Formal Verification of an Optimal Air Traffic Conflict Resolution and Recovery Algorithm*

André L. Galdino[1,**], César Muñoz[2], and Mauricio Ayala-Rincón[1,**]

[1] Universidade de Brasília, Brazil
{galdino,ayala}@unb.br
[2] National Institute of Aerospace, USA
munoz@nianet.org

Abstract. Highly accurate positioning systems and new broadcasting technology have enabled air traffic management concepts where the responsibility for aircraft separation resides on pilots rather than on air traffic controllers. The Formal Methods Group at the National Institute of Aerospace and NASA Langley Research Center has proposed and formally verified an algorithm, called KB3D, for distributed three dimensional conflict resolution. KB3D computes resolution maneuvers where only one component of the velocity vector, i.e., ground speed, vertical speed, or heading, is modified. Although these maneuvers are simple to implement by a pilot, they are not necessarily optimal from a geometrical point of view. In general, optimal resolutions require the combination of all the components of the velocity vector. In this paper, we propose a two dimensional version of KB3D, which we call KB2D, that computes resolution maneuvers that are optimal with respect to ground speed and heading changes. The algorithm has been mechanically verified in the Prototype Verification System (PVS). The verification relies on algebraic proof techniques for the manipulation of the geometrical concepts relevant to the algorithm as well as standard deductive techniques available in PVS.

1 Introduction

Air traffic management concepts such as *Free Flight* [12,5] and *Distributed Air/-Ground Traffic Management* [10] target the predicted increase of air traffic by distributing the responsibility for separation among pilots and air traffic controllers. This new mode of operation is supported by advances in surveillance, communications, and software technologies. In particular, conflict detection and resolution (CD&R) systems are designed to increase traffic awareness and provide corrective maneuvers for impending conflicts [7]. On-board CD&R systems

* This work was supported by the National Aeronautics and Space Administration, Langley Research Center under the Research Cooperative Agreement No. NCC-1-02043 and by the Brazilian Research Council CNPq under Grant 471791/2004-0.
** Authors partially supported by the Brazilian Research Council CNPq.

D. Leivant and R. de Queiroz. (Eds.): WoLLIC 2007, LNCS 4576, pp. 177–188, 2007.

are particularly attractive as they support the required decentralized decision making of new air traffic management concepts.

KB3D [2] is a three dimensional (3D) distributed CD&R algorithm, designed and formally verified by the Formal Methods Group at NIA and NASA Langley Research Center. Since KB3D uses state information, e.g., position and velocity vectors, to detect and solve conflicts between two aircraft, namely, *ownship* and *traffic*, it can be characterized as a *pairwise tactical* algorithm. KB3D assumes that the ownship and traffic aircraft are surrounded by a cylindrical protected zone of diameter D and height H centered at the aircraft's positions. A *loss of separation* between the two aircraft is defined as the overlapping of their protected zones. A *conflict* is a predicted loss of separation within a lookahead time T. When a conflict is detected, KB3D outputs a choice of resolution maneuvers for the ownship. Each resolution maneuver is expressed as a new velocity vector for the ownship that yields a conflict-free trajectory. The resolutions computed by KB3D modify only one component of the ownship's velocity vector:

- *Vertical speed*: the aircraft keeps the horizontal component of its velocity vector, but modifies its vertical speed;
- *Ground speed*: the aircraft keeps its heading and vertical speed but modifies its ground speed;
- *Heading*: the aircraft keeps its ground speed and vertical speed but modifies its heading.

Not all conflict situations have all three kinds of resolutions. However, if the aircraft are in conflict but they are still separated, KB3D guarantees at least one theoretical vertical solution.

Two important extensions to KB3D are (1) *time arrival constraints* and (2) *coordinated strategies.*

1. **Time Arrival Constraints.** KB3D has been extended to compute *recovery* maneuvers [4,8]. A recovery maneuver brings back the ownship to its target point at the scheduled time, once the conflict has been avoided.
2. **Coordinated Strategies**. A *strategy* is a procedure that selects a subset of the resolution maneuvers proposed by a CD&R algorithm. A strategy is *coordinated* if it ensures separation when both aircraft simultaneously maneuver to solve the conflict. Coordinated strategies guarantee that aircraft using the same CD&R system will not fly into each other during the resolution maneuver. The coordination is *implicit* if the only communication required to achieve the coordination is the broad-casted state information of the aircraft. It has been formally verified that KB3D supports an implicitly coordinated resolution strategy [3].

