \geq 0 \), where

\[
\Delta(s, v) \equiv (s_{(x,y)} \cdot v_{(x,y)})^2 - \|v_{(x,y)}\|^2(\|s_{(x,y)}\|^2 - D^2). \tag{28}
\]
The discriminant of the polynomial is given by $4\Delta(s, v)$. If $\Delta(s, v) \geq 0$, then

$$\Theta_D(s, v, 1) - \Theta_D(s, v, -1) = \frac{2\sqrt{(s(x,y) \cdot v(x,y))^2 - \|v(x,y)\|^2(\|s(x,y)\|^2 - D^2)}}{\|v(x,y)\|^2}.$$ 

Thus, $\Theta_D(s, v, -1) \leq \Theta_D(s, v, 1)$, and these two numbers are equal if and only if there is only one solution to the quadratic equation (26), which is equivalent to the statement that the line with direction $v$ that passes through $s$ is tangent to the circle of radius $D$ around the origin. It has been formally proved that $\iota(s(x,y) + \Theta_D(s, v, \iota) v(x,y)) \cdot v(x,y) \geq 0$ for $\Delta(s, v) \geq 0$, $v(x,y) \neq 0$, and $\iota = \pm 1$.

**Lemma 13.** If $v(x,y) \neq 0$, then $\|(s + t v(x,y))\| = D$ if and only if $\Delta(s, v) \geq 0$ and $t = \Theta_D(s, v, -1)$ or $t = \Theta_D(s, v, 1)$.

**Corollary 14.** If $\Delta(s, v) \geq 0$ and $v(x,y) \neq 0$, then $\text{circle_case}_2D? (s, v, t, \iota)$ if and only if $t = \Theta_D(s, v, \iota)$.

**Lemma 15.** If $v(x,y) \neq 0$, then $\|(s + t v(x,y))\| < D$ if and only if $\Delta(s, v) > 0$ and $\Theta_D(s, v, -1) < t < \Theta_D(s, v, 1)$.

The next result follows directly from corollaries 11 and 14.

**Corollary 16.** If $\Delta(s, v) \geq 0$, $v(x,y) \neq 0$, and $v_z \neq 0$, then $\text{circle_case}_3D? (s, v, t, \iota)$ if and only if $t > 0$ and the following string of equalities holds:

$$t = \Theta_D(s, v, \iota) = \Theta_H(s_z, v_z, -\iota).$$

Finally, the predicate defined in Section 4.4, $\text{line_case}?$, can also be be written in terms of the function $\Theta_D$. It is clear from definitions that $\Theta_D(s, v, -1) = \Theta_D(s, v, 1)$ is equivalent to the statement that the 2 dimensional trajectory from $s(x,y)$ along $v(x,y)$ is tangent to the circle of radius $D$ around the origin. This statement is made precise in the following corollary, which can be formally proven.

**Corollary 17.** If $s(x,y) \cdot v(x,y) \leq 0$ and $v(x,y) \neq 0$, then $\text{line_case}? (s, v, -1)$ or $\text{line_case}? (s, v, 1)$ holds if and only if $\Delta(s, v) \geq 0$ and $\Theta_D(s, v, -1) = \Theta_D(s, v, 1)$.

It follows from algebraic manipulations that if $\|s(x,y)\| \geq D$ and $\Omega(v) = 1$, then $s(x,y) \cdot v(x,y) \leq 0$.

## 5 Track Angle Prevention Bands

This section presents a formally verified algorithm, namely $\text{track.bands}$, for track angle prevention bands over the closed interval $[0, 2\pi]$, for the function $\nu_{\text{trk}} : \mathbb{R} \rightarrow \mathbb{R}^3$, defined by Equation (3) in Section 2.2. Given vectors $s$, $v_\circ$, and $v_i$, this algorithm computes track angle maneuvers, i.e., vectors $v'_\circ$ that satisfy $\|v'_\circ(x,y)\| = \|v_\circ(x,y)\|$ and $v'_\circ z = v_\circ z$.

The definition of $\text{track.bands}$ depends on the algorithms $\text{track.line}$, $\text{track.circle.2D}$, and $\text{track.circle.3D}$, which compute track angle maneuvers that are line solutions,
2D circle solutions, and 3D circle solutions, respectively. These three algorithms are proved to be complete, i.e., they compute all vectors that satisfy their respective predicate, and correct, i.e., only vectors that satisfy their respective predicate are computed. The correctness of track bands depends on the completeness of track_line, track_circle3D, and track_circle2D.

5.1 A Special Version of $\Omega_{\nu_{\text{trk}}}$

For $\nu = \nu_{\text{trk}}$, the function $\Omega_\nu$, defined in Equation (15) of Section 3, characterizes conflict in the sense of Corollary 7 (Section 3.3). In this section, $\nu$ will refer exclusively to the track angle function $\nu_{\text{trk}}$. To prove the correctness of a track angle prevention bands algorithm, it must be shown that the finite sequence $L_\nu$ returned by the algorithm contains all track angles $\alpha \in [0, 2\pi]$ such that $\Omega_\nu(\alpha) = 1$. An obvious requirement is that there be only finitely many track angles in the interval $[0, 2\pi]$ for which this equation holds. As it turns out, there are several special cases where this equation has infinitely many solutions for track angles $\alpha \in [0, 2\pi]$. Thus, a variant of $\Omega_\nu$, namely $\Omega^*_{\text{trk}}$, must be defined for these special cases.

Suppose that $s$, $v_o$, and $v_i$ satisfy $s(x,y) = Tv_i(x,y)$, $\|v_o(x,y)\|^2 = \frac{D^2}{T^2}$, and $|s_z + T v_z| < H$, where $v = v_o - v_i$. In this case,

$$
\|s(x,y) + T v(x,y)\|^2 = \|Tv_i(x,y) + (v_o(x,y) - v_i(x,y))\|^2 = \|Tv_o(x,y)\|^2 = T \|v_o(x,y)\|^2 = D^2.
$$

In addition, if $\alpha \in [0, 2\pi]$ is any track angle, then $\|v_{\text{trk}}(\alpha)(x,y)\| = \|v_o(x,y)\|$, and therefore this equality hold if $v_o$ is replaced with the vector $v_{\text{trk}}(\alpha)$. It follows immediately that for any $\alpha$, $\Delta(s, v_{\text{trk}}(\alpha) - v_i) \geq 0$. Lemma 13 in Section 4.6 implies that $T$ is equal to $\Theta_D(s, v_{\text{trk}}(\alpha) - v_i, \ell)$ for some $\ell = \pm 1$. If $\|s(x,y)\| > 1$, there are infinitely many track angles $\alpha$ such that $T = \Theta_D(s, v_{\text{trk}}(\alpha) - v_i, -1)$, in which case Lemma 15 in Section 4.6 implies that the minimum $\min_{t \in T} \|s + t (v_{\text{trk}}(\alpha) - v_i)\|_{\text{cyl}}$ is attained at $t = T$. Thus, if $\|s(x,y)\| > 1$, then the function $\Omega_\nu(\alpha) \equiv \Omega(v_{\text{trk}}(\alpha) - v_i)$ intersects the line at 1 at infinitely many points between 0 and $2\pi$. This special case is specified by the following predicate and illustrated in Figure 10.

$$
\text{track_spec}(s, v_o, v_i, t) \equiv s(x,y) = tv_i(x,y) \text{ and } \|v_o(x,y)\|^2 = D^2/T^2.
$$

An appropriate replacement $\Omega^*_{\text{trk}}$ for $\Omega_\nu$ in this case would have to satisfy the following two properties.

$$
\Omega^*_{\text{trk}}(\alpha) < 1 \iff \Omega_\nu(\alpha) < 1, \\
\Omega^*_{\text{trk}}(\alpha) \geq 1 \iff \Omega_\nu(\alpha) \geq 1.
$$

In addition, the function $\Omega^*_{\text{trk}}$ should allow only finitely many solutions to the equation $\Omega_{\text{trk}}(\alpha) = 1$ for $\alpha \in [0, 2\pi]$. If $\text{track_spec}(s, v_o, v_i, T)$ holds as above, then
the track angles \( \alpha \) such that \( T = \Theta_D(s, \nu_{\text{trk}}(\alpha) - v_i, -1) \) are precisely those angles \( \alpha \) such that \( \Omega_\nu(\alpha) \geq 1 \), and the angles \( \alpha \) such that \( T = \Theta_D(s, \nu_{\text{trk}}(\alpha) - v_i, 1) \) are precisely those angles such that \( \Omega_\nu(\alpha) < 1 \). Thus, it is easy to see that \( \Omega_{\text{trk}}^*(\alpha) = 1 \) should imply that the following two equalities hold.

\[
T = \Theta_D(s, \nu_{\text{trk}}(\alpha) - v_i, -1) = \Theta_D(s, \nu_{\text{trk}}(\alpha) - v_i, 1).
\]

By Corollary 14 and the definition of the predicate \( \text{circle_case_2D'} \), it follows that

\[
(s_{(x,y)} + T(\nu_{\text{trk}}(\alpha) - v_i)_{(x,y)}) \cdot (\nu_{\text{trk}}(\alpha) - v_i)_{(x,y)} = 0.
\]

