Coordination Logic for Repulsive Resolution Maneuvers

Anthony Narkawicz   César A. Muñoz   Aaron Dutle

NASA Langley Research Center, Hampton, VA, 23881, US

This paper presents an algorithm for determining the direction an aircraft should maneuver in the event of a potential conflict with another aircraft. The algorithm is implicitly coordinated, meaning that with perfectly reliable computations and information, it will independently provide directional information that is guaranteed to be coordinated without any additional information exchange or direct communication. The logic is inspired by the logic of TCAS II, the airborne system designed to reduce the risk of mid-air collisions between aircraft. TCAS II provides pilots with only vertical resolution advice, while the proposed algorithm, using a similar logic, provides implicitly coordinated vertical and horizontal directional advice.

Nomenclature

Dlim	Desired horizontal separation at time of closest point of approach
Dcol	Minimum horizontal separation
ϵ	Directionality parameter whose value is ± 1
$\mathbf{s}$	Two-dimensional aircraft position
t	Time variable
t_{stp}	Time step for piecewise linear maneuvers
$\mathbf{v}$	Two-dimensional aircraft velocity
β	Bank angle
Subscripts	
o	Ownship state
i	Intruder state
x	Northern component of a position or velocity vector
y	Eastern component of a position or velocity vector

I.   Introduction

The *Traffic Collision Avoidance System* (TCAS) is a family of airborne devices that are designed to reduce the risk of mid-air collisions between aircraft with operating transponders.[8] TCAS II is the second generation of TCAS devices and is currently mandated in the U.S. for aircraft with greater than 30 seats or a maximum takeoff weight greater than 33,000 pounds. Although it is not required, TCAS II is also installed on many turbine-powered general aviation aircraft. In contrast to the first generation of TCAS devices, TCAS II provides *resolution advisories* (RAs) to pilots. RAs are visual and vocalized alerts that direct pilots to maintain or increase vertical separation with intruders that are collision threats. While explicit communication may occur between two aircraft using TCAS II, in many cases the RA logic exhibits *implicit coordination* in the sense that resolution maneuvers in opposite directions will be computed by the TCAS II RA algorithm using the aircraft state information. In previous work,[5] the core RA logic of TCAS II was formalized in a mathematical model that assumes accurate state information for the two aircraft in the form of position and velocity vectors.

This paper presents an algorithm that provides directional guidance for resolving impending air traffic conflicts such as loss of separation, as in the case of conflict resolution systems,[3] well-clear violation, as in the case of detect and avoid systems for Unmanned Aircraft Systems,[1] or near-miss, as in the case of

collision avoidance systems.[8] The logic of this algorithm is inspired by the RA logic of TCAS II. However, the proposed logic is more complicated than simply replacing the notions of up/down in TCAS II with the notions of right/left. For instance, some situations require both aircraft to turn right, such as when the aircraft are approaching head-on, and others require them to turn in different directions, such as when one aircraft is overtaking the other. Thus, instead of using the notions of absolute right/left, this paper uses the notions of right/left in the *relative* frame of reference for each aircraft playing the role of *ownship*, where the other aircraft, i.e., the *intruder*, is viewed as being fixed at the origin. A function is presented in Section V.I for computing a relative direction for horizontal resolution maneuvers, which returns the same answer for both aircraft. Section V.J defines an algorithm for computing an actual maneuver for each aircraft that corresponds to this direction in the relative frame of reference.

II.   Mathematical Notation

This paper assumes that accurate aircraft surveillance information is available as horizontal and vertical components in a three-dimensional airspace. Letters in **bold-face** denote two-dimensional (2-D) vectors. Vector operations such as addition, subtraction, scalar multiplication, dot product, i.e., $\mathbf{s} \cdot \mathbf{v} = \mathbf{s}_x \mathbf{v}_x + \mathbf{s}_y \mathbf{v}_y$, the square of a vector, i.e., $\mathbf{s}^2 = \mathbf{s} \cdot \mathbf{s}$, and the norm of a vector, i.e., $\|\mathbf{s}\| = \sqrt{\mathbf{s}_x^2 + \mathbf{s}_y^2}$, are defined in a 2-D Euclidean geometry. Furthermore, the expression $\mathbf{v}^\perp$ denotes the 2-D right perpendicular of $\mathbf{v}$, i.e., $\mathbf{v}^\perp = (\mathbf{v}_y, \mathbf{v}_x)$, and $\mathbf{0}$ denotes the 2-D vector whose components are 0, i.e., $\mathbf{0} = (0, 0)$. Further, $\mathtt{sign}$ is the function that for any real number x, $\mathtt{sign}(x) = 1$ when $x \geq 0$ and $\mathtt{sign}(x) = -1$ when $x < 0$.

The mathematical models presented in this paper consider two aircraft referred to as the ownship and the intruder. For the ownship, the current horizontal position and velocity are denoted $\mathbf{s}_o$ and $\mathbf{v}_o$, respectively. Its altitude and vertical speed are denoted s_{oz} and v_{oz}, respectively. Similarly, the horizontal position and velocity of the intruder aircraft are denoted $\mathbf{s}_i$ and $\mathbf{v}_i$, respectively, and its vertical altitude and speed are denoted s_{iz} and v_{iz}, respectively. As it simplifies the mathematical development, this paper uses a relative coordinate system where the intruder is static at the center of the coordinate system. In this relative system, $\mathbf{s} = \mathbf{s}_o - \mathbf{s}_i$ and $\mathbf{v} = \mathbf{v}_o - \mathbf{v}_i$ represent the horizontal relative position and velocity of the aircraft, respectively. Furthermore, $s_z = s_{oz} - s_{iz}$ and $v_z = v_{oz} - v_{iz}$ represent the vertical relative position and speed of the aircraft, respectively. Assuming constant relative horizontal velocity $\mathbf{v}$, the horizontal range between the aircraft at any time t is given by

$$r(t) = \|\mathbf{s} + t\mathbf{v}\| = \sqrt{\mathbf{s}^2 + 2t(\mathbf{s}\mathbf{v}) + t^2\mathbf{v}^2}. \tag{1}$$

The time of horizontal closest point of approach, denoted t_{cpa}, is the time t that satisfies $\dot{r}(t) = 0$. A quick calculations shows that when $\mathbf{v} \neq \mathbf{0}$ this occurs at time $t = -\mathbf{s} \cdot \mathbf{v}/\mathbf{v}^2$. The dot product $\mathbf{s} \cdot \mathbf{v}$ characterizes whether the aircraft are horizontally diverging, when $\mathbf{s} \cdot \mathbf{v} \geq 0$, or horizontally converging, when $\mathbf{s} \cdot \mathbf{v} < 0$. By convention, t_{cpa} is defined to be 0 when $\mathbf{v} = \mathbf{0}$. Hence, $t_{\mathrm{cpa}}(\mathbf{s}, \mathbf{v})$ is formally defined as $-\mathbf{s} \cdot \mathbf{v}/\mathbf{v}^2$ when $\mathbf{v} \neq \mathbf{0}$ and 0 otherwise.

III.   Generalizing TCAS II Resolution Advisory Logic

This section presents a model of the TCAS II resolution advisory logic and its coordination properties. Then, it presents a slightly modified version of this logic that produces, in most geometries, implicitly coordinated directions for two aircraft. Finally, it presents a function $\mathtt{dir\_select}$ that generalizes this modified TCAS II RA logic and computes relative maneuver directions for both horizontal and vertical maneuvers. It has functions as parameters, and different choices of these functions result in different resolution logics. Later, in Sections V.I and V.J, this logic will be used to define the algorithm $\mathtt{horiz\_rel\_dir}$, which determines a *relative* horizontal direction for maneuvering, and the algorithm $\mathtt{horiz\_coord\_man}$, which determines an actual maneuver for each aircraft corresponding to this relative direction.