KB3D resolutions yield trajectories for the ownship where the protected zones of the aircraft touch but not overlap. These trajectories require a small change of course for the ownship. However, since KB3D resolutions modify only one component of the original velocity vector, i.e., heading, vertical speed, or ground speed; KB3D resolutions are not necessarily optimal from a geometrical point

of view. In general, an optimal resolution requires simultaneous variations in all three components. In this paper, we propose a two dimensional (2D) version of KB3D, called KB2D, that computes optimal resolution maneuvers for combined variations of ground speed and heading. As KB3D, KB2D has been extended with time arrival constraints and coordinated resolutions. Moreover, KB2D has been specified and mechanically verified in the Prototype Verification System (PVS) [11]. In addition to the standard deductive techniques available in PVS, we have extensively used strategies in the PVS packages Field [9] and Manip [13]. These packages provide tools for the algebraic manipulations required to deal with the geometrical concepts used by the algorithm.

This paper is organized as follows. Section 2 introduces the geometric framework for specifying the 2D CD&R problem. Section 3 presents the KB2D algorithm. The formal proofs that the resolution and recovery maneuvers computed by KB2D are correct and optimal, and that KB2D supports coordinated maneuvers are presented in Section 4.

2 Geometric Framework

We consider the airspace as a 2D Cartesian coordinate system in $\mathbb{R}_2$. The ownship's and traffic's initial positions, at time $t = 0$, are given by the vectors $\boldsymbol{s_o} = (s_{ox}, s_{oy})$ and $\boldsymbol{s_i} = (s_{ix}, s_{iy})$, respectively. The ownship's and traffic's original velocity vectors are given by $\boldsymbol{v_o} = (v_{ox}, v_{oy})$ and $\boldsymbol{v_i} = (v_{ix}, v_{iy})$, respectively.

The representation of the airspace by a 2D Cartesian system hints that the KB2D logic is based on a flat earth assumption. Indeed, we represent aircraft dynamics by a simple point moving at constant speed along a linear trajectory, Hence, the course of an aircraft can be described by a position, a velocity vector, and a time interval. We also assume that aircraft can change course and speed in zero time. All these unrealistic assumptions are typical of tactical CD&R systems with short lookahead times (usually, $T = 5$ minutes) and large *protected zones* (usually, $D = 5$ nautical miles and $H = 1000$ feet), to be defined below.

As it simplifies the mathematical development, we consider the ownship's motion relative to the traffic aircraft. Hence, we introduce a *relative coordinate system* where the traffic's position is at the origin, and the relative position and velocity vectors of the ownship with respect to the traffic aircraft are given by $\boldsymbol{s} = (s_x, s_y) = \boldsymbol{s_o} - \boldsymbol{s_i}$ and $\boldsymbol{v} = (v_x, v_y) = \boldsymbol{v_o} - \boldsymbol{v_i}$, respectively. In the relative coordinate system, the protected zone P is a cylinder of diameter $2D$ located in the center of the coordinate system:

$$P = \{(x, y, 0) \mid x^2 + y^2 < D^2\}. \tag{1}$$

Consequently, we define *loss of separation* at time t as the incursion of the ownship in the relative protected zone of the traffic aircraft at time t, i.e.,

$$\boldsymbol{s} + t\boldsymbol{v} \in P. \tag{2}$$

Given a lookahead time T, the aircraft are said to be in *conflict* if there exists a time $0 < t < T$ when they are predicted to lose separation.

We note that the relative protected zone P is twice the size of each individual protected zone in the absolute coordinate system. It can be easily checked that the definitions of "conflict" and "loss of separation" in the relative coordinate system are equivalent to the definitions in the absolute one.

A *resolution* is a new velocity vector $\boldsymbol{v_o'}$ for the ownship. The resolution is *correct* if for all $t > 0$,

$$\boldsymbol{s} + t(\boldsymbol{v_o'} - \boldsymbol{v_i}) \notin P. \tag{3}$$

A resolution $\boldsymbol{v_o'}$ for the ownship is *smaller* than a resolution $\boldsymbol{v_a'}$, denoted by the order relation $\boldsymbol{v_o'} \preceq \boldsymbol{v_a'}$, if and only if

$$||\boldsymbol{v_o'} - \boldsymbol{v_o}|| \ \leq \ ||\boldsymbol{v_a'} - \boldsymbol{v_o}||, \tag{4}$$

where $||\boldsymbol{v}||$ denotes the norm of $\boldsymbol{v}$.