Replacing \( s_{(x,y)} \) with \( T v_i_{(x,y)} \) and factoring out \( T \), this is equivalent to the statement that

\[
0 = \nu_{\text{trk}}(\alpha)_{(x,y)} \cdot (\nu_{\text{trk}}(\alpha) - v_i)_{(x,y)}
= \|\nu_{\text{trk}}(\alpha)_{(x,y)}\|^2 - \nu_{\text{trk}}(\alpha)_{(x,y)} \cdot v_i_{(x,y)}
= \|v_0_{(x,y)}\|^2 - \nu_{\text{trk}}(\alpha)_{(x,y)} \cdot v_i_{(x,y)}
= \frac{D^2}{T^2} - \nu_{\text{trk}}(\alpha)_{(x,y)} \cdot v_i_{(x,y)}.
\]

This motivates the following definition of the function \( \Omega_{\text{trk}}^*: \mathbb{R} \to \mathbb{R}^3 \), which depends on the explicit parameters \( v_0, v_i, t \in \mathbb{R} \), and \( \epsilon = \pm 1 \).

\[
\Omega_{\text{trk}}^*(v_0, v_i, t, \epsilon)(\alpha) = \epsilon \left( \nu_{\text{trk}}(\alpha)_{(x,y)} \cdot v_i_{(x,y)} - \frac{D^2}{T^2} \right) + 1.
\]  \( \text{(31)} \)

Identical reasoning to that above can be used to prove that if \( \text{track_spc}(s, v_0, v_i, t) \) holds and \( \Omega_{\text{trk}}^*(v_0, v_i, t, \epsilon)(\alpha) = 1 \), then the following equalities hold.

\[
t = \Theta_D(s, \nu_{\text{trk}}(\alpha) - v_i, -1) = \Theta_D(s, \nu_{\text{trk}}(\alpha) - v_i, 1).
\]

In particular, the following theorem can be formally proved using Corollary 17 in Section 4.6.

**Theorem 18.** If \( \text{track_spc}(s, v_0, v_i, t) \) holds and \( \Omega_{\text{trk}}^*(v_0, v_i, t, \epsilon)(\alpha) = 1 \), then \( \text{line_case}(s, \nu_{\text{trk}}(\alpha) - v_i, \epsilon) \) holds for some \( \epsilon = \pm 1 \).
When proving the correctness of the track angle prevention bands algorithm presented in the next sections, the function $\Omega_{trk}^*$ will be used in the place of $\Omega_v$ when $\text{track_spc?(s, v_o, v_i, T)}$ holds. Thus, it is necessary to prove that $\Omega_{trk}^*$ characterizes conflict in some special cases.

**Theorem 19.** If $\text{track_spc?(s, v_o, v_i, t)}$, then the equivalence

$$\Omega_{trk}^*(v_o, v_i, t, i)(\alpha) < 1 \iff \text{conflict?(s, } v_{\text{trk}}(\alpha) - v_i)$$

holds in each of the following three cases.

1. $\|s_{(x,y)}\| \geq D$, $v_{oz} \neq v_{iz}$, $t = \Theta_H(s_z, v_{oz} - v_{iz}, t)$, and $0 < t < T$.
2. $\|s_{(x,y)}\| \geq D$, $v_{oz} \neq v_{iz}$, $i = 1$, $t = \Theta_H(s_z, v_{oz} - v_{iz}, 1)$, and $t = T$.
3. $i = 1$, $t = T$, and $|s_z + T (v_{oz} - v_{iz})| < H$.

### 5.2 Line Solutions For Track Angle Maneuvers

The algorithm $\text{track line}$, defined in this section, takes as parameters $s$, $v_o$, $v_i$, $t$, $\epsilon = \pm 1$, and $i = \pm 1$. It returns a vector $v'_o \in \mathbb{R}^3$ that is either the zero vector or is equal to $v_{\text{trk}}(\alpha)$ for some $\alpha \in [0, 2\pi)$ such that the relative velocity vector $v' = v'_o - v_i$ is tangent to the circle, i.e., it satisfies $\text{line.case?}(s, v', \epsilon)$. The main theorem in this section states that $\text{track line}$ is correct and complete for line solutions that are track angle maneuvers.

The definition of $\text{track line}$ requires the definition an auxiliary function, namely $\text{tangent line}$, that takes as parameter a relative position vector $s \in \mathbb{R}^3$ such that $\|s_{(x,y)}\| \geq D$ and a number $\epsilon = \pm 1$, and returns a vector in $\mathbb{R}^3$ that is tangent to the protected zone.

$$\text{tangent line}(s, \epsilon) \equiv$$

\[
\text{if } \|s_{(x,y)}\| = D \text{ then} \]

\[
\epsilon s^\perp \]

\[
\text{else} \]

\[
\text{let } d = \|s_{(x,y)}\|^2 \text{ in} \]

\[
\left(\frac{D^2}{d} - 1\right) s + \frac{\epsilon D \sqrt{d - D^2}}{d} s^\perp \]

\[
\text{endif} \tag{32} \]

The proofs of the following lemmas rely on standard vector algebra.

**Lemma 20.** If $\|s_{(x,y)}\| \geq D$ and $\epsilon = \pm 1$, then $\text{line.case?}(s, \text{tangent line}(s, \epsilon), \epsilon)$ holds.

**Lemma 21.** If $\|s_{(x,y)}\| \geq D$, then $\text{line.case?}(s, v, \epsilon)$ holds if and only if there exists $k \geq 0$ such that

$$v_{(x,y)} = k \text{ tangent line}(s, \epsilon)_{(x,y)}.$$
If $v'_o \in \mathbb{R}^3$ is a track angle maneuver for the ownship such that $\text{line_case?}(s,v'_o-v_i,\varepsilon)$ holds, then it holds that
\begin{equation}
\|v_{o(x,y)}\|^2 = \|k \text{ tangent_line}(s,\varepsilon)(x,y) + v_{i(x,y)}\|^2.
\end{equation}

Equation (33) has the form $\|v_{o(x,y)}\|^2 = \|k u + v_{i(x,y)}\|^2$, where $u \in \mathbb{R}^2$. Since it will be necessary in later sections to solve similar equations of this form, a function is needed that explicitly solves this equation for $k \in \mathbb{R}$.

It follows from the equation $\|v_{o(x,y)}\|^2 = \|k u + v_{i(x,y)}\|^2$ that
\begin{equation}
0 = (k u + v_{i(x,y)}) \cdot (k u + v_{i(x,y)}) - \|v_{o(x,y)}\|^2
= \|u\|^2 k^2 + 2(v_{i(x,y)} \cdot u) k + (\|v_{i(x,y)}\|^2 - \|v_{o(x,y)}\|^2).
\end{equation}

This is a quadratic equation in $k$. If $\ell = \pm 1$, then $-b+\ell \sqrt{b^2-4ac}$ is a root of this equation, where
\begin{align*}
a &= \|u\|^2, \\
b &= 2v_{i(x,y)} \cdot u, \\
c &= \|v_{i(x,y)}\|^2 - \|v_{o(x,y)}\|^2.
\end{align*}

Thus, if $b^2 - 4ac \geq 0$ and $k = -b+\ell \sqrt{b^2-4ac}$ is nonnegative, then the unique vector $v'_o$ such that $v'_{oz} = v_{oz}$ and $v'_{o(x,y)} = k (u) + v_{i(x,y)}$ satisfies both $\|v'_{o(x,y)}\| = \|v_{o(x,y)}\|$ and $\text{line_case?}(s,v'_o-v_i,\varepsilon)$. This motivates the definition of the function $\text{track_only_line}$, which returns a real number.

\textit{track_only_line}(u,v_o,v_i,\ell) \equiv
\begin{align*}
\text{let} \\
a &= \|u\|^2, \\
b &= 2v_{i(x,y)} \cdot u, \\
c &= \|v_{i(x,y)}\|^2 - \|v_{o(x,y)}\|^2
\end{align*}
\begin{align*}
in \\
\text{if } b^2 - 4ac \geq 0 \text{ then} \\
&\frac{-b+\ell \sqrt{b^2-4ac}}{2a} \\
\text{else} \\
&0
\end{align*}
\end{verbatim}

The next lemma states that the algorithm $\text{track_only_line}$ computes solutions for $k$ to the equation $v'_{o(x,y)} = k u + v_{i(x,y)}$, where $\|v'_{o(x,y)}\| = \|v_{o(x,y)}\|$.

\textbf{Lemma 22.} If $u \neq 0$, then $\|v'_{o(x,y)}\| = \|v_{o(x,y)}\|$ and $k u = v'_{o(x,y)} - v_{i(x,y)}$ if and only if $k = \text{track_only_line}(u,v_o,v_i,\ell)$, for some $\ell = \pm 1$. 
Using \texttt{track-only-line}, the algorithm \texttt{track-line}, which computes track angle maneuvers \( v'_o \in \mathbb{R}^3 \) that satisfy \( \text{line-case}(s, v'_o - v_i, \varepsilon) \) for \( \varepsilon = \pm 1 \), can be defined as follows.

\[
\texttt{track-line}(s, v_o, v_i, \varepsilon, t) =
\]
\[
\text{let } k = \texttt{track-only-line}(\texttt{tangent-line}(s, \varepsilon)(x,y), v_o, v_i, t),
\]
\[
v'_o = (k \texttt{tangent-line}(s, \varepsilon)(x,y) + v_i(x,y)) \text{ with } [z \leftarrow v_{oz}]
\]
\[
in
\]
\[
\text{if } k \geq 0 \text{ then}
\]
\[
v'_o
\]
\[
\text{else}
\]
\[
0
\]
\[
\text{endif}
\]

The correctness and completeness of \texttt{track-line} follow from its definition and Lemma 22.