III.A.   TCAS II Resolution Advisory Logic

When the TCAS system detects a collision threat, it issues a resolution advisory. It then calculates a resolution direction, either upward or downward. This direction is then used to issue an instruction to climb, descend, not climb, or not descend. The logic that computes this direction is the inspiration for the algorithm presented in this paper for determining appropriate horizontal/vertical maneuvers to resolve

American Institute of Aeronautics and Astronautics

impending conflicts. The TCAS II direction selection logic is described in a previous work by the authors.[5] For completeness, this logic is also described in this section.

The TCAS II RA logic selects a direction by considering the vertical separation of the aircraft at a future time τ, which is a function of $\mathbf{s}$ and $\mathbf{v}$. This time is an estimate of the time of closest point of approach. It is defined as the current range r divided by the range rate $\dot{r}$, and is therefore given by the following formula

$$\tau(\mathbf{s}, \mathbf{v}) \;\equiv\; -\frac{\mathbf{s} \cdot \mathbf{v}}{\mathbf{s}^2}.$$

When $t_{\mathrm{cpa}}(\mathbf{s}, \mathbf{v})$ is large, $\tau(\mathbf{s}, \mathbf{v})$ is a reasonably good over-approximation of $t_{\mathrm{cpa}}(\mathbf{s}, \mathbf{v})$. In contrast to t_{cpa}, the computations of τ can be done without directional information.

The function $\mathtt{sep\_at}$, defined by Formula (2), predicts the vertical separation between the aircraft at time τ assuming a target vertical speed v for the ownship. The ownship is assumed to fly at constant ground speed and constant vertical acceleration a. Once the target vertical speed v is reached the ownship continues to fly at constant vertical speed. The function $\mathtt{own\_alt\_at}$, defined by Formula (3), computes the vertical altitude of the ownship at any time t given a target vertical speed v and acceleration a. The parameter ϵ specifies a possible direction for the vertical ownship maneuver, which is *upward* when $\epsilon = 1$ and *downward* when $\epsilon = -1$. The intruder is assumed to continue its trajectory at its current vertical speed.

$$
\begin{aligned}
&\mathtt{sep\_at}(\mathbf{s}, \mathbf{v}, s_{oz}, v_{oz}, s_{iz}, v_{iz}, v, a, \epsilon) \equiv \\
&\qquad \texttt{let } alt_o = \mathtt{own\_alt\_at}(s_{oz}, v_{oz}, |v|, a, \epsilon\, \mathtt{sign}(v), \tau(\mathbf{s}, \mathbf{v})), \\
&\qquad\quad alt_i = s_{iz} + \tau(\mathbf{s}, \mathbf{v}) \cdot v_{iz} \texttt{ in} \\
&\qquad \epsilon\,(alt_o - alt_i).
\end{aligned}
\tag{2}
$$

$$
\begin{aligned}
&\mathtt{own\_alt\_at}(s_{oz}, v_{oz}, v, a, \epsilon, t) \equiv \\
&\qquad \texttt{let } s = \mathtt{stop\_accel}(v_{oz}, v, a, \epsilon, t), \\
&\qquad\quad q = \min(t, s), \\
&\qquad\quad l = \max(0, t - s) \texttt{ in} \\
&\qquad \epsilon\, q^2\,\frac{a}{2} + q v_{oz} + s_{oz} + \epsilon\, l\, v.
\end{aligned}
\tag{3}
$$

The function $\mathtt{stop\_accel}$ computes the time at which the ownship reaches the target vertical speed v. It is defined as follows.

$$
\begin{aligned}
&\mathtt{stop\_accel}(v_{oz}, v, a, \epsilon, t) \equiv \\
&\qquad \texttt{if } t \leq 0 \texttt{ or } \epsilon\, v_{oz} \geq v \texttt{ then } 0 \\
&\qquad \texttt{else } \frac{\epsilon\, v - v_{oz}}{\epsilon\, a} \\
&\qquad \texttt{endif.}
\end{aligned}
\tag{4}
$$

The sense of an RA is computed based on the direction for the ownship maneuver that provides a greater vertical separation, with a bias towards the non-crossing direction. The function $\mathtt{RA\_sense}$ computes such a direction, where $\mathtt{ALIM}_\ell$ is the altitude limit for a given sensitivity level ℓ.

$$
\begin{aligned}
&\mathtt{RA\_sense}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, v, a) \equiv \\
&\qquad \texttt{let } o_\uparrow = \mathtt{own\_alt\_at}(s_{oz}, v_{oz}, v, a, 1, \tau(\mathbf{s}, \mathbf{v})), \\
&\qquad\quad o_\downarrow = \mathtt{own\_alt\_at}(s_{oz}, v_{oz}, v, a, -1, \tau(\mathbf{s}, \mathbf{v})), \\
&\qquad\quad alt_i = s_{iz} + \tau(\mathbf{s}, \mathbf{v}) v_{iz}, \\
&\qquad\quad \mathtt{dist}_{o\uparrow} = o_\uparrow - alt_i, \\
&\qquad\quad \mathtt{dist}_{o\downarrow} = alt_i - o_\downarrow \texttt{ in} \\
&\qquad \texttt{if } \mathtt{sign}(s_{oz} - s_{iz}) = 1 \texttt{ and } \mathtt{dist}_{o\uparrow} \geq \mathtt{ALIM}_\ell \texttt{ then } 1 \\
&\qquad \texttt{elsif } \mathtt{sign}(s_{oz} - s_{iz}) = -1 \texttt{ and } \mathtt{dist}_{o\downarrow} \geq \mathtt{ALIM}_\ell \texttt{ then } -1 \\
&\qquad \texttt{elsif } \mathtt{dist}_{o\uparrow} \geq \mathtt{dist}_{o\downarrow} \texttt{ then} 1 \\
&\qquad \texttt{else } -1 \\
&\qquad \texttt{endif.}
\end{aligned}
\tag{5}
$$

American Institute of Aeronautics and Astronautics

To understand how the function `RA_sense` selects the direction, consider a situation where the ownship is currently above the intruder. If an upward maneuver will provide at least $\mathtt{ALIM}_\ell$ of vertical separation above the other aircraft at time $\tau(\mathbf{s}, \mathbf{v})$, then an upward maneuver, denoted by the numerical value 1, is returned. If this is not the case and a downward maneuver will provide $\mathtt{ALIM}_\ell$ separation at $\tau(\mathbf{s}, \mathbf{v})$, then a downward maneuver, denoted by the numerical value -1, is returned. Otherwise, it returns the direction that will provide the larger vertical separation at $\tau(\mathbf{s}, \mathbf{v})$.

III.B.   Coordination in TCAS II

For some geometries, the function `RA_sense` is implicitly coordinated in the sense that the two aircraft will pick opposite, i.e., coordinated, directions for their resolution maneuvers. This is not true in every geometry, however, such as when one of the aircraft already has a vertical speed greater than the maximum vertical speed v. Because of this, TCAS II, after computing direction, communicates this information to the other aircraft, and an explicit decision is made for coordinating maneuver directions.

It is possible to modify the algorithm `RA_sense` in a way that makes it implicitly coordinated for a wider range of geometries assuming reliable state information. The intuition behind this modification is that both aircraft have all information needed to compute a coordinated direction, namely the explicit state information $\mathbf{s}_o, \mathbf{v}_o, \mathbf{s}_i, \mathbf{v}_i, s_{oz}, v_{iz}$. That is, if it is possible for the two aircraft to decide between themselves how to coordinate directions, that selection logic can be computed independently on each aircraft, given that this explicit state information gives them near-perfect knowledge of the situation.