Resolutions $\boldsymbol{v_o'}$ and $\boldsymbol{v_i'}$ are *coordinated* for the ownship and traffic aircraft, respectively, if for all $t > 0$,

$$\boldsymbol{s} + t(\boldsymbol{v_o'} - \boldsymbol{v_i'}) \notin P. \tag{5}$$

An *arrival time* $t'' > 0$ determines a way point s'', called *target point*:

$$\boldsymbol{s''} = \boldsymbol{s} + t''\boldsymbol{v}. \tag{6}$$

A *resolution/recovery maneuver* for an arrival time t'' is a triple $(t', \boldsymbol{v_o'}, \boldsymbol{v_o''})$ where $t' > 0$ is a *time of switch*, $\boldsymbol{v_o'}$ is a resolution velocity for the ownship, and $\boldsymbol{v_o''}$ is a recovery velocity for the ownship. The resolution/recovery maneuver is *correct* if and only if

- $\boldsymbol{v_o'}$ is a correct resolution for the ownship, and
- $\boldsymbol{v_o''}$ is a correct recovery for the ownship, i.e., for all times $0 \leq t \leq t'' - t'$,

$$\boldsymbol{s} + t'\boldsymbol{v'} + t\boldsymbol{v''} \notin P, \text{ and} \tag{7}$$

$$\boldsymbol{s} + t'\boldsymbol{v'} + (t'' - t')\boldsymbol{v''} = \boldsymbol{s''}, \tag{8}$$

where $\boldsymbol{v'} = \boldsymbol{v_o'} - \boldsymbol{v_i}$ and $\boldsymbol{v''} = \boldsymbol{v_o''} - \boldsymbol{v_i}$.

3 KB2D

KB2D, like KB3D, computes resolution maneuvers that yield trajectories that are tangent to the protected zone in the relative coordinate system. Figure 1 shows a conflict situation and the resolution and recovery courses in a two dimensional geometry. This figure illustrates two symmetries in the conflict resolution and recovery problem. First, resolution and recovery maneuvers are solved in a symmetric way, i.e., a recovery course for the conflicting situation described by the relative position and velocity vectors $\boldsymbol{s}$, $\boldsymbol{v}$, and arrival time t'' is a resolution

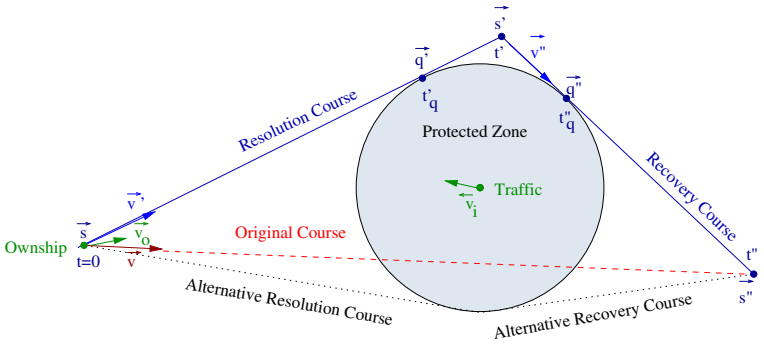

Fig. 1. Resolution and recovery courses (2D)

course for the conflicting situation described by the target point $\boldsymbol{s''} = \boldsymbol{s} + t''\boldsymbol{v}$, relative velocity vector $-\boldsymbol{v}$, and arrival time t''.

The second symmetry is due to the fact that for any relative position $\boldsymbol{s}$ exterior to P, i.e., $s_x^2 + s_y^2 > D^2$, there are two tangent courses to the protected zone. Each tangent to the protected zone determines a family of resolution and recovery maneuvers. Intuitively, the optimal relative resolution maneuver, with respect to the order relation $\preceq$ defined by Formula 4, is member of one of these families. Furthermore, for each family, the optimal relative resolution maneuver $\boldsymbol{v'}$ is the perpendicular projection of the relative vector $\boldsymbol{v}$ on the corresponding tangent course.