\textbf{Theorem 23} (Correctness and completeness of \texttt{track-line}). If \( \|s(x,y)\| \geq D \) and \( v'_o(x,y) \neq 0 \), then \( \|v'_o(x,y)\| = \|v_o(x,y)\|, v'_{ox} = v_{ox}, \) and \( \text{line-case}(s, v'_o - v_i, \varepsilon) \) holds if and only if

\[
v'_o = \texttt{track-line}(s, v_o, v_i, \varepsilon, t),
\]

for some \( t = \pm 1 \).

\subsection{2D Circle Solutions For Track Angle Maneuvers}

The algorithm \texttt{track-circle-2D}, defined in this section, takes as parameters \( s, v_o, v_i, t, \varepsilon = \pm 1, \) and \( \varepsilon = \pm 1 \). It returns a vector \( v'_o \in \mathbb{R}^3 \) that is either the zero vector or is equal to \( \nu_{\text{trk}}(\alpha) \) for some \( \alpha \in [0, 2\pi) \) such that the relative velocity vector \( v' = v'_o - v_i \) satisfies \( \text{circle-case}_2D?(s, v', t, \varepsilon) \). The main theorems in this section state that \texttt{track-circle-2D} is correct and complete for 2D circle solutions that are track angle maneuvers.

If \( \text{circle-case}_2D?(s, v', t, \varepsilon) \) holds, then the vector \( v'_o \) must satisfy

\[
\|s(x,y) + tv'(x,y)\|^2 = D^2.
\]

If \( \|v'_o(x,y)\| = \|v_o(x,y)\| \), then algebraic manipulations can be used to show that

\[
\|s(x,y) + tv'(x,y)\|^2 = \|s(x,y)\|^2 + t^2\|v_o(x,y)\|^2 + 2t(s(x,y) - tv_i(x,y)) \cdot v'(x,y) - t^2\|v_i(x,y)\|^2.
\]

Thus, if \( t > 0 \), then

\[
(s(x,y) - tv_i(x,y)) \cdot v'(x,y) = \frac{1}{2t} (D^2 - \|s(x,y)\|^2 - t^2(\|v_o(x,y)\|^2 - \|v_i(x,y)\|^2)).
\]

This equation has the form \( u \cdot v'(x,y) = j \), where \( u = s(x,y) - tv_i(x,y) \) and

\[
j = \frac{1}{2t} (D^2 - \|s(x,y)\|^2 - t^2(\|v_o(x,y)\|^2 - \|v_i(x,y)\|^2))
\]

23
Since it will be necessary in later sections to solve similar equations of the form $u \cdot v'_{(x,y)} = j$, a function is needed that explicitly solves this equation for $v'_o$ when $\|v'_o(x,y)\| = \|v_o(x,y)\|$. 

Assuming $u \neq 0$, Equation (1) yields

$$v'_o(x,y) = \frac{1}{\|u\|^2}((u \cdot v'_o(x,y))u + (u^\perp \cdot v'_o(x,y))u^\perp)$$

$$= \frac{1}{\|u\|^2}(ju + ku^\perp),$$

where $k = u^\perp \cdot v'_{(x,y)}$. Lemma 22 in Section 5.2 can be used to prove that

$$k = \text{track_only_line}(u^\perp_{(x,y)}, v_o, v_i, \frac{j}{\|u_{(x,y)}\|^2}u, \iota),$$

for some $\iota = \pm 1$.

It follows from this that for $u \neq 0$, the function $\text{track_only_dot}$, defined below, solves the equation $u \cdot (v'_o(x,y) - v_i(x,y)) = j$ for $v'_o$, when $\|v'_o(x,y)\| = \|v_o(x,y)\|$.

$$\text{track_only_dot}(u, v_o, v_i, j, \iota) =$$

$$\text{let } k = \text{track_only_line}(u^\perp_{(x,y)}, v_o, v_i, \frac{j}{\|u_{(x,y)}\|^2}u, \iota) \text{ in }$$

$$(ku^\perp + v_i(x,y) + \frac{j}{\|u_{(x,y)}\|^2}u) \text{ with } [z \leftarrow v_o z],$$

Lemma 24. If $u \neq 0$ and $v'_o(x,y) \neq 0$, then $\|v'_o(x,y)\| = \|v_o(x,y)\|$, $v'_o = v_o z$, and $u \cdot (v'_o(x,y) - v_i(x,y)) = j$ if and only if

$$v'_o = \text{track_only_dot}(u, v_o, v_i, j, \iota),$$

for some $\iota = \pm 1$.

The function $\text{track_only_dot}$ is used to solve Equation (38) when $v_i \neq 0$ and $t > 0$. Using $\text{track_only_dot}$, the algorithm $\text{track_circle.2D}$, which computes track angle maneuvers $v'_o \in R^3$ that satisfy $\text{circle_case.2D}(s, v'_o - v_i, t, \iota)$ for $\iota = \pm 1$, can be defined as follows.
track_circle_2D(s, v_o, v_i, t, t, c, e) ≜
let
u = (s — tv_i)(x,y),
\[ j = \frac{1}{2t} \left( D^2 - \| s(x,y) \|^2 - t^2(\| v_o(x,y) \|^2 - \| v_i(x,y) \|^2) \right) \]
in
if u ≠ 0 then
let
v'_o = track_only_dot(u, v_o, v_i, j, e)
in
if t (s + t(v'_o — v_i)) ≥ 0 then
v'_o
else
0
endif
else
0
endif

The correctness and completeness of track_circle_2D follow from its definition and Lemma 24.

**Theorem 25 (Correctness of track_circle_2D).** If \( v'_o(x,y) ≠ 0 \) and

\[ v'_o = track_circle_2D(s, v_o, v_i, t, t, c, e), \]
then \( \| v'_o(x,y) \| = \| v_o(x,y) \|, v'_o = v_o, \) and \( circle_case_2D?(s, v'_o — v_i, t, t) \) holds.

**Theorem 26 (Completeness of track_circle_2D).** If \( \| v'_o(x,y) \| = \| v_o(x,y) \|, v'_o = v_o, \) and \( circle_case_2D?(s, v'_o — v_i, t, t) \) holds, then either \( track_sp?c?(s, v_o, v_i, t) \) holds or

\[ v'_o = track_circle_2D(s, v_o, v_i, t, t, c, e) \]
for some \( \varepsilon = ±1 \).

### 5.4 3D Circle Solutions For Track Angle Maneuvers

Theorems 25 and 26 imply that the algorithm track_circle_2D can be used to compute vectors \( v'_o \) such that \( circle_case_2D?(s, v'_o — v_i, t, t) \) holds, where \( t > 0 \). By the definition of the predicate \( circle_case_3D? \) in Section 4.3, this algorithm can be used to compute vectors \( v'_o \) such that \( circle_case_3D?(s, v'_o — v_i, \Theta_H(s_z, v_o, v_i, v_i, —t), t) \) holds when \( \Theta_H(s_z, v_o, v_i, v_i, —t) > 0 \). This motivates the definition of the algorithm track_circle_3D, which takes as a parameters \( s, v_o, v_i, t = ±1, \) and \( \varepsilon = ±1 \). It returns a vector \( v'_o ∈ R^3 \) that is either the zero vector or is equal
to \( \nu_{\text{trk}}(\alpha) \) for some \( \alpha \in [0,2\pi) \) such that the relative velocity vector \( \mathbf{v}' = \mathbf{v}'_o - \mathbf{v}_i \) satisfies \( \text{circle\_case\_3D}(s, \mathbf{v}', \Theta_H(s_z, \mathbf{v}_{oz} - \mathbf{v}_{iz}, -t), t) \).

\[
\text{track\_circle\_3D}(s, \mathbf{v}_o, \mathbf{v}_i, t, \varepsilon) \equiv \\
\text{if } \mathbf{v}_{oz} = \mathbf{v}_{iz} \text{ then} \\
\quad 0 \\
\text{else} \\
\quad \text{let } t = \Theta_H(s_z, \mathbf{v}_{oz} - \mathbf{v}_{iz}, -t) \text{ in} \\
\quad \text{if } t > 0 \text{ then} \\
\quad \quad \text{track\_circle\_2D}(s, \mathbf{v}_o, \mathbf{v}_i, t, t, \varepsilon) \\
\quad \text{else} \\
\quad \quad 0 \\
\text{endif} \\
\text{endif}
\]

The following theorems state that \text{track\_circle\_3D} is correct and complete for 3D circle solutions that are track angle maneuvers. These properties follow from theorems 25 and 26, and properties of the function \( \Theta_H \) presented in Section 4.6.