The function `RA_sense_new`, defined below, is one example of a vertical direction selection inspired on the TCAS II RA direction selection logic as that guarantees implicit coordination in a wider range of geometries because it considers the situation from each aircraft's viewpoint before deciding.

$$
\begin{aligned}
&\mathtt{RA\_sense\_new}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, v, a) \equiv \\
&\qquad \mathtt{let}\ \ o_\uparrow = \mathtt{own\_alt\_at}(s_{oz}, v_{oz}, v, a, 1, \tau(\mathbf{s}, \mathbf{v})), \\
&\qquad\qquad o_\downarrow = \mathtt{own\_alt\_at}(s_{oz}, v_{oz}, v, a, -1, \tau(\mathbf{s}, \mathbf{v})), \\
&\qquad\qquad i_\uparrow = \mathtt{own\_alt\_at}(s_{iz}, v_{iz}, v, a, 1, \tau(\mathbf{s}, \mathbf{v})), \\
&\qquad\qquad i_\downarrow = \mathtt{own\_alt\_at}(s_{iz}, v_{iz}, v, a, -1, \tau(\mathbf{s}, \mathbf{v})), \\
&\qquad\qquad alt_o = s_{oz} + \tau(\mathbf{s}, \mathbf{v}) v_{oz}, \\
&\qquad\qquad alt_i = s_{iz} + \tau(\mathbf{s}, \mathbf{v}) v_{iz}, \\
&\qquad\qquad \mathtt{dist}_{o\uparrow} = o_\uparrow - alt_i, \\
&\qquad\qquad \mathtt{dist}_{o\downarrow} = alt_i - o_\downarrow \\
&\qquad\qquad \mathtt{dist}_{i\uparrow} = i_\uparrow - alt_o, \\
&\qquad\qquad \mathtt{dist}_{i\downarrow} = alt_o - i_\downarrow\ \ \mathtt{in} \\
&\qquad \mathtt{if}\ \mathtt{sign}(s_{oz} - s_{iz}) = 1\ \mathtt{and}\ \min(\mathtt{dist}_{o\uparrow}, \mathtt{dist}_{i\downarrow}) \geq \mathtt{ALIM}_\ell\ \mathtt{then}\ 1 \\
&\qquad \mathtt{elsif}\ \mathtt{sign}(s_{oz} - s_{iz}) = -1\ \mathtt{and}\ \min(\mathtt{dist}_{o\downarrow}, \mathtt{dist}_{i\uparrow}) \geq \mathtt{ALIM}_\ell\ \mathtt{then}\ -1 \\
&\qquad \mathtt{elsif}\ \min(\mathtt{dist}_{o\uparrow}, \mathtt{dist}_{i\downarrow}) \geq \min(\mathtt{dist}_{o\downarrow}, \mathtt{dist}_{i\uparrow})\ \mathtt{then}\ 1 \\
&\qquad \mathtt{else}\ -1 \\
&\qquad \mathtt{endif}.
\end{aligned}
\tag{6}
$$

To understand how the function `RA_sense_new` works, consider a situation where the ownship is currently above the intruder. The algorithm considers the separation at $\tau(\mathbf{s}, \mathbf{v})$ if the ownship goes up and the intruder continues linearly. It then considers the separation if the intruder goes down and the ownship continues linearly. If the minimum of these is at least $\mathtt{ALIM}_\ell$, then an upward direction, represented by the numerical value 1, is chosen for the ownship and a downward direction, represented by the numerical value -1, is chosen for the intruder. Otherwise, a similar minimum is computed where the ownship is assumed to go down or the intruder is assumed to go up. If that minimum is at least $\mathtt{ALIM}_\ell$, then the ownship selects the downward direction, represented by the numerical value -1, and the intruder selects the upward direction, represented by the numerical value 1. If not, then the two minimums are compared and the greater one determines the direction selection. While not exactly the TCAS II RA direction selection logic, this function is very similar and guarantees implicit coordination in a greater number of geometries.

American Institute of Aeronautics and Astronautics

III.C. Generalizing TCAS Direction Logic

This section generalizes the function `RA_sense_new`, which is inspired by the TCAS II direction selection logic. The generalization is an algorithm that can be used to compute resolution directions for vertical, horizontal, or even other types of maneuvers. This is because the following functions are parameters to this algorithm.

- The function that computes the preferred direction, e.g., $\texttt{sign}(s_{oz} - s_{iz})$ for `RA_sense_new`,

- The function that computes distance at a given time, e.g., vertical distance at time τ for `RA_sense_new`.

For vertical maneuvers, the notion of direction selection for coordinating maneuvers is fairly easy to understand because it is clear that one aircraft should go up, while the other should go down. The reason is that an upward (resp. downward) maneuver in the relative frame of reference is also an upward (resp. downward) maneuver in the absolute frame of reference. This is not so for horizontal maneuvers. A right (resp. left) maneuver in the relative frame of reference does not necessarily correspond to a right (resp. left) maneuver in the absolute frame of reference.

Another difference between coordinating vertical maneuvers and coordinating horizontal maneuvers is that coordinated vertical maneuvers tend to be in *opposite* directions (one up and the other down), while coordinated horizontal maneuvers are often in the *same* direction. For instance, in a head-on conflict, the right-of-way air traffic rules state that each aircraft should alter course to its *right*.[2,9]

Thus, in the general algorithm presented in this section, namely `dir_select`, the function that computes the preferred direction is generic. For vertical maneuvers, it will typically return the opposite direction if $\mathbf{s}_o, \mathbf{v}_o, s_{oz}, v_{oz}$ are swapped with $\mathbf{s}_i, \mathbf{v}_i, s_{iz}, v_{iz}$, while for horizontal maneuvers it will return the same direction. This direction for horizontal maneuvers is understood as being in the relative frame of reference, a concept that is explained in detail in Section V. This algorithm uses the following generic functions.

- $\texttt{eps\_choice}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz})$: returns a relative direction -1 or 1

- $\texttt{eps\_to\_dist}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \epsilon)$: returns a distance corresponding to a maneuver associated with direction ϵ. This can be thought of as the worst case scenario separation distance if this direction ϵ is chosen for maneuvering.

As shown in the algorithm `RA_sense_new` above, the choice of `eps_to_dist` is the output of `own_alt_at` (with the appropriate choice of ϵ) *minus* the altitude $s_{iz} + \tau(\mathbf{s}, \mathbf{v})v_{iz}$. Thus, the output of `eps_to_dist` can be *negative* if the difference in altitudes at the time $\tau(\mathbf{s}, \mathbf{v})$ is in the wrong direction. For instance, when the ownship is currently above the intruder and $\epsilon = -1$ is considered in the function `eps_to_dist`, if the ownship's downward maneuver will place it above the intruder at $\tau(\mathbf{s}, \mathbf{v})$, then this distance will be negative. Similarly, as described in Section V, for horizontal maneuvers, a maneuver corresponding to a particular ϵ value that is to the wrong side of the intruder (in the relative frame of reference) at t_{cpa} will produce a negative distance when evaluated by `eps_to_dist`. The algorithm `dir_select` is defined as follows.