Based on all these observations, we define two functions `kb2d` and `recovery`. The function `kb2d` has as inputs

- the initial relative ownship's position $\boldsymbol{s} = (s_x, s_y)$,
- the absolute velocities $\boldsymbol{v_o} = (v_{ox}, v_{oy})$, $\boldsymbol{v_i} = (v_{ix}, v_{iy})$ of the ownship and traffic aircraft, respectively, and
- a parameter $e = \pm 1$, which determines a particular tangent for the resolution course.

It returns a couple (v'_{ox}, v'_{oy}) that defines a resolution velocity vector for the ownship $\boldsymbol{v'_o} = (v'_{ox}, v'_{oy})$. The function `recovery` has the same inputs as `kb2d` and, additionally, an arrival time t''. It returns a triple (t', v''_{ox}, v''_{oy}) that defines a time of switch t' (from resolution to recovery) and a recovery velocity vector for the ownship $\boldsymbol{v''_o} = (v''_{ox}, v''_{oy})$. The functions are defined in Listing 1.1.

We have formally verified that the resolution/recovery maneuvers computed by `kb2d` and `recovery` are *correct*, i.e., they satisfy formulas 3, 7, and 8. In other words, if the ownship flies the resolution course from time 0 to t' and the recovery course from time t' to t'', then

(a) it shall not be in conflict at any time and
(b) it shall arrive to the target point s'' at the arrival time t''.

Furthermore, we also show that the resolution $\boldsymbol{v'_o}$ computed by `kb2d` is optimal with respect to the order $\preceq$ defined by Formula 4, and that it is coordinated with

Listing 1.1. The functions `kb2d` and `recovery`

```
kb2d(s_x,s_y,v_ox,v_oy,v_ix,v_iy,e) : [real,real] =
  let (v_x, v_y) = (v_ox − v_ix, v_oy − v_iy) in
  let (q'_x, q'_y) = (Q(s_x,s_y,e),Q(s_y,s_x,−e)) in
  let t'_q = contact_time(s_x,s_y,q'_x,q'_y,v_x,v_y,e) in
  if t'_q > 0 then ((q'_x − s_x)/t'_q + v_ix , (q'_y − s_y)/t'_q + v_iy)
  elsif t'_q = 0 then(v_ix,v_iy)
  else (0,0)
  endif

recovery(s_x,s_y,v_ox,v_oy,v_ix,v_iy,t'',e) : [real,real,real] =
  let (v_x, v_y) = (v_ox − v_ix, v_oy − v_iy) in
  let (s''_x, s''_y) = (s_x + t''v_x, s_y + t''v_y) in
  let (v'_ox,v'_oy) = kb2d(s_x,s_y,v_ox,v_oy,v_ix,v_iy,e) in
  let (v'_x, v'_y) = (v'_ox − v_ix, v'_oy − viy) in
  let t' = switching_time(s_x,s_y,s''_x,s''_y,v'_x,v'_y,e) in
  if t' > 0 AND t'' − t' > 0 then
    (t', (t''v_x − t'v'_x)/(t'' − t') + v_ix,(t''v_y − t'v'_y)/(t'' − t') + v_iy)
  else (0,0,0)
  endif

alpha(s_x,s_y) : real = D^2/(s_x^2 + s_y^2)

beta(s_x,s_y) : real = D√(s_x^2 + s_y^2 − D^2)/(s_x^2 + s_y^2)

Q(s_x,s_y,e):real = alpha(s_x,s_y)s_x + e beta(s_x,s_y)s_y

contact_time(s_x,s_y,q_x,q_y,v_x,v_y,e) : real =
  let d = v_x(q_x − s_x) + v_y(q_y − s_y) in
  if d ≠ 0 then ((q_x − s_x)^2 + (q_y − s_y)^2)/d
  else 0
  endif

switching_time(s_x,s_y,s''_x,s''_y,v'_x,v'_y,e) : real =
  if s''_x^2 + s''_y^2 > D^2 then
    let (q''_x, q''_y) = (Q(s''_x,s''_y,−e),Q(s''_y,s''_x,e)) in
    let (u_x, u_y) = (q''_x − s''_x, q''_y − s''_y) in
    let d = v'_y u_x − v'_x u_y in
    if d ≠ 0 then ((s_x − s''_x)u_y + (s''_y − s_y)u_x)/d
    else 0
    endif
  else 0
  endif
```

respect to the resolution $\boldsymbol{v_i'}$ computed for the traffic aircraft, i.e., $\boldsymbol{v_o'}$ and $\boldsymbol{v_i'}$ satisfy Formula 5.