**Theorem 27** (Correctness of track\_circle\_3D). If \( \mathbf{v}'_{o(x,y)} \neq 0 \) and

\[
\mathbf{v}'_o = \text{track\_circle\_3D}(s, \mathbf{v}_o, \mathbf{v}_i, t, \varepsilon),
\]

then \( ||\mathbf{v}'_{o(x,y)}|| = ||\mathbf{v}_{o(x,y)}||, \mathbf{v}'_{oz} = \mathbf{v}_{oz}, \) and \( \text{circle\_case\_3D}(s, \mathbf{v}'_o - \mathbf{v}_i, \Theta_H(s_z, \mathbf{v}_{oz} - \mathbf{v}_{iz}, -t), t) \) holds.

**Theorem 28** (Completeness of track\_circle\_3D). If \( ||\mathbf{v}'_{o(x,y)}|| = ||\mathbf{v}_{o(x,y)}||, \mathbf{v}'_{oz} = \mathbf{v}_{oz}, \mathbf{v}_{oz} \neq \mathbf{v}_{iz}, \) and \( \text{circle\_case\_3D}(s, \mathbf{v}'_o - \mathbf{v}_i, \Theta_H(s_z, \mathbf{v}_{oz} - \mathbf{v}_{iz}, -t), t) \) holds, then either \( \text{track\_spc}(s, \mathbf{v}_o, \mathbf{v}_i, \Theta_H(s_z, \mathbf{v}_{oz} - \mathbf{v}_{iz}, -t)) \) holds or

\[
\mathbf{v}'_o = \text{track\_circle\_3D}(s, \mathbf{v}_o, \mathbf{v}_i, t, \varepsilon),
\]

for some \( \varepsilon = \pm 1 \).

### 5.5 A Prevention Bands Algorithm For Track Angle Maneuvers

Using the functions defined in the previous section, the prevention bands algorithm \text{track\_bands} for the function \( \nu_{\text{trk}}: \mathbb{R} \to \mathbb{R}^3 \) can be defined as follows, where \( V \) is a sequence of vectors, \( |V| \) is its length, \( \mathcal{L} \) is a set of real numbers, and \text{sort} is a function that takes as parameter a set of real numbers and returns the sequence of elements in the set that is sorted by increasing order.\(^4\)

\(^4\)For readability, the algorithm is written using pseudo-code including assignment and bounded loop constructions. The PVS development provides a functional version of this code.
\text{track\_bands}(s, v_0, v_i) =
\begin{align*}
V_0 & := \text{track\_circle\_3D}(s, v_0, v_i, -1, -1); \\
V_1 & := \text{track\_circle\_3D}(s, v_0, v_i, -1, 1); \\
V_2 & := \text{track\_circle\_3D}(s, v_0, v_i, 1, -1); \\
V_3 & := \text{track\_circle\_3D}(s, v_0, v_i, 1, 1); \tag{42}
\end{align*}

\text{if} \|s_{(x,y)}\| \geq D \text{ then}
\begin{align*}
V_4 & := \text{track\_circle\_2D}(s, v_0, v_i, T, -1, -1); \\
V_5 & := \text{track\_circle\_2D}(s, v_0, v_i, T, -1, 1); \\
V_6 & := \text{track\_line}(s, v_0, v_i, -1, -1); \\
V_7 & := \text{track\_line}(s, v_0, v_i, -1, 1); \\
V_8 & := \text{track\_line}(s, v_0, v_i, 1, -1); \\
V_9 & := \text{track\_line}(s, v_0, v_i, 1, 1); \\
\end{align*}
\text{endif}
\begin{align*}
\mathcal{L} & = \{0, 2\pi\}; \\
\text{for} \ i = 1 \ \text{to} \ |V| \ \text{do} \\
\text{if} \ V_i \not\equiv 0 \ \text{then} \\
\mathcal{L} & := \mathcal{L} \cup \{\text{track}(V_i)\}; \\
\text{endif} \\
\text{endfor} \\
L_{\nu_{\text{trk}}} & := \text{sort}(\mathcal{L});
\end{align*}

The finite, ordered sequence $L_{\nu_{\text{trk}}}$ returned by \text{track\_bands} is computed using every possible instantiation of the parameters $\varepsilon$ and $\tau$, both of which can be $\pm 1$, in the functions \text{track\_line}, \text{track\_circle\_2D}, and \text{track\_circle\_3D}. For each vector $v'_o$ returned by one of these three algorithms for $s$, $v_o$, and $v_i$ with the property that $v'_{o(x,y)} \not\equiv 0$, the track angle of $v'_o$ is an element of the sequence returned by \text{track\_bands}.

**Theorem 29 (Correctness of \text{track\_bands}).** The track angle prevention bands algorithm \text{track\_bands} is correct for $\nu_{\text{trk}}$ over the interval $[0, 2\pi]$.

**Proof.** By Theorem 2 in Section 2.5, it suffices to find a continuous function $\Omega_\nu : \mathbb{R} \mapsto \mathbb{R}$, parameterized by $s$, $v_o$, and $v_i$, that satisfies the following two properties.

1. For all $\alpha \in [0, 2\pi],$
   \[ \Omega_\nu(\alpha) < 1 \iff \text{conflict}(s, \nu_{\text{trk}}(\alpha) - v_i). \]

2. For all $\alpha \in [0, 2\pi],$
   \[ \Omega_\nu(\alpha) = 1 \implies \alpha \in \text{track\_bands}(s, v_0, v_i). \]
In most cases, the function $\Omega_\nu$, where $\nu = \nu_{trk}$, defined in Equation (15) of Section 3, will suffice. However, in some special cases, the function $\Omega^*_{trk}$, defined in Equation (31) of Section 5.1, will be used. The latter case is considered first.

Suppose that $track_spc?(s, v_0, v_1, t)$, where $t > 0$, and that one of the following conditions holds.

1. $\|s(x,y)\| \geq D$, $v_0 \neq v_1$, $t = \Theta_H(s_z, v_0z - v_1z, \epsilon)$, and $0 < t < T$.
2. $\|s(x,y)\| \geq D$, $v_0 \neq v_1$, $t = 1$, $t = \Theta_H(s_z, v_0z - v_1z, 1)$, and $t = T$.
3. $t = 1$, $t = T$, and $|s_z + T (v_0z - v_1z)| < H$.

By Theorem 19 in Section 5.1,

$$\Omega^*_{trk}(v_0, v_1, t, \epsilon)(\alpha) < 1 \iff conflict?(s, v_{trk}(\alpha) - v_1)$$

holds for any $\alpha \in \mathbb{R}$. Thus, all that is required to complete the proof in this special case is to prove that for all $\alpha \in [0, 2\pi]$, $\Omega^*_{trk}(\alpha) = 1$ implies

$$\alpha \in track\_bands(s, v_0, v_1).$$

If $\Omega^*_{trk}(\alpha) = 1$, then Theorem 18 implies that $line\_case?(s, v_{trk}(\epsilon) - v_1, \epsilon)$, for some $\epsilon = \pm 1$. By the completeness of the algorithm $track\_line$ (Theorem 23 in Section 5.2), if $|s(x,y)| \geq D$, then $v_{trk}(\alpha)$ is equal to $track\_line(s, v_0, v_1, \epsilon, t)$, for some $t = \pm 1$. Thus, $\alpha = track(v_{trk}(\alpha))$ is equal to $track(track\_line(s, v_0, v_1, \epsilon, t))$, which, by definition, is an element of $track\_bands(s, v_0, v_1)$. If $|s(x,y)| < D$, then it must be true that the third condition holds: $t = 1$, $t = T$, and $|s_z + T (v_0z - v_1z)| < H$. In this case, it is easy to prove that for any $\alpha \in \mathbb{R}$, $conflict?(s, v_{trk}(\alpha) - v_1)$, and therefore $\Omega^*_{trk}(\alpha) < 1$. This completes the proof in the case where one of the three conditions above holds.

Now suppose that the second condition above holds, but where $t = 1$ is replaced with $t = -1$. That is, suppose that $|s(x,y)| \geq D$, $v_0 \neq v_1$, $t = \Theta_H(s_z, v_0z - v_1z, -1)$, $t = T$, and $track\_spec?(s, v_0, v_1, T)$. Since $v_{trk}(\alpha)_z = v_0z$ for any $\alpha \in \mathbb{R}$, Lemma 12 of Section 4.6 can be used to show that $conflict?(s, v_{trk}(\alpha) - v_1)$ does not hold for any $\alpha \in \mathbb{R}$. In this case, the correctness of the algorithm $track\_bands$ is trivial.

The proof has now been reduced to the case where neither of the following conditions hold.

1. $v_0 \neq v_1$ and there exists $t = \pm 1$ such that $track\_spec?(s, v_0, v_1, t)$ and $0 < t \leq T$, where $t = \Theta_H(s_z, v_0z - v_1z, \epsilon)$.

2. $track\_spec?(s, v_0, v_1, T)$ and $|s_z + T (v_0z - v_1z)| < H$.

By Corollary 7 of Section 3.3, the function $\Omega_\nu$, where $\nu = \nu_{trk}$, characterizes conflict. Suppose that $\alpha \in [0, 2\pi]$ and $\Omega_\nu(\alpha) = 1$. Since $\Omega_\nu(\alpha) = \Omega(v_{\nu_{trk}}(\alpha) - v_1)$, Theorem 9 in Section 4.5 implies that one of the following conditions holds, where $v = v_{\nu_{trk}}(\alpha) - v_1$.