$$
\begin{aligned}
&\texttt{dir\_select}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}) \equiv \\
&\qquad \texttt{let } \epsilon_o = \texttt{eps\_choice}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}), \\
&\qquad\qquad \epsilon_i = \texttt{eps\_choice}(\mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}), \\
&\qquad\qquad \texttt{dist}_o = \texttt{eps\_to\_dist}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \epsilon_o), \\
&\qquad\qquad \texttt{dist}_i = \texttt{eps\_to\_dist}(\mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \epsilon_i), \\
&\qquad\qquad \texttt{dist}_o^* = \texttt{eps\_to\_dist}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, -\epsilon_o), \\
&\qquad\qquad \texttt{dist}_i^* = \texttt{eps\_to\_dist}(\mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, -\epsilon_i), \\
&\qquad\qquad \texttt{min\_dist} = \min(\texttt{dist}_o, \texttt{dist}_i), \\
&\qquad\qquad \texttt{min\_dist}^* = \min(\texttt{dist}_o^*, \texttt{dist}_i^*) \texttt{ in} \\
&\qquad \texttt{if min\_dist} \geq \texttt{Dlim then } \epsilon_o \\
&\qquad \texttt{elsif min\_dist}^* \geq \texttt{Dlim then } -\epsilon_o \\
&\qquad \texttt{elsif min\_dist} \geq \min(\texttt{min\_dist}^*, \texttt{Dcol}) \texttt{ then } \epsilon_o \\
&\qquad \texttt{elsif min\_dist}^* \geq \texttt{Dcol then } -\epsilon_o \\
&\qquad \texttt{else } 0 \\
&\qquad \texttt{endif}.
\end{aligned}
\tag{7}
$$

The algorithm `dir_select` uses the parametric constants `Dlim` and `Dcol` that represent distances. The distance `Dlim` is analogous the distance $\mathtt{ALIM}_\ell$ in the TCAS II RA direction selection function `RA_sense`. If the primary choices of epsilon for the aircraft, namely ϵ_o and ϵ_i, both produce at least `Dlim` of separation via the function `eps_to_dist`, then that epsilon value is returned. If the opposite choices of epsilon, $-\epsilon_o$ and $-\epsilon_i$, produce at least `Dlim` of separation, then those values are returned. If not, then the minimum distance, comparing the ownship and intruder, is computed for each of these possible pairs of return values. The function returns the epsilon for which the minimum distance is *largest*, assuming that it is at least `Dcol`. If it is not at least `Dcol`, then 0 is returned, meaning that there is no solution that guarantees sufficient separation. The distance `Dcol` is therefore used as an absolute minimum acceptable distance for resolutions, and if it is not possible to coordinate maneuvers that each guarantee this separation when the other aircraft is assumed not to maneuver, then no direction is returned, which is represented by the value 0.

The algorithm `dir_select` exhibits implicit coordination, as stated in the next theorem.

Theorem 1. *Let*

- $epso = \mathit{dir\_select}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz})$,

- $epsi = \mathit{dir\_select}(\mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz})$,

- $\epsilon_o = \mathit{eps\_choice}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz})$, *and*

- $\epsilon_i = \mathit{eps\_choice}(\mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz})$.

Suppose that neither epso nor epsi is 0. *Then either both aircraft pick the preferred epsilon, epso* $= \epsilon_o$ *and epsi* $= \epsilon_i$, *or both aircraft pick the negation of the preferred epsilon, epso* $= -\epsilon_o$ *and epsi* $= -\epsilon_i$.

For vertical maneuvers, the function `eps_choice` will usually be chosen to return negated values when computed for the ownship versus the intruder. Conversely, for horizontal maneuvers, it will usually be chosen to return the same value for the ownship and intruder. The next corollary shows that the same is true for the function `dir_select`.

Corollary 2. *Using the notation of Theorem 1, suppose that neither epso nor epsi is* 0. *If* $\epsilon_o = \epsilon_i$, *then epso* $=$ *epsi, and if* $\epsilon_o = -\epsilon_i$, *then epso* $= -$*epsi.*

IV. Vertical Direction Selection With `dir_select`

The vertical direction selection algorithm `RA_sense_new`, which is inspired by the direction selection algorithm of the TCAS II RA logic, can be defined as a specific instance of the general algorithm `dir_select`. Let

$$\mathtt{tcas\_vert\_eps}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}) = \mathtt{sign}(s_{oz} - s_{iz})$$

and define the function

$$
\begin{aligned}
\mathtt{tcas\_vert\_dist}(\mathbf{s}_o, &s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \epsilon) \equiv \\
&\mathtt{let}\ \ alt_o = \mathtt{own\_alt\_at}(s_{oz}, v_{oz}, v, a, \epsilon, \tau(\mathbf{s}, \mathbf{v})), \\
&\qquad alt_i = s_{iz} + \tau(\mathbf{s}, \mathbf{v}) \cdot v_{iz}\ \ \mathtt{in} \\
&\quad \epsilon \cdot (alt_o - alt_i).
\end{aligned}
\tag{8}
$$

Therefore,

$$\mathtt{RA\_sense\_new}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, v, a) = \mathtt{dir\_select}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}),$$

where `eps_choice` is `tcas_vert_eps`, `eps_to_dist` is `tcas_vert_dist`, `Dlim` is $\mathtt{ALIM}_\ell$, and `Dcol` is a negative distance that is smaller than any possible value of `tcas_vert_dist`, e.g., $-70,000$ ft.

V. Resolution Logic for Horizontal Maneuvers

Section IV shows that the vertical resolution direction logic in the function `RA_sense_new` can be implemented using the general algorithm `dir_select`. In fact, a similar algorithm for horizontal maneuver selection can be implemented using `dir_select` as well. To define a horizontal maneuver algorithm using `dir_select`, instantiations are needed for the ϵ-choosing-function `eps_choice` and the distance function `eps_to_dist`.

American Institute of Aeronautics and Astronautics

V.A. Horizontal Relative Direction Parameters

In the horizontal frame of reference, a sign ϵ corresponds to direction in the relative frame of reference. Figure 1 illustrates velocity vectors that will cause the ownship to go to the right/left of the intruder, in the relative frame of reference. It is actually possible for a right turn, in the absolute frame of reference, to result in a left turn in the relative frame. Whether a given relative velocity $\mathbf{v}'$ goes to the right/left of the
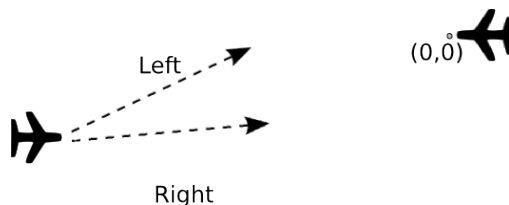
Figure 1. Right and Left in the Relative Frame of Reference

intruder in the relative frame of reference is determined by the sign

$$\epsilon_0 = -\texttt{sign}(\mathbf{s} \cdot \mathbf{v}'^{\perp}), \tag{9}$$

where, as above, $\mathbf{s} = \mathbf{s}_o - \mathbf{s}_i$. If $\epsilon_0 = -1$, then the relative velocity $\mathbf{v}'$ goes to the *right* of the origin in the relative frame, if $\epsilon_0 = 1$, it goes to the left. This formula is therefore used to determine direction (left or right of the intruder in the relative frame) of any given relative velocity vector.

V.B. Choosing the Preferred Horizontal ϵ

For choosing the preferred horizontal direction parameter ϵ, the current direction is chosen as the preferred choice of ϵ in the function `dir_select`:

$$\texttt{horiz\_eps\_choice}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}) = -\texttt{sign}((\mathbf{s}_o - \mathbf{s}_i) \cdot (\mathbf{v}_o - \mathbf{v}_i)^{\perp})$$

The function `horiz_eps_choice` will be used in place of the parameter `eps_choice` in the function `dir_select`.