These properties are proven under the following assumptions:

- Absolute ground speeds are different from zero, i.e., $v_{ox}^2 + v_{oy}^2 \neq 0$ and $v_{ix}^2 + v_{iy}^2 \neq 0$.
- Initial relative position $\boldsymbol{s}$ and target point $\boldsymbol{s''}$ are external to P, i.e., $\boldsymbol{s}, \boldsymbol{s''} \in \{(x, y) \mid x^2 + y^2 > D^2\}$.
- The aircraft are in conflict before t'', i.e., $\boldsymbol{s}+t\boldsymbol{v} \in P$ for some time $0 < t < t''$.

4 Formal Verification of KB2D

In this section, we present the verification of KB2D in the Prototype Verification System. The full PVS development is available at `http://research.nianet.org/fm-at-nia/KB3D/kb2d.dump`. The proofs use general results on real analysis available in the NASA PVS Libraries,[1] and strategies for the manipulation of real expressions from the PVS packages Field [9] and Manip [13]. We also adapt some predicates and results from the formal verification of KB3D [8].

4.1 Geometrical Analysis

The function `kb2d` discriminates left and right maneuvers (in the relative coordinate sytem) by using the parameter e ($= \pm 1$). For a given value of e, `kb2d` finds a point $\boldsymbol{q'}$, which defines a line L that is tangent to the protected zone (see Figure 2). The time when the velocity vector $\boldsymbol{v'}$ reaches the point $\boldsymbol{q'}$ is

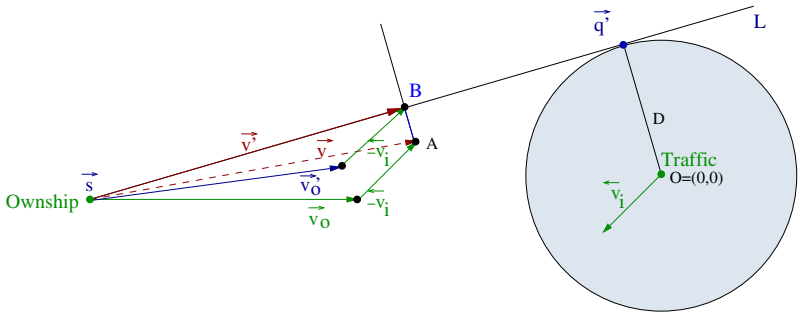

Fig. 2. Geometrical analysis of `kb2d`

called *contact time*. The relative resolution vector $\boldsymbol{v'}$ is defined as the vector that lies on the tangent line L and such that its ending point B is the orthogonal projection of the ending point A of the original relative velocity vector $\boldsymbol{v}$. Finally, `kb2d` returns the absolute ownship resolution $\boldsymbol{v_o'} = \boldsymbol{v'} + \boldsymbol{v_i}$. Special care

[1] `http://shemesh.larc.nasa.gov/fm/ftp/larc/PVS-library/pvslib.html`

is taken when the contact time is negative or when $\boldsymbol{v'}$ does not point toward $\boldsymbol{q}$. In those cases, a null value is returned to indicate that there is no solution.

The function `recovery` first finds the time of switch t' which defines a point that lies on L and on the tangent line that passes through the target point $\boldsymbol{s''}$. Then, it uses the fact that resolution and recovery maneuvers are solved in a symmetric way to compute the relative recovery vector $\boldsymbol{v''}$. Finally, `recovery` returns a triple formed by the time of switch t' and the components of the absolute ownship recovery vector $\boldsymbol{v''_o} = \boldsymbol{v''} + \boldsymbol{v_i}$.

The predicate conflict?($\boldsymbol{s}$,$\boldsymbol{v}$,T) holds if the aircraft are in conflict, i.e., if there is a time $0 < t < T$ that satisfies Formula 2:

$$\texttt{conflict?}(\boldsymbol{s}, \boldsymbol{v}, T) \equiv \exists 0 < t < T : (s_x + tv_x)^2 + (s_y + tv_y)^2 < D^2.$$

or equivalently, $\exists 0 < t < T : \boldsymbol{s} + t\boldsymbol{v} \in P$ according to Equation 2.