- $\|s(x,y)\| \geq D$ and either $line\_case?(s, v, \epsilon)$, for some $\epsilon = \pm 1$. 

28
• \(|s_z + T v_z| < H\) and circle_case_2D?(s, v, T, -1).

• There is some real number \(t > 0\) such that circle_case_3D?(s, v, t, \(v_i\)), for some \(t = \pm 1\).

• \(|s(x,y) + T v(x,y)| \leq D\) and vertical_case?(s_z, v_z, T, -1).

These cases are now considered individually.

• Suppose first that \(|s(x,y)| \geq D\) and line_case?(s, \(v_{trk}(\alpha) - v_i, \varepsilon\)), for some \(\varepsilon = \pm 1\). By completeness of track_line (Theorem 23), \(v_{trk}(\alpha)\) is equal to track_line(s, v_o, v_i, \(\varepsilon, t\)), for some \(t = \pm 1\). Thus, \(\alpha = \text{track}(v_{trk}(\alpha))\) is equal to \(\text{track}(\text{track_line}(s, v_o, v_i, \varepsilon, t))\), which, by definition, is an element of \(\text{track_bands}(s, v_o, v_i)\).

• Suppose that \(|s_z + T v_z| < H\) and circle_case_2D?(s, \(v_{trk}(\alpha) - v_i, T, -1\)). By completeness of the algorithm track_circle_2D (Theorem 26), \(v_{trk}(\alpha)\) is equal to track_circle_2D(s, v_o, v_i, t, \varepsilon) for some \(t = \pm 1\) and \(\varepsilon = \pm 1\). Thus, \(\alpha = \text{track}(v_{trk}(\alpha)) = \text{track}\left(\text{track_circle_2D}(s, v_0, v_i, t, \varepsilon)\right)\). Hence, \(\alpha\) is an element of \(\text{track_bands}(s, v_o, v_i)\).

• Suppose that there is a real number \(t > 0\) such that circle_case_3D?(s, \(v, t, \varepsilon\)), where \(t = \pm 1\). Assume that \(v_{oz} \neq v_{iz}\). By completeness of track_circle_3D (Theorem 28), \(v_{trk}(\alpha) = \text{track_circle_3D}(s, v_0, v_i, t, \varepsilon)\) for some \(t = \pm 1\) and \(\varepsilon = \pm 1\). Thus, as above,

\[
\alpha = \text{track}(v_{trk}(\alpha)) = \text{track}(\text{track_circle_3D}(s, v_0, v_i, t, \varepsilon)).
\]

Hence, \(\alpha\) is an element of \(\text{track_bands}(s, v_0, v_i)\). The case where \(v_{oz} = v_{iz}\) can be equally discharged.

• Finally, suppose that \(|s(x,y) + T v(x,y)| \leq D\) and vertical_case?(s_z, v_z, T, -1). In this case, the fact that \(v_{trk}(\alpha) = v_{oz}\) implies that conflict?(s, \(v_{trk}(\alpha) - v_i\)) does not hold for any \(\alpha \in \mathbb{R}\). From there, the correctness of the algorithm track_bands is trivial.

6 Ground Speed Prevention Bands

This section presents a formally verified algorithm, namely \texttt{gs.bands}, for ground speed prevention bands over an arbitrary interval \([\text{gsmin}, \text{gsmax}]\) for the function \(v_{gs}: \mathbb{R} \rightarrow \mathbb{R}^3\), defined by Equation (4) in Section 2.2. The boundaries of the interval, \texttt{gsmin} and \texttt{gsmax}, represent (positive) minimum and maximum ground speeds for the ownship aircraft, respectively. Given vectors \(s, v_o,\) and \(v_i\), this algorithm computes ground speed maneuvers, i.e., vectors \(v'_o\) that satisfy \(v'_o(x,y) = \ell v_o(x,y)\), for some \(\ell > 0\), and \(v'_o = v_{oz}\).

The definition of \texttt{gs.bands} depends on the algorithms \texttt{gs.line}, \texttt{gs.circle.2D}, and \texttt{gs.circle.3D}, which compute ground speed maneuvers that are line solutions,
2D circle solutions, and 3D circle solutions, respectively. These three algorithms are proved to be complete and correct for ground speed maneuvers that satisfy their respective predicate. The correctness of \texttt{gs.bands} depends on the completeness of \texttt{gs.line}, \texttt{gs.circle.3D}, and \texttt{gs.circle.2D}.

If \( v'_o \) is a ground speed maneuver for the ownship, then there is some positive \( p \in \mathbb{R} \) such that \( \nu_{gs}(p) = v'_o \). Therefore, \( v'_{o(x,y)} = \ell v_{o(x,y)} \), where \( \ell = \frac{p}{||v_{o(x,y)}||} \) and \( \ell > 0 \).

6.1 Line Solutions For Ground Speed Maneuvers

The algorithm \texttt{gs.line}, defined in this section, takes as parameters \( s, v_o, v_i, t, \) and \( \varepsilon = \pm 1 \). It returns a vector \( v'_o \in \mathbb{R}^3 \) that is either the zero vector or is equal to \( \nu_{gs}(p) \) for some \( p \in \mathbb{R} \) such that the relative velocity vector \( v' = v'_o - v_i \) is tangent to the circle, i.e., it satisfies \texttt{line.case?}(s, \( v' \), \( \varepsilon \)). The main theorem in this section states that \texttt{gs.line} is correct and complete for line solutions that are ground speed maneuvers.

Suppose \( ||s(x,y)|| \geq D \) and that \( v'_o \) is a vector in \( \mathbb{R}^3 \) such that \( v'_{o(x,y)} = \ell v_{o(x,y)} \). Suppose further that \texttt{line.case?}(s, \( v'_o - v_i \), \( \varepsilon \)) holds for some \( \varepsilon = \pm 1 \). By Lemma 21 of Section 5.2, there is some \( k \geq 0 \) such that \( \ell v_{o(x,y)} = k \texttt{tangent.line}(s, \varepsilon) + v_i(x,y) \). This equation has the form

\[
\ell v_{o(x,y)} = ku + v_i(x,y),
\]

where \( u \in \mathbb{R}^3 \). Functions can be defined that explicitly solve this equation for \( k \) and \( \ell \). It is easily proved that

\[
(v_i(x,y) \cdot u^\perp) v_{o(x,y)} - (v_{o(x,y)} \cdot u^\perp) v_i(x,y) = (v_i(x,y) \cdot v_{o(x,y)}^\perp) u.
\]

Thus, if \( v_{o(x,y)} \cdot u^\perp \neq 0 \), then

\[
k = \frac{v_i(x,y) \cdot v_{o(x,y)}^\perp}{v_{o(x,y)} \cdot u^\perp},
\]

\[
\ell = \frac{v_i(x,y) \cdot u^\perp}{v_{o(x,y)} \cdot u^\perp}.
\]

This motivates the definition of the algorithms \texttt{gs.line.k} and \texttt{gs.line.l}, which solve Equation (43) for \( k \) and \( \ell \).

\[
\texttt{gs.line.k}(u, v_o, v_i) \equiv
\begin{align*}
\text{if } & v_{o(x,y)} \cdot u^\perp \neq 0 \text{ then} \\
& \frac{v_i(x,y) \cdot v_{o(x,y)}^\perp}{v_{o(x,y)} \cdot u^\perp} \\
\text{else} & 0 \\
\text{endif}
\end{align*}
\]
Lemma 30. If $\ell > 0$ and either $v_i(x, y) \cdot u_o(x, y) \neq 0$ or $v_o(x, y) \cdot u_o \neq 0$, then Equation (43) holds if and only if $k = g_s.l_i.e\_k(u, v_o, v_i)$ and $\ell = g_s.l_i.e\_l(u, v_o, v_i)$.

Using $g_s.l_i.e\_k$ and $g_s.l_i.e\_l$, the algorithm $g_s.l_i.e\_i.e\_l$, which computes ground speed maneuvers $v'_o \in \mathbb{R}^3$ that satisfy line_case?($s, v'_o - v_i, \varepsilon$) for $\varepsilon = \pm 1$, can be defined as follows.

$$g_s.l_i.e\_i.e\_l(s, v_o, v_i, \varepsilon)$$

let

$$u = t a n g e n t\_l_i.e\_i.e\_i.e\_n(e)(x, y)$$

$$k = g_s.l_i.e\_k(u, v_o, v_i)$$

$$\ell = g_s.l_i.e\_l(u, v_o, v_i)$$

in

if $k \geq 0$ then

$$\ell v_o(x, y)$$

with $[z \leftarrow v_o(z)]$

else

0

endif

endif

The correctness and completeness of $g_s.l_i.e\_i.e\_l$ follow from its definition and Lemma 30.