V.C. Approach to Computing a Distance for Each ϵ

Before presenting the actual formula for computing a distance for each ϵ, this section describes the basic approach. Recall the vertical direction selection algorithm described in Section IV, where the distance for each ϵ was chosen by considering a single maneuver (either upward or downward depending on ϵ) and computing the vertical distance at $\tau(\mathbf{s}, \mathbf{v})$ for that maneuver. Computing a distance for a given horizontal (relative) direction parameter ϵ is more difficult. In this case, *two* maneuvers are considered, one for the ownship and one for the intruder, each corresponding to ϵ in the relative frame of reference when the other aircraft does not maneuver. Each of these maneuvers is given by a constant-bank turn, and the bank angles are commonly known to both aircraft. The ownship computes three distances and takes the minimum of the three: (1) The distance at closest point of approach, assuming the intruder continues linearly and the ownship maneuvers, (2) The distance at closest point of approach, assuming the ownship continues linearly and the intruder maneuvers, and (3) The distance at closest point of approach if both aircraft choose maneuvers corresponding to the relative direction parameter ϵ. The minimum of these three gives a *worst-case* minimum separation distance if this particular value of ϵ is chosen for maneuvering. This worse case separation distance is computed for both possible values of ϵ (-1 or 1). If for the preferred value of ϵ (Section V.B), this value is greater than predetermined threshold `Dlim`, then the preferred ϵ value is chosen by both aircraft for maneuvering. Otherwise, the value of ϵ is chosen such that its worst case separation distance is *greatest*. The details of this approach are presented in later sections.

V.D. Approach to Choosing Candidate Maneuvers

As noted above, for each ϵ (-1 or 1) representing a direction in the relative frame of reference, two maneuvers are computed, one for the ownship and another for the intruder. Each maneuver is of the same type, a circular

American Institute of Aeronautics and Astronautics

arc representing the trajectory of a banked turn with a constant bank angle, as illustrated in Figure 2. The formula for a banked turn is well-known. This is modeled inside the resolution algorithm as a *piecewise*
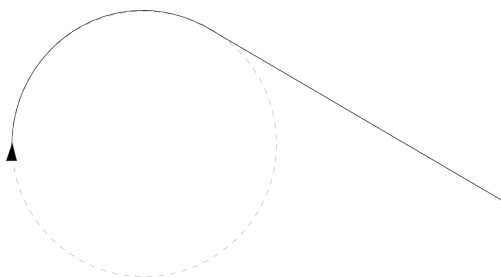

Figure 2. Banked Turn and Linear Segment

linear turn followed by a linear segment, as illustrated in Figure 3. To ensure that the turn results in a
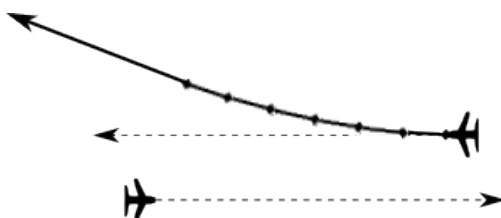

Figure 3. Piecewise Linear Model of a Kinematic Turn

maneuver in the relative frame corresponding to the correct value of ϵ, it is checked that the velocity on each small piecewise linear segment on the turn is *repulsive* (in the ϵ relative direction) with respect to the velocity on the previous segment. This means that each segment moves the trajectory of the aircraft further in the ϵ-direction, which is determined analytically using the *repulsive criterion* described below. Once a segment is found whose velocity does not satisfy the repulsive criterion with respect to the previous velocity, the maneuver continues linearly at the velocity on the previous segment.

V.E. Repulsive Criterion for Instantaneous Velocity Changes

Given -1 or 1, describing left or right in the relative frame of reference, the formula that determines whether a new relative velocity $\mathbf{v}'$ corresponds to that value, with respect to the current positions $\mathbf{s}_o$, $\mathbf{s}_i$ and velocity $\mathbf{v}$ ($= \mathbf{v}_o - \mathbf{v}_i$), is the repulsive criterion. This criterion has been studied in other papers,[4, 6, 7] although with a slightly different formulation. The repulsive criterion for $\mathbf{s}$, $\mathbf{v}$, and ϵ is denoted `repulsive_criterion`$(\mathbf{s}, \mathbf{v}, \epsilon)$ and is the set of new (relative) horizontal velocity vectors $\mathbf{v}'$ such that the following conditions hold.

$$
\begin{aligned}
&\mathbf{s} \neq \mathbf{0} \text{ and } \mathbf{v}' \neq \mathbf{0} \text{ and } (\epsilon \mathbf{s} \cdot \mathbf{v}^\perp \geq 0 \text{ or } \epsilon \mathbf{s} \cdot \mathbf{v}'^\perp < 0) \\
&\quad \text{and } ([\mathbf{s} \cdot \mathbf{v} < 0 \text{ and } \epsilon \mathbf{v}' \cdot \mathbf{v}^\perp < 0] \\
&\qquad \text{or } [\mathbf{s} \cdot \mathbf{v} \geq 0 \text{ and } \epsilon \mathbf{v}' \cdot \mathbf{v}^\perp \leq 0 \text{ and } (\mathbf{v} = \mathbf{0} \Rightarrow \mathbf{s} \cdot \mathbf{v}' \geq 0) \text{ and } (\mathbf{v} \neq \mathbf{0} \Rightarrow \mathbf{s} \cdot \mathbf{v}' > \mathbf{s} \cdot \mathbf{v})]).
\end{aligned}
\tag{10}
$$

This is a condition on a new relative velocity $\mathbf{v}'$, and it is with respect to the original relative velocity $\mathbf{v}$ (which is a parameter to it, along with the relative position $\mathbf{s}$). It also has as a parameter ϵ, which is equal to either -1 or 1. This parameter ϵ determines whether the new maneuver, represented by the velocity vector $\mathbf{v}'$, is right, when $\epsilon = 1$, or left, when $\epsilon = -1$, in the *relative* frame of reference. In contrast to vertical maneuvers where different directions are needed for the two aircraft, the horizontal repulsiveness criterion requires that each aircraft choose the *same* value of ϵ from its perspective; otherwise, the result would not be coordinated.

The condition `repulsive_criterion`$(\mathbf{s}, \mathbf{v}, \epsilon)$ divides the possible turn maneuvers for the ownship into two sets, those velocities $\mathbf{v}'_o$ for which `repulsive_criterion`$(\mathbf{s}, \mathbf{v}, -1)(\mathbf{v}'_o - \mathbf{v}_i)$ holds, and those for which `repulsive_criterion`$(\mathbf{s}, \mathbf{v}, 1)(\mathbf{v}'_o - \mathbf{v}_i)$ holds. This is depicted in Figure 4. Each possible new track angle in Figure 4 represents an instantaneous turn of the ownship to a new relative velocity $\mathbf{v}' = \mathbf{v}'_o - \mathbf{v}_i$.
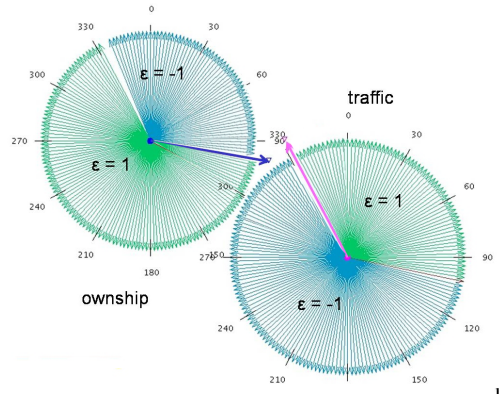
American Institute of Aeronautics and Astronautics

Figure 4. Repulsive Criterion

V.F. Piecewise Linear Repulsive Maneuvers

As noted in Section V.D, the maneuvers considered for the ownship and intruder follow a circular arc in piecewise linear segments, followed by a linear segment. Each velocity in this sequence is required to satisfy the instantaneous repulsive criterion with respect to the previous velocity, at the position where the velocities change. Thus, such a kinematic, piecewise linear trajectory for the ownship (similarly for the intruder) is represented with several objects:

- The original position $\mathbf{s}_o$.