The predicate separation?($\boldsymbol{s}$,$\boldsymbol{v}$) holds if the relative velocity $\boldsymbol{v}$ guarantees separation for all time $t > 0$:

$$\texttt{separation?}(\boldsymbol{s}, \boldsymbol{v}) \equiv \forall t > 0 : (s_x + tv_x)^2 + (s_y + tv_y)^2 \geq D^2.$$

or equivalently, $\forall t > 0 : \boldsymbol{s} + t\boldsymbol{v} \notin P$ according to Equation 2.

A moving point $\boldsymbol{s} + t\boldsymbol{v}$ intersects the surface of the protected zone P if and only if

$$(s_x + tv_x)^2 + (s_y + tv_y)^2 = D^2. \tag{9}$$

Since we consider that the ground speed is not null, i.e., $v_x^2 + v_y^2 \neq 0$, Formula 9 reduces to a quadratic equation in t:

$$t^2(v_x^2 + v_y^2) + 2t(s_x v_x + s_y v_y) + s_x^2 + s_y^2 - D^2) = 0. \tag{10}$$

The discriminant of Formula 10, $\Delta(\boldsymbol{s}, \boldsymbol{v})$, is defined as

$$\Delta(\boldsymbol{s}, \boldsymbol{v}) = D^2(v_x^2 + v_y^2) - (s_x v_y - s_y v_x)^2. \tag{11}$$

If $\Delta(\boldsymbol{s}, \boldsymbol{v}) < 0$ then the moving point does not intersect P. Furthermore, if $\Delta(\boldsymbol{s}, \boldsymbol{v}) = 0$ we have the tangent case.

Lemma 1 (`tangent_correctness`). *For all* $\boldsymbol{s}$, $\boldsymbol{v} \neq \boldsymbol{0}$,

$s_x^2 + s_y^2 \geq D^2 \ \wedge$
$\Delta(\boldsymbol{s}, \boldsymbol{v}) = 0$
$\Rightarrow$
separation?($\boldsymbol{s}$*,*$\boldsymbol{v}$*).*

Proof. See Maddalon et al [8]. □

A resolution maneuver is said to be a *line solution* for parameter e if it lies on the tangent corresponding to e. Formally [3],

$$\texttt{line\_solution?}(\boldsymbol{s}, \boldsymbol{v}, e) \equiv s_x v_y - s_y v_x = e\frac{D(s_x v_x + s_y v_y)}{\sqrt{{s_x}^2 + {s_y}^2 - D^2}}. \tag{12}$$

The corresponding resolution maneuvers of two conflicting aircraft are coordinated if they are line solutions for the same parameter e.

Lemma 2 (`coordinated_line`). *For all* $\boldsymbol{s}$, $\boldsymbol{v} = \boldsymbol{v_o} - \boldsymbol{v_i}$, $T > 0$, $D > 0$, $\boldsymbol{v'} = \boldsymbol{v'_o} - \boldsymbol{v'_i}$, $e = \pm 1$,

$$
\begin{array}{l}
\mathit{conflict?}(\boldsymbol{s},\boldsymbol{v},T) \ \wedge \\
s_x^2 + s_y^2 > D^2 \ \wedge \\
\mathit{line\_solution?}(\boldsymbol{s},\boldsymbol{v'_o} - \boldsymbol{v_i},e) \ \wedge \\
\mathit{line\_solution?}(-\boldsymbol{s},\boldsymbol{v'_i} - \boldsymbol{v_o},e) \\
\Rightarrow \\
\quad \mathit{separation?}(\boldsymbol{s},\boldsymbol{v'}).
\end{array}
$$

Proof. See Dowek and Munoz [3]. □

4.2 Correctness of Resolution

The following theorem states that if the resolution computed by `kb2d` is not null, then it is correct, i.e., the resolution verifies Formula 3.

Theorem 1 (`kb2d_correct`). *For all* $\boldsymbol{s}$, $\boldsymbol{v} = \boldsymbol{v_o} - \boldsymbol{v_i}$, $T > 0$, $D > 0$, $\boldsymbol{v'}$, $\boldsymbol{v'_o}$, $e = \pm 1$,

$$
\begin{array}{l}
\mathit{conflict?}(\boldsymbol{s},\boldsymbol{v},T) \ \wedge \\
s_x^2 + s_y^2 > D^2 \ \wedge \\
\boldsymbol{v'_o} = \mathit{kb2d}(s_x,s_y,v_{ox},v_{oy},v_{ix},v_{iy},e) \ \wedge \\
\boldsymbol{v'} = \boldsymbol{v'_o} - \boldsymbol{v_i} \ \wedge \ \boldsymbol{v'_o} \neq \boldsymbol{0} \\
\Rightarrow \\
\quad \mathit{separation?}(\boldsymbol{s},\boldsymbol{v'}).
\end{array}
$$