Theorem 31 (Correctness and completeness of $g_s.l_i.e\_i.e\_l$). If $\|s(x, y)\| \geq D$, $v'_o(x, y) \neq 0$, and either $v_i(x, y) \cdot v'_o(x, y) \neq 0$ or $v_o(x, y) \cdot u_o \neq 0$, then $v'_o = v_o$, $v'_o(x, y) = \ell v_o(x, y)$ for some $\ell > 0$, and line_case?($s, v'_o - v_i, \varepsilon$) holds if and only if

$$v'_o = g_s.l_i.e\_i.e\_l(s, v_o, v_i, \varepsilon).$$

This theorem does not hold if $\|s(x, y)\| \geq D$, $v_i(x, y) \cdot v'_o(x, y) = 0$, and $v_o(x, y) \cdot t a n g e n t\_l_i.e\_i.e\_i.e\_n(e)(x, y) = 0$. This case has to be handled separately in the verification of correctness of the ground speed prevention bands algorithm.

31
6.2 2D Circle Solutions For Ground Speed Maneuvers

The algorithm $\text{gs\_circle\_2D}$, defined in this section, takes as parameters $s$, $v_o$, $v_i$, $t$, $\varepsilon = \pm 1$, and $\varepsilon = \pm 1$. It returns a vector $v'_o \in \mathbb{R}^3$ such that $v'_o$ is either the zero vector or is equal to $\nu_{gs}(p)$ for some $p > 0$ such that the relative velocity vector $v = v'_o - v_i$ satisfies $\text{circle\_case\_2D}(s, v'_o - v_i, t, \varepsilon)$. The main theorems in this section state that $\text{gs\_circle\_2D}$ is correct and complete for 2D circle solution that are ground speed maneuvers.

If $\text{circle\_case\_2D}(s, v'_o - v_i, t, \varepsilon)$ holds, then the vector $v'_o$ must satisfy

$$||s_{(x,y)} + t (v'_{o(x,y)} - v_{i(x,y)})||^2 = D^2.$$ 

If $v'_{o(x,y)} = \ell v_{o(x,y)}$, then simple algebraic manipulation can be used to show that $\ell$, $\ell^2 + b \ell + c = 0$, where

$$a = t^2 \|v_{o(x,y)}\|^2,$$
$$b = 2t (s - t v_i) \cdot v_{o(x,y)},$$
$$c = \| (s - t v_i) \|^2 - D^2.$$

This is a quadratic equation in $\ell$, which can be solved using the quadratic formula. Note that if $v'_o$ represents a ground speed maneuver for the ownship, then $\ell$ must be positive, since $v'_{o(x,y)} = \ell v_{o(x,y)}$. This motivates the following definition of the algorithm $\text{gs\_circle\_2D}$, which computes ground speed maneuvers $v'_o \in \mathbb{R}^3$. 

32
that satisfy \( \text{circle\_case\_2D?}(s, v'_o - v_i, t, \epsilon) \) for \( \epsilon = \pm 1 \).

\[
\text{gs\_circle\_2D}(s, v_o, v_i, t, \epsilon) \equiv
\]
\[
\text{let}
\]
\[
a = t^2 \|v_o(x,y)\|^2
\]
\[
b = 2t (s - t v_i) \cdot v_o(x,y)
\]
\[
c = \|(s - t v_i)(x,y)\|^2 - D^2
\]
\[
in
\]
\[
\text{if } b^2 - 4ac \geq 0 \text{ then}
\]
\[
\text{let}
\]
\[
\ell = \frac{b^2 + \epsilon \sqrt{b^2 - 4ac}}{2a}
\]
\[
v'_o = \max(\ell, 0) \ v_o(x,y) \text{ with } [z \leftarrow v_{oz}]
\]
\[
in
\]
\[
\text{if } (s + t(v'_o - v_i))(x,y) \cdot (v'_o - v_i)(x,y) \geq 0 \text{ then}
\]
\[
v'_o
\]
\[
\text{else}
\]
\[
0
\]
\[
\text{endif}
\]
\[
\text{else}
\]
\[
0
\]
\[
\text{endif}
\]

The correctness and completeness of \text{gs\_circle\_2D} follow from its definition and the correctness and completeness of the quadratic formula, which has been proved in PVS.

**Theorem 32 (Correctness of \text{gs\_circle\_2D}).** If \( v_o(x,y) \neq 0 \) and

\[
v'_o = \text{gs\_circle\_2D}(s, v_o, v_i, t, \epsilon),
\]

then \( \text{circle\_case\_2D?}(s, v'_o - v_i, t, \epsilon) \) holds, \( v'_{oz} = v_{oz} \), and \( v'_o(x,y) = \ell v_o(x,y) \) for some \( \ell > 0 \).

**Theorem 33 (Completeness of \text{gs\_circle\_2D}).** If \( v'_o(x,y) = \ell v_o(x,y) \), \( \ell > 0 \), \( v'_{oz} - v_{oz} \), and \( \text{circle\_case\_2D?}(s, v'_o - v_i, t, \epsilon) \) holds, then

\[
v'_o = \text{gs\_circle\_2D}(s, v_o, v_i, t, \epsilon),
\]

for some \( \epsilon = \pm 1 \).

### 6.3 3D Circle Solutions For Ground Speed Maneuvers

Theorems 32 and 33 imply that the algorithm \text{gs\_circle\_2D} can be used to compute vectors \( v'_o \) such that \( \text{circle\_case\_2D?}(s, v'_o - v_i, t, \epsilon) \) holds, where \( t > 0 \). By the
of the predicate \texttt{circle\_case\_3D}\? in Section 4.3, this algorithm can be used to compute vectors \( \mathbf{v}' \) such that \( \text{circle\_case\_3D}(s, \mathbf{v}'_o - \mathbf{v}_i, \Theta_H(s_z, v_{oz} - v_{iz}, -\ell), \ell) \) holds when \( \Theta_H(s_z, v_{oz} - v_{iz}, -\ell) > 0 \). This motivates the definition of the algorithm \texttt{gs\_circle\_3D}, which takes as a parameters \( s, \mathbf{v}_o, \mathbf{v}_i, \ell, \) and \( \varepsilon \). It returns a vector \( \mathbf{v}'_o \in \mathbb{R}^3 \) such that the relative velocity vector \( \mathbf{v}' = \mathbf{v}'_o - \mathbf{v}_i \) satisfies \( \text{circle\_case\_3D}(s, \mathbf{v}', \Theta_H(s_z, v_{oz} - v_{iz}, -\ell), \ell) \).

\begin{equation}
\text{gs\_circle\_3D}(s, \mathbf{v}_o, \mathbf{v}_i, \ell, \varepsilon) =
\begin{cases}
0 & \text{if } v_{oz} = v_{iz} \\
\text{let } t = \Theta_H(s_z, v_{oz} - v_{iz}, -\ell) \\
\text{in } \begin{cases}
 0 & \text{if } t > 0 \\
 0 & \text{else}
\end{cases}
\end{cases}
\end{equation}

The following theorems state that \texttt{gs\_circle\_3D} is correct and complete for 3D circle solutions that are ground speed maneuvers. These properties follow from theorems 32 and 33, and properties of the function \( \Theta_H \) presented in Section 4.6.

\textbf{Theorem 34 (Correctness of gs\_circle\_3D).} If \( \mathbf{v}'_{o(x,y)} \neq 0 \) and

\[ \mathbf{v}'_o = \text{gs\_circle\_3D}(s, \mathbf{v}_o, \mathbf{v}_i, \ell, \varepsilon), \]

then \( \text{circle\_case\_3D}(s, \mathbf{v}'_o - \mathbf{v}_i, \Theta_H(s_z, v_{oz} - v_{iz}, -\ell), \ell) \) holds, \( \mathbf{v}'_{oz} = \mathbf{v}_{oz} \), and \( \mathbf{v}'_{o(x,y)} = \ell \mathbf{v}_{o(x,y)} \) for some \( \ell > 0 \).

\textbf{Theorem 35 (Completeness of gs\_circle\_3D).} If \( \mathbf{v}'_{o(x,y)} = \ell \mathbf{v}_{o(x,y)}, \ell > 0, \mathbf{v}'_{oz} = \mathbf{v}_{oz}, \mathbf{v}_{oz} \neq \mathbf{v}_{iz} \), and \( \text{circle\_case\_3D}(s, \mathbf{v}'_o - \mathbf{v}_i, \Theta_H(s_z, v_{oz} - v_{iz}, -\ell), \ell) \) holds, then

\[ \mathbf{v}'_o = \text{gs\_circle\_3D}(s, \mathbf{v}_o, \mathbf{v}_i, \ell, \varepsilon), \]

for some \( \varepsilon = \pm 1 \).