- The original velocity $\mathbf{v}_o$.

- A time step t_{stp}, e.g. one second, to spend on each linear segment before the final linear segment.

- A sequence $V'_o = \{\mathbf{v}'_{o,1}, \ldots, \mathbf{v}'_{o,k}\}$ of velocities.

Given these objects, the (absolute, not relative) position and velocity of the aircraft following this trajectory at a future time $t \geq 0$ (the current time is 0) are given by the following functions.

$$\texttt{position}_o(V'_o, t) = \mathbf{s}_o + (t_{\text{stp}} \cdot \mathbf{v}'_{o,1} + \cdots + t_{\text{stp}} \cdot \mathbf{v}'_{o,j}) + (t - j \cdot t_{\text{stp}}) \cdot \mathbf{v}'_{o,j+1}$$
$$\texttt{velocity}_o(V'_o, t) = \mathbf{v}'_{o,j+1}$$

where $j = \texttt{min}\{\lfloor \frac{t}{t_{\text{stp}}} \rfloor, k - 1\}$.

Given the time step t_{stp}, the original velocity $\mathbf{v}_o$, the bank angle β, the gravitational acceleration constant gn, and a value $\delta = \pm 1$ representing a right or left turn (-1 for a right turn and 1 for a left), the rotation matrix

$$M(\mathbf{v}_o, t_{\text{stp}}, \beta, \delta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \text{ where } \theta = \delta \cdot t_{\text{stp}} \cdot \text{gn} \cdot \tan(|\beta|)/||\mathbf{v}_o||$$

is such that multiplication of a velocity vector by this matrix corresponds to rotating it in the direction of δ the same amount that a would result from turning with bank angle β for t_{stp} seconds. That is, if the sequence $V'_0 = \{\mathbf{v}'_{o,1}, \ldots, \mathbf{v}'_{o,k}\}$ corresponds to the banked turn with ground speed $||\mathbf{v}_o||$ in the direction of δ with original velocity $\mathbf{v}_o$, then

$$\mathbf{v}'_{o,1} = M(\mathbf{v}_o, t_{\text{stp}}, \beta, \delta) \cdot \mathbf{v}_o \quad \text{and} \quad \mathbf{v}'_{o,i+1} = M(\mathbf{v}_o, t_{\text{stp}}, \beta, \delta) \cdot \mathbf{v}'_{o,i} \tag{11}$$

for every i. Using this rotation matrix and the repulsive criterion, the specific candidate maneuvers chosen for the ownship and the intruder for a given relative direction parameter ϵ are able to be defined.

American Institute of Aeronautics and Astronautics

V.G. Choosing a Direction to Turn for the Candidate Maneuvers

As described above in Section V.C, given a value for ϵ corresponding to a direction in the relative frame of reference, a maneuver is computed for each aircraft in the relative direction corresponding to ϵ. This section describes whether a right turn or a left turn is chosen for each aircraft. The maneuver is chosen in the ϵ-repulsive direction if one exists, meaning the direction allowing a piecewise linear maneuver such that each velocity satisfies the repulsive criterion (for ϵ) with respect to the velocity on the previous segment at the position where the velocity of the maneuver changes from one to the other.

For each aircraft, recall that a given maneuver is represented by the original position and velocity as well as a time step t_{stp} and a sequence of velocities for the maneuver. There is a matrix, described above, that determines the velocities in this sequence from the previous velocities. For instance, for the ownship, this matrix is $M(\mathbf{v}_o, t_{\text{stp}}, \beta, \delta)$, where $\delta = -1$ represents a right turn and $\delta = 1$ a left turn.

For choosing the absolute direction to choose for each aircraft (right or left in the absolute frame of reference), this matrix is simply used to determine if the first entry $\mathbf{v}'_{o,1}$ in the velocity sequence V'_o (Equation (11)) is in the repulsive direction or not. That is, whether

$$\text{The vector} \quad M(\mathbf{v}_o, t_{\text{stp}}, \beta, \delta) \cdot \mathbf{v}_o \quad \text{satisfies} \quad \texttt{repulsive\_criterion}(\mathbf{s}, \mathbf{v}, \epsilon) \tag{12}$$

where, as above, $\mathbf{s} = \mathbf{s}_o - \mathbf{s}_i$ and $\mathbf{v} = \mathbf{v}_o - \mathbf{v}_i$. If this is satisfied for $\delta = -1$, then a right turn is chosen, and $M(\mathbf{v}_o, t_{\text{stp}}, \beta, -1)$ is used to form the velocity sequence according to Equation (11). If this is not the case but Equation (12) is satisfied for $\delta = 1$, then a left turn is chosen, and $M(\mathbf{v}_o, t_{\text{stp}}, \beta, 1)$ is used to form the velocity sequence according to Equation (11).

Once the direction has been chosen, it is simple to recursively compute the velocity sequences $V'_o = \{\mathbf{v}'_{o,1}, \ldots, \mathbf{v}'_{o,k}\}$ and $V'_i = \{\mathbf{v}'_{i,1}, \ldots, \mathbf{v}'_{i,k}\}$. This is accomplished through the function $\texttt{man\_seq}$, defined below. The function call

$$\texttt{man\_seq}(\epsilon, \mathbf{s}_o, \mathbf{s}_i, \mathbf{v}_o, \mathbf{v}_i, t_{\text{stp}}, \beta_o) \tag{13}$$

returns the velocity sequence produced by the following process.

1. Set $\mathbf{s} = \mathbf{s}_o - \mathbf{s}_i$ and $\mathbf{v} = \mathbf{v}_o - \mathbf{v}_i$.

2. If the vector $M(\mathbf{v}_o, t_{\text{stp}}, \beta_o, -1) \cdot \mathbf{v}_o$ satisfies $\texttt{repulsive\_criterion}(\mathbf{s}, \mathbf{v}, \epsilon)$, then define the matrix M by $M = M(\mathbf{v}_o, t_{\text{stp}}, \beta_o, -1)$. Otherwise define $M = M(\mathbf{v}_o, t_{\text{stp}}, \beta_o, 1)$.

3. Set $\mathbf{v}'_{o,1} = M \cdot \mathbf{v}_o$. If $\mathbf{v}'_{o,1}$ does not satisfy $\texttt{repulsive\_criterion}(\mathbf{s}, \mathbf{v}, \epsilon)$, then return the sequence $\{\mathbf{v}_o\}$.

4. Recursively define a velocity sequence by $\mathbf{v}'_{o,i+1} = M \cdot \mathbf{v}'_{o,i}$. Let k be the first natural number such that $\mathbf{v}'_{o,k+1}$ *does not* satisfy the condition $\texttt{repulsive\_criterion}((\mathbf{s}_o + \Sigma_{i=1}^{k} t_{\text{stp}} \mathbf{v}'_{o,i}) - (\mathbf{s}_i + k t_{\text{stp}} \mathbf{v}_i), \mathbf{v}'_{o,k}, \epsilon)$.