\textbf{6.4 A Prevention Bands Algorithm For Ground Speed Maneuvers}

The prevention bands algorithm \texttt{gs\_bands} for the function \( \nu_{gs}: \mathbb{R} \mapsto \mathbb{R}^3 \) that computes a sorted sequence \( L_{\nu_{gs}} \) is defined in a similar way to algorithm \texttt{track\_bands} in Section 5.5.
\[ \text{gs.bands}(s, v_o, v_i) \equiv \]
\[ V_0 := \text{gs.circle}_3D(s, v_o, v_i, -1, -1); \]
\[ V_1 := \text{gs.circle}_3D(s, v_o, v_i, -1, 1); \]
\[ V_2 := \text{gs.circle}_3D(s, v_o, v_i, 1, -1); \]
\[ V_3 := \text{gs.circle}_3D(s, v_o, v_i, 1, 1); \]
\[ \text{if } \|s(x,y)\| \geq D \text{ then} \]
\[ V_4 := \text{gs.circle}_2D(s, v_o, v_i, T, -1, -1); \]
\[ V_5 := \text{gs.circle}_2D(s, v_o, v_i, T, -1, 1); \]
\[ V_6 := \text{gs.line}(s, v_o, v_i, -1); \]
\[ V_7 := \text{gs.line}(s, v_o, v_i, -1); \]
\[ \text{endif} \]
\[ \mathcal{L} = \{\text{gsmin}, \text{gsmax}\}; \]
\[ \text{for } i = 1 \text{ to } |V| \text{ do} \]
\[ \text{if } V_i(x,y) \neq 0 \text{ and } \text{gsmin} \leq \|V_i(x,y)\| \leq \text{gsmax} \text{ then} \]
\[ \mathcal{L} := \mathcal{L} \cup \{\|V_i(x,y)\|\}; \]
\[ \text{endif} \]
\[ \text{endfor} \]
\[ L_{\nu_{gs}} := \text{sort}(\mathcal{L}); \]

**Theorem 36 (Correctness of gs.bands).** The ground speed prevention bands algorithm gs.bands is correct for \( \nu_{gs} \) over the interval \([\text{gsmin}, \text{gsmax}]\).

*Proof.* The first step in the proof is to consider the special case where \( \|s(x,y)\| \geq D \), \( v_i(x,y) \cdot v_o(x,y) = 0 \), and \( v_o(x,y) \cdot \text{tangent.line}(s, e)(x,y) = 0 \). This case is handled separately because it is explicitly excluded from the hypotheses of Theorem 31. In this case, it can be proved that the vectors \( \text{tangent.line}(s, e)(x,y), v_o(x,y), \) and \( v_i(x,y) \) are all co-linear.

To prove correctness of the algorithm for the special case, it suffices to show that if \( p \in [\text{gsmin}, \text{gsmax}] \), then \( \text{conflict?}(s, \nu_{gs}(p) - v_i) \) does not hold. Since \( \nu_{gs}(p) \) is a ground speed maneuver of \( v_o \), both vectors point in the same direction. Therefore, \( \nu_{gs}(p)(x,y) - v_i(x,y) \) is also co-linear with \( \text{tangent.line}(s, e)(x,y) \). The trajectory from \( s(x,y) \) along \( \nu_{gs}(p)(x,y) - v_i(x,y) \) is therefore tangent to the circle of radius \( D \) around the origin and is never in horizontal conflict.

In the general case, suppose that it is not true that \( \|s(x,y)\| \geq D \), \( v_i(x,y) \cdot v_o(x,y) = 0 \), and \( v_o(x,y) \cdot \text{tangent.line}(s, e)(x,y) = 0 \). In this case, by Theorem 2 in Section 2.5, it suffices to prove that the function \( \Omega_\nu \), where \( \nu = \nu_{gs} \), satisfies the following two properties.

1. For all ground speeds \( p \in [\text{gsmin}, \text{gsmax}] \),
\[ \Omega_\nu(p) < 1 \iff \text{conflict?}(s, \nu_{gs}(p) - v_i). \]
2. For all ground speeds \( p \in [gs_{\text{min}}, gs_{\text{max}}] \),

\[
\Omega_v(p) = 1 \implies p \in gs_{\text{bands}}(s, v_o, v_i).
\]

The first of these properties follows immediately from Corollary 7 in Section 3.3. All that is left to verify is the second property, the proof of which is identical in form and substance to the general case of the proof of Theorem 29 in Section 5.5.

7 Vertical Speed Prevention Bands

This section presents a formally verified algorithm, namely \( \text{vs\_bands} \), for vertical speed prevention bands over an arbitrary interval \([vs_{\text{min}}, vs_{\text{max}}]\) for the function \( v_{vs} : \mathbb{R} \rightarrow \mathbb{R}^3 \), defined by Equation (5) in Section 2.2. The boundaries of the interval, \( vs_{\text{min}} \) and \( vs_{\text{max}} \), represent minimum and maximum vertical speeds for the ownship aircraft, respectively. Given vectors \( s, v_o \), and \( v_i \), this algorithm computes vertical speed maneuvers, i.e., vectors \( v'_o \) that satisfy \( v'_o(x, y) = v_o(x, y) \).

The definition of \( \text{vs\_bands} \) depends on the algorithm \( \text{vs\_circle} \), which computes vertical speed maneuvers that are 3D circle solutions and vertical solutions. This algorithm is proved to be complete for vertical speed maneuvers. The correctness of \( \text{vs\_bands} \) depends on the completeness of \( \text{vs\_circle} \).

By the definition of \( \text{circle\_case\_3D} \) in Equation (21), \( \text{circle\_case\_3D}(s, v, t, i) \) implies \( \text{vertical\_case}(s_z, v_z, t, -i) \) and \( \text{circle\_case\_2D}(s, v, t, i) \) for any \( t \in \mathbb{R} \) and \( i = \pm 1 \). Thus, an algorithm for computing 3D circle solutions for vertical speed maneuvers will also compute vertical solutions and 2D circle solutions. If \( \text{vertical\_case}(s_z, v_z, t, -i) \) holds, then \( |s_z + tv_z| = H \), and therefore there is some \( \varepsilon = \pm 1 \) such that \( s_z + tv_z = \varepsilon H \). The function \( \text{vs\_at} \), defined below, takes as parameters \( s_z \), a nonzero real number \( t \), and \( \varepsilon = \pm 1 \). It returns the real number \( v_z \) such that \( s_z + tv_z = \varepsilon H \).

\[
\text{vs\_at}(s_z, t, \varepsilon) = e H - s_z.
\]

Lemma 37. If \( v_z \) is a real number, then \( s_z + tv_z = \varepsilon H \) if and only if

\[
v_z = \text{vs\_at}(s_z, t, \varepsilon).
\]

The next lemma states that the function \( \text{vs\_at} \) can be used to compute vertical solutions. The proof follows from Equation (19) and Lemma 37.

Lemma 38. If \( t > 0 \) and \( \text{vertical\_case}(s_z, v_z, t, -1) \) holds, then

\[
v_z = \text{vs\_at}(s_z, t, \text{sign}(s_z)).
\]

7.1 3D Circle and Vertical Solutions For Vertical Speed Maneuvers

The algorithm \( \text{vs\_circle} \), defined in this section, takes as parameters \( s, v_o, v_i, t, \) and \( \varepsilon = \pm 1 \). It returns a vector \( v'_o \) that is either the zero vector or is equal
to \( v_{vs}(r) \) for some \( r \in \mathbb{R} \) such that the relative vector \( v = v'_o - v_i \) satisfies circle_case_3D?(\( s, v, t, \epsilon \)). The main theorems in this section state that \( vs\_circle \) computes all 3D circle solutions and all vertical solutions that are vertical speed maneuvers.

Suppose that \( \Delta(s, v'_o - v_i) > 0 \) and circle_case_3D?(\( s, v'_o - v_i, t, \epsilon \)) holds, where \( v'_{o(x,y)} = v_{o(x,y)} \) and \( v'_{o(z)} = r \). It is easy to prove that \( \Delta(s, v_o - v_i) > 0 \) implies \( v_{o(x,y)} \neq v_i(x,y) \). Since \( \Delta(s, v_o - v_i) = \Delta(s, v'_o - v_i) \), Corollary 14 in Section 4.6 implies that \( t = \Theta_D(s, v_o - v_i, \epsilon) \). Since vertical_case?(\( s_z, (r - v_{i_z}), t, -\epsilon \)) holds, there is some \( \epsilon = \pm 1 \) such that

\[
s_z + \Theta_D(s, v_o - v_i, \epsilon) (r - v_{i_z}) = \epsilon H. \tag{53}
\]

Since \( \Theta_D(s, v_o - v_i, \epsilon) > 0 \), the following equivalence holds.

\[
r = v_{i_z} \iff H = \epsilon s_z.
\]

Suppose that \( H \neq \epsilon s_z \). Multiplying both sides of Equation (53) by \( \epsilon \) and applying the fact that \( \epsilon^2 = 1 \) yields

\[
\epsilon s_z + \Theta_D(s, v_o - v_i, \epsilon) (r - v_{i_z}) = H.
\]

Since \( \Theta_D(s, v_o - v_i, \epsilon) > 0 \), it follows that

\[
- \text{sign}(\epsilon (r - v_{i_z})) = \text{sign}(\epsilon s_z - H). \tag{54}
\]

Since vertical_case?(\( s_z, (r - v_{i_z}), \Theta_D(s, v_o - v_i, \epsilon), -\epsilon \)) holds,

\[
- \epsilon (s_z + \Theta_D(s, v_o - v_i, \epsilon) (r - v_{i_z})) (r - v_{i_z}) \geq 0.
\]

It therefore follows from Equation (53) that \( -\epsilon H (r - v_{i_z}) \geq 0 \). Since \( H > 0 \) and \( r \neq v_{i_z} \), basic arithmetic manipulations can be used to deduce that

\[
- \text{sign}(\epsilon (r - v_{i_z})) = \epsilon. \tag{55}
\]

Putting equations (54) and (55) together, the following equality holds.

\[
\text{sign}(\epsilon s_z - H) = \epsilon \tag{56}
\]

This equation is used to select the appropriate choice of \( \epsilon \) in the algorithm \( vs\_circle \), defined below in Equation (52), even in the case where \( \epsilon s_z = H \). It follows from Lemma 37 that

\[
r = v_{i_z} + vs\_at(s_z, \Theta_D(s, v_o - v_i, \epsilon), \epsilon).
\]

This equation also appears in the definition of \( vs\_circle \), which is given below. It returns a vector \( v'_o \in \mathbb{R}^3 \) such that either \( v'_o(x,y) = v_{o(x,y)} \) or \( v'_o = 0 \).
The completeness of vs\_circle for 3D circle and vertical solutions follows from its definition, Lemma 38, and properties of the function $\Theta_D$ presented in Section 4.6.

**Theorem 39 (Completeness of vs\_circle for 3D Circle Solutions).** If $\nu(x, y) = \nu(x, y)$, $\Delta(s, \nu - \nu_i) > 0$, $t = \pm 1$, $t > 0$, circle\_case\_3D?$(s, \nu'_o - \nu_i, t', t)$, for some $t' \in \mathbb{R}$, and either $\nu(s, \nu - \nu_i) < 0$ or $\Theta_D(s, \nu - \nu_i, 1) \leq t$, then

$$
\nu = \text{vs\_circle}(s, \nu, \nu_i, t, \varepsilon),
$$

for some $\varepsilon = \pm 1$.

**Theorem 40 (Completeness of vs\_circle for Vertical Solutions).** If $\nu'(x, y) = \nu_o(x, y)$, vertical\_case?$(s_z, \nu'_o z - \nu_i z, t, -1)$, and either $\Delta(s, \nu - \nu_i) \leq 0$ or $t \leq \Theta_D(s, \nu - \nu_i, 1)$, then

$$
\nu' = \text{vs\_circle}(s, \nu, \nu_i, t, \varepsilon),
$$

for some $\varepsilon = \pm 1$. 

38
7.2 A Prevention Bands Algorithm For Vertical Speed Maneuvers

The prevention bands algorithm \texttt{vs.bands} for the function \(\nu_{vs}: \mathbb{R} \rightarrow \mathbb{R}^3\) that computes a sorted sequence \(L_{\nu_{vs}}\) is defined in a similar way to the previous algorithms \texttt{track.bands} in Section 5.5 and \texttt{gs.bands} in Section 6.4.

\[
\texttt{vs.bands}(s, v_o, v_i) \equiv \\
V_0 := \texttt{vs.circle}(s, v_o, v_i, T, -1); \\
V_1 := \texttt{vs.circle}(s, v_o, v_i, T, 1); \\
\mathcal{L} = \{v_{smin}, v_{smax}\}; \\
\text{for } i = 1 \text{ to } |V| \text{ do} \\
\quad \text{if } V_{i(x,y)} \neq 0 \text{ and } v_{smin} \leq V_{i_z} \leq v_{smax} \text{ then} \\
\quad \quad \mathcal{L} := \mathcal{L} \cup \{V_{i_z}\}; \\
\text{endif} \\
\text{endfor} \\
L_{\nu_{vs}} := \text{sort}(\mathcal{L});
\]

**Theorem 41 (Correctness of \texttt{vs.bands}).** The vertical speed prevention bands algorithm \texttt{vs.bands} is correct for \(\nu_{vs}\) over the interval \([v_{smin}, v_{smax}]\).

**Proof.** By Theorem 2 in Section 2.5, it suffices to prove that the function \(\Omega_\nu\), where \(\nu = \nu_{vs}\), satisfies the following two properties.

1. For all vertical speeds \(r \in [v_{smin}, v_{smax}]\),
   \[\Omega_\nu(r) < 1 \iff \text{conflict?(s, } \nu_{vs}(r) - v_i)\).

2. For all vertical speeds \(r \in [v_{smin}, v_{smax}]\),
   \[\Omega_\nu(r) = 1 \implies r \in \texttt{vs.bands}(s, T, v_o, v_i)\).

The first of these properties follows immediately from Corollary 7 in Section 3.3. To prove the second property, suppose that \(r \in [v_{smin}, v_{smax}]\) and \(\Omega_\nu(r) = 1\), where \(\nu = \nu_{vs}\). Since \(\Omega_\nu(r) = \Omega(s, \nu_{vs}(r) - v_i)\), Theorem 9 implies that one of the following conditions holds, where \(\nu = \nu_{vs}(r) - v_i\).

- \(|s_{(x,y)}| \geq D\) and either \texttt{line.case?}(s, v, -1) or \texttt{line.case?}(s, v, 1).
- \(|s_z + T v_z| < H\) and \texttt{circle_case_2D?}(s, v, T, -1).
- There is some real number \(t > 0\) such that either \texttt{circle_case_3D?}(s, v, t, 1), for some \(t = \pm 1\).
- \(|s_{(x,y)} + T v_{(x,y)}| \leq D\) and \texttt{vertical_case?}(s_z, v_z, T, -1).

In either of the first two cases, it can easily be shown that \texttt{conflict?}(s, \nu_{vs}(x) - v_i) does not hold for any \(x \in \mathbb{R}\). In this case, by Definition 2 in Section 2.4, it follows that the prevention bands algorithm \texttt{vs.bands} is correct for \(\nu_{vs}\) over \([v_{smin}, v_{smax}]\).
The latter two cases are considered individually. For the rest of the proof, it is assumed that \( \Delta(s, v_o - v_i) > 0 \). The proof in the case where \( \Delta(s, v_o - v_i) \leq 0 \) is left to the reader. Since \( \Delta(s, v_o - v_i) > 0 \), it is easy to prove that \( v_o(x, y) \neq v_i(x, y) \).

Suppose that there is some real number \( t > 0 \) such that \( circle_case_3D?(s, v, t, 1', \alpha) \), where \( \alpha = \pm 1 \). By the definition of \( circle_case_3D? \) (Equation (21) in Section 4.3) and Corollary 14 in Section 4.6, it follows that \( t = \Theta_D(s, v_o - v_i, \alpha) \). By Theorem 39 in Section 7.1 (completeness of \( vs.circle \) for 3D circle solutions), if either \( \alpha = -1 \) or \( \Theta_D(s, v_o - v_i, \alpha) \leq T \), then \( \nu_{vs}(r) \) is equal to \( vs.circle(s, v_o, v_i, T, \epsilon) \), for some \( \epsilon = \pm 1 \). Thus, \( r \in vs.bands(s, T, v_o, v_i) \). Alternatively, if \( \alpha = 1 \) and \( \Theta_D(s, v_o - v_i, 1) > T \), it can be proved from the definition of the function \( \Omega_{\nu} \) that \( \Omega_{\nu}(r) > 1 \), a contradiction.

Finally, suppose that \( \|s(x, y) + Tv(x, y)\| \leq D \) and \( vertical_case?(s, v_z, T, -1) \). The proof in this case is similar to the case above. By Theorem 40 in Section 7.1 (completeness of \( vs.circle \) for vertical solutions), if \( T \leq \Theta_D(s, v_o - v_i, 1) \), then \( \nu_{vs}(r) \) is equal to \( vs.circle(s, v_o, v_i, T, \epsilon) \), for some \( \epsilon = \pm 1 \). Thus, it holds that \( r \in vs.bands(s, T, v_o, v_i) \). Alternatively, if \( T > \Theta_D(s, v_o - v_i, 1) \), it can be proved that \( \Omega_{\nu}(r) > 1 \), a contradiction. \( \square \)

8 Conclusion

In [3], Maddalon et al. present, without verification, 3D algorithms for track angle, ground speed, and vertical speed prevention bands. Formal verification of horizontal versions of these algorithms was presented in [2]. This paper provides correct versions of the algorithms presented in [3], namely \( track.bands \) (Section 5.5), \( gs.bands \) (Section 6.4), and \( vs.bands \) (Section 7.2). The correctness of these algorithms has been formally verified using the PVS theorem prover.

Although this paper focuses on track angle, ground speed, and vertical speed prevention bands, the techniques presented here applied to arbitrary conflict prevention bands algorithms that are based on state information. More precisely, given a function \( \nu: \mathbb{R} \rightarrow \mathbb{R}^3 \) and an interval \( I \subset \mathbb{R} \), Section 2.5 describes a general strategy that can be followed to prove that a given prevention bands algorithm is correct. In fact, Section 3 develops the theory of a universal function \( \Omega \) that can be used as a tool in the verification of prevention bands algorithms for many different choices of \( \nu \).

References


Formal Verification of Air Traffic Conflict Prevention Bands Algorithms

In air traffic management, a pairwise conflict is a predicted loss of separation between two aircraft, referred to as the ownship and the intruder. A conflict prevention bands system computes ranges of maneuvers for the ownship that characterize regions in the airspace that are either conflict-free or 'don't go' zones that the ownship has to avoid. Conflict prevention bands are surprisingly difficult to define and analyze. Errors in the calculation of prevention bands may result in incorrect separation assurance information being displayed to pilots or air traffic controllers. This paper presents provably correct 3-dimensional prevention bands algorithms for ranges of track angle, ground speed, and vertical speed maneuvers. The algorithms have been mechanically verified in the Prototype Verification System (PVS). The verification presented in this paper extends in a non-trivial way that of previously published 2-dimensional algorithms.