5. Return $\{\mathbf{v}'_{o,1}, \ldots, \mathbf{v}'_{o,k}\}$.

Note that the same process can be used to compute a maneuver sequence $V'_i = \{\mathbf{v}'_{i,1}, \ldots, \mathbf{v}'_{i,k}\}$ for the intruder by calling $\texttt{man\_seq}(\epsilon, \mathbf{s}_i, \mathbf{s}_o, \mathbf{v}_i, \mathbf{v}_o, t_{\text{stp}}, \beta_i)$.

V.H. Computing the Minimum Distance Between Two Trajectories

Given two trajectories,

- one for the ownship represented by original position $\mathbf{s}_o$ and velocity $\mathbf{v}_o$, the time step t_{stp} to spend at each velocity, and a velocity sequence $V'_o = \{\mathbf{v}'_{o,1}, \ldots, \mathbf{v}'_{o,k}\}$,

- one for the intruder represented by original position $\mathbf{s}_i$ and velocity $\mathbf{v}_i$, the time step t_{stp} to spend at each velocity, and a velocity sequence $V'_i = \{\mathbf{v}'_{i,1}, \ldots, \mathbf{v}'_{i,m}\}$,

which represent maneuvers corresponding to the relative direction associated with a given ϵ value, it is possible to compute the minimum distance between these trajectories, and the time at which this minimum occurs. This time of minimum distance is denoted with the following function.

$$\texttt{min\_time}(\mathbf{s}_o, \mathbf{s}_i, t_{\text{stp}}, V'_o, V'_i): \quad \text{time of minimum distance between the trajectories}$$

However, the distance at this time is not (quite) what is used in the algorithm `dir_select` in the place of the ϵ-to-distance function `eps_to_dist`. The minimum of three such distances is actually used, and each of these may be negated before this minimum is computed. Each distance is *negated* if in the relative frame of reference, the corresponding trajectories are passing on the *wrong* side of the origin with respect to the chosen direction. That is, if $\epsilon = -1$ and the trajectories pass to the left of the origin (in the relative frame) at their closest point, then the negation of their minimum distance is computed. Thus, the functions `position` and `velocity` from Section V.F are used to compute the relative position and velocity between the two trajectories at this minimum time, which are then substituted into the function `horiz_eps_choice` to determine the relative direction value (-1 or 1) at that time of minimum distance. If this value does not match the value ϵ, then the minimum distance is negated. This is simply accomplished by multiplying the minimum distance by the product of ϵ and this output of `horiz_eps_choice`.

$$
\begin{aligned}
\texttt{horiz\_eps\_dist}&(\epsilon, \mathbf{s}_o, \mathbf{s}_i, t_{\text{stp}}, V'_o, V'_i) = \\
&\texttt{let } t^{\min} = \texttt{min\_time}(\mathbf{s}_o, \mathbf{s}_i, t_{\text{stp}}, \{\mathbf{v}'_{o,1}, \dots, \mathbf{v}'_{o,k}\}, \{\mathbf{v}'_{i,1}, \dots, \mathbf{v}'_{i,m}\}), \\
&\qquad \mathbf{s}_o^{\min} = \texttt{position}_o(V'_o, t^{\min}) \\
&\qquad \mathbf{v}_o^{\min} = \texttt{velocity}_o(V'_o, t^{\min}) \\
&\qquad \mathbf{v}_i^{\min} = \texttt{position}_i(V'_i, t^{\min}) \\
&\qquad \mathbf{v}_i^{\min} = \texttt{velocity}_i(V'_i, t^{\min}) \texttt{ in} \\
&\qquad \epsilon \cdot \texttt{horiz\_eps\_choice}(\mathbf{s}_o^{\min}, 0, \mathbf{v}_o^{\min}, 0, \mathbf{s}_i^{\min}, 0, \mathbf{v}_i^{\min}, 0) \cdot \|\mathbf{s}_o^{\min} - \mathbf{s}_i^{\min}\|
\end{aligned}
$$

When `horiz_eps_choice` is called in this function, zeros can be used in all vertical components because they do not affect the answer due to the definition of `horiz_eps_choice` given in Section V.B.

Also note that the function `horiz_eps_dist` can be used to compute not only the minimum distance between these two piecewise linear trajectories, but also the minimum distance when only one aircraft maneuvers and the other continues along its original linear trajectory. For the case of the intruder following a linear trajectory, this can be achieved by replacing the computed intruder velocity sequence $\{\mathbf{v}'_{i,1}, \dots, \mathbf{v}'_{i,m}\}$ with the sequence $\{\mathbf{v}_i\}$ in the function `horiz_eps_dist`. The case of the ownship not maneuvering is handled similarly.

The function `horiz_eps_to_dist`, defined below, is designed to compute the worst case separation distance (as described in Section V.C) when choosing a particular ϵ value for relative direction selection. It works as follows. The function computes the minimum distance function `horiz_eps_dist` for three pairs of maneuvers: (1) The intruder continues linearly and the ownship maneuvers corresponding to ϵ, (2) The ownship continues linearly and the intruder maneuvers corresponding to ϵ, and (3) Both aircraft maneuver corresponding to ϵ. The minimum of these three distances is then chosen. Formally, `horiz_eps_to_dist`, which will be used in place of the parameter `eps_to_dist` in the function `dir_select` to define the horizontal (relative) direction selection logic, is defined as follows.

$$
\begin{aligned}
\texttt{horiz\_eps\_to\_dist}&_{t_{\text{stp}}, \beta_o, \beta_i}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \epsilon) = \\
&\texttt{let } V'_o = \texttt{man\_seq}(\epsilon, \mathbf{s}_o, \mathbf{s}_i, \mathbf{v}_o, \mathbf{v}_i, t_{\text{stp}}, \beta_o), \\
&\qquad V'_i = \texttt{man\_seq}(\epsilon, \mathbf{s}_i, \mathbf{s}_o, \mathbf{v}_i, \mathbf{v}_o, t_{\text{stp}}, \beta_i), \\
&\qquad \texttt{D\_both} = \texttt{horiz\_eps\_dist}(\epsilon, \mathbf{s}_o, \mathbf{s}_i, t_{\text{stp}}, V'_o, V'_i), \\
&\qquad \texttt{D\_ownship} = \texttt{horiz\_eps\_dist}(\epsilon, \mathbf{s}_o, \mathbf{s}_i, t_{\text{stp}}, V'_o, \{\mathbf{v}_i\}), \\
&\qquad \texttt{D\_intruder} = \texttt{horiz\_eps\_dist}(\epsilon, \mathbf{s}_o, \mathbf{s}_i, t_{\text{stp}}, \{\mathbf{v}_o\}, V'_i) \texttt{ in} \\
&\texttt{min}(\texttt{D\_both}, \texttt{D\_ownship}, \texttt{D\_intruder})
\end{aligned}
$$

V.I. Horizontal Direction Selection

Recall that the function `dir_select` from Section III.C returns a relative direction parameter, either -1 or 1, and this can be used to coordinate vertical maneuvers as in Section IV, where each aircraft computes a different direction parameter, e.g., one goes up and the other goes down. As noted earlier, the function `dir_select` can also be used to coordinate horizontal maneuvers by returning to each aircraft a relative horizontal direction. However, for horizontal maneuvers, each aircraft computes the same horizontal direction parameter. A simple example of this is that in a head-on conflict, both aircraft can resolve it by turning to the right.

The function `dir_select` uses two generic functions and two parametric distances that uniquely determine its behavior once they are set.

- The function `eps_choice`, which determines the preferred relative direction parameter ϵ. For horizontal coordination, the function `horiz_eps_choice` from Section V.B is used in place of `eps_choice`.

- The function `eps_to_dist`, which returns a distance corresponding to a maneuver associated with a given relative direction ϵ. For horizontal coordination, the function `horiz_eps_to_dist` from Section V.H is used in place of `eps_to_dist`.

- The parameter `Dlim`, which is a distance. This is any distance determined by the user that gives a *desired* separation. For horizontal coordination, any distance may be used.

- The parameter `Dcol`, which is a distance. This is any distance determined by the user that gives a *minimum* acceptable separation. If no such separation can be guaranteed, the algorithm returns 0, meaning no solution is found. For horizontal coordination, any distance may be used.

This now enables the definition of the function `horiz_rel_dir`, which computes a relative direction parameter.

$$\texttt{horiz\_rel\_dir}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, t_{\text{stp}}, \beta_o, \beta_i, \texttt{Dlim}) =$$
$$\texttt{dir\_select}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}). \tag{14}$$

The function `horiz_rel_dir` returns a relative direction parameter -1 or 1. Here, as above, t_{stp} is a small time step for each linear segment on the approximated turns of the aircraft, β_o and β_i are the bank angles for turning for each aircraft, and `Dlim` is a distance that gives a minimum acceptable separation. The most important property of the function `horiz_rel_dir` is that it is implicitly coordinated in the sense of the following theorem.

Theorem 3. *The ownship and intruder, when computing the function from their own perspectives, will compute the same answer. That is,*

$$horiz\_rel\_dir(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, t_{stp}, \beta_o, \beta_i, Dlim) =$$
$$horiz\_rel\_dir(\mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, \mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, t_{stp}, \beta_i, \beta_o, Dlim) \tag{15}$$

V.J. Computing Actual Horizontal Maneuvers

The function `horiz_rel_dir` defined above computes a relative direction for each aircraft, a value equal to -1 or 1 that can be used in the repulsive criterion `repulsive_criterion` to compute turn maneuvers for the aircraft. Recall from Section V.G that the function call $\texttt{man\_seq}(\epsilon, \mathbf{s}_o, \mathbf{s}_i, \mathbf{v}_o, \mathbf{v}_i, t_{\text{stp}}, \beta_o)$ computes a sequence of velocities for the ownship that, when paired with the time step t_{stp} for each velocity segment, corresponds to a maneuver that moves in the direction of ϵ in the relative frame of reference. Thus, for computing coordinated horizontal resolution maneuvers for these two aircraft, the output of `horiz_rel_dir` is simply used in place of ϵ in the function `man_seq`. The following function computes a maneuver sequence for the ownship and can similarly be used for the intruder.

$$\texttt{horiz\_coord\_man}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, t_{\text{stp}}, \beta_o, \beta_i, \texttt{Dlim}) =$$
$$\texttt{man\_seq}(\texttt{horiz\_rel\_dir}(\mathbf{s}_o, s_{oz}, \mathbf{v}_o, v_{oz}, \mathbf{s}_i, s_{iz}, \mathbf{v}_i, v_{iz}, t_{\text{stp}}, \beta_o, \beta_i, \texttt{Dlim}), \mathbf{s}_o, \mathbf{s}_i, \mathbf{v}_o, \mathbf{v}_i, t_{\text{stp}}, \beta_o) \tag{16}$$

The function `horiz_coord_man` will return a maneuver sequence for each aircraft that will be coordinated in the sense of the repulsive criterion `repulsive_criterion`. Alternatively, assuming that the pilot maneuvers in accordance with the parameters, i.e., bank angle, input into the algorithm, the final velocity vector of the sequence or simply its heading can be returned to the pilot as resolution advice.

VI. Conclusion

This paper presents an algorithm that provides directional information for resolving impending air traffic conflicts such as loss of separation, as in the case of conflict resolution systems,[3] well-clear violation, as in the case of detect and avoid systems for Unmanned Aircraft Systems,[1] or near-miss, as in the case of

collision avoidance systems.[8] The logic of this algorithm is inspired by the RA logic of TCAS II. However, the proposed logic is more complicated than simply replacing the notions of up/down in TCAS II with the notions of right/left. Rather, this paper uses the notions of right/left (of the origin) in the *relative* frame of reference for each aircraft, where the intruder is viewed as being fixed at the origin. A function is presented in Section V.I for computing a relative direction for the maneuvers, which returns the same answer for both aircraft. Section V.J defines an algorithm for computing an actual maneuver for each aircraft that corresponds to this direction in the relative frame of reference. The main theorem presented in this paper states that if both aircraft execute maneuvers according to the proposed functions, then these maneuvers are coordinated. This assumes common knowledge of each other's state vectors (position and velocity) at the same time, so future work may examine strategies to mitigate the effects of possible data errors on this method.

Using the repulsive criterion to determine whether to turn left or right is not new.[6,7] However, this paper presents a function to pick the ϵ parameter (relative direction) for the repulsive criterion that uses the TCAS II resolution direction logic as its inspiration. In previous papers,[6,7] a very simple formula was used to pick the ϵ parameter (right or left in the relative frame), namely the *canonical* choice $\epsilon = -\texttt{sign}(\mathbf{s} \cdot \mathbf{v}^{\perp})$. In most cases, this choice for the ϵ parameter works well. In fact, it can be understood as determining whether the ownship is *currently* going left or right of the intruder in the relative frame. The motivation of the current work is that this choice of ϵ is not always optimal in near−term conflict situations because a manuever satisfying the criterion for this ϵ may not provide adequate separation, while choosing the opposite value for ϵ, namely $-\epsilon$, does. This paper addresses this problem by using the resolution advisory logic of TCAS II as a guide.

References

[1]Stephen Cook, Dallas Brooks, Rodney Cole, Davis Hackenberg, and Vincent Raska. Defining Well Clear for Unmanned Aircraft Systems. In *Proceedings of the 2015 AIAA SciTech Conference*, number AIAA-2015-0481, Kissimmee, Florida, USA, January 2015.

[2]International Civil Aviation Organization (ICAO). Annex 2 to the Convention on International Civil Aviation, July 2005.

[3]James Kuchar and Lee Yang. A review of conflict detection and resolution modeling methods. *IEEE Transactions on Intelligent Transportation Systems*, 1(4):179–189, December 2000.

[4]César Muñoz, Ricky Butler, Anthony Narkawicz, Jeffrey Maddalon, and George Hagen. A criteria standard for conflict resolution: A vision for guaranteeing the safety of self-separation in NextGen. Technical Memorandum NASA/TM-2010-216862, NASA, Langley Research Center, Hampton VA 23681-2199, USA, October 2010.

[5]César Muñoz, Anthony Narkawicz, and James Chamberlain. A TCAS-II resolution advisory detection algorithm. In *Proceedings of the AIAA Guidance Navigation, and Control Conference and Exhibit 2013*, number AIAA-2013-4622, Boston, Massachusetts, August 2013.

[6]Anthony Narkawicz and César Muñoz. State-based implicit coordination and applications. Technical Publication NASA/TP-2011-217067, NASA, Langley Research Center, Hampton VA 23681-2199, USA, March 2011.

[7]Anthony Narkawicz, César Muñoz, and George Hagen. An independent and coordinated criterion for kinematic aircraft maneuvers. In *Proceedings of the 14th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, number AIAA-2014-2859, Georgia, Atlanta, USA, June 2014.

[8]RTCA SC-147. RTCA-DO-185B, Minimum operational performance standards for traffic alert and collision avoidance system II (TCAS II), July 2009.

[9]US Code of Federal Regulations. Title 14 Aeronautics and Space; Part 91 General operating and fight rules, 1967.