Proof (Sketch). We prove that the line L is tangent to the protected zone and that the relative velocity vector $\boldsymbol{v'}$ lies on L; i.e., $\Delta(\boldsymbol{s}, \boldsymbol{v'}) = 0$. We conclude by using Lemma 1 (`tangent_correctness`). □

4.3 Optimality

We prove that the resolution velocity $\boldsymbol{v'_o}$ computed by `kb2d` is optimal with respect to any velocity vector $\boldsymbol{v'_a}$, where $\boldsymbol{v'_a} - \boldsymbol{v_i}$ lies on the same tangent L, i.e, $\boldsymbol{v'_o} \preceq \boldsymbol{v'_a}$ as defined by Formula 4.

Theorem 2 (`kb2d_optimal`). *For all* $\boldsymbol{s}$, $\boldsymbol{v} = \boldsymbol{v_o} - \boldsymbol{v_i}$, $T > 0$, $D > 0$, $\boldsymbol{v'}$, $\boldsymbol{v'_o}$, $\boldsymbol{v'_a}$, k, t'_q, $e = \pm 1$,

$$
\begin{array}{l}
\mathit{conflict?}(\boldsymbol{s},\boldsymbol{v},T) \ \wedge \\
s_x^2 + s_y^2 > D^2 \ \wedge \\
\boldsymbol{v'_o} = \mathit{kb2d}(s_x,s_y,v_{ox},v_{oy},v_{ix},v_{iy},e) \ \wedge \\
t'_q = \mathit{contact\_time}(s_x, s_y, q_x, q_y, v_x, v_y, e) \ \wedge \ t'_q > 0 \ \wedge \\
\boldsymbol{v'_a} - \boldsymbol{v_i} = k\boldsymbol{v'} \\
\Rightarrow \\
\quad \boldsymbol{v'_o} \preceq \boldsymbol{v'_a}.
\end{array}
$$

Proof (Sketch). The result follows by establishing that the factor $\frac{1}{t'_q}$ minimizes the distance between the point A (ending point of $\boldsymbol{v}$) and the ending point of a velocity vector that lies on L. As this factor is used for the definition of $\boldsymbol{v'}$, we have that for all k, $||\boldsymbol{v'} - \boldsymbol{v}|| \leq ||k\boldsymbol{v'} - \boldsymbol{v}||$. Therefore, $||\boldsymbol{v'_o} - \boldsymbol{v_o}|| \leq ||\boldsymbol{v'_a} - \boldsymbol{v_o}||$. □

4.4 Correctness of Recovery

The following theorem states that if the time of switch satisfies the condition $0 < t' < t''$ and the resolution velocity computed by `recovery` is not null, then the recovery is correct, i.e., it satisfies formulas 7 and 8.

Theorem 3 (`recovery_correct`). *For all* $\boldsymbol{s}$, $\boldsymbol{s'}$, $\boldsymbol{v} = \boldsymbol{v_o} - \boldsymbol{v_i}$, $T > 0$, $D > 0$, $\boldsymbol{v'}$, $\boldsymbol{v'_o}$, $\boldsymbol{v''}$, $\boldsymbol{v''_o}$, t', t'', $e = \pm 1$,

$$
\begin{array}{l}
\mathit{conflict?}(\boldsymbol{s},\boldsymbol{v},T) \ \wedge \\
s_x^2 + s_y^2 > D^2 \ \wedge \\
\boldsymbol{s''} = \boldsymbol{s} + t''\boldsymbol{v} \ \wedge \\
s_x''^2 + s_y''^2 > D^2 \ \wedge \\
\boldsymbol{v'_o} = \mathit{kb2d}(s_x,s_y,v_{ox},v_{oy},v_{ix},v_{iy},e) \ \wedge \\
(t', \boldsymbol{v''_o}) = \mathit{recovery}(s_x,s_y,v_{ox},v_{oy},v_{ix},v_{iy},t'',e) \ \wedge \\
\boldsymbol{v'} = \boldsymbol{v'_o} - \boldsymbol{v_i} \ \wedge \ \boldsymbol{v''} = \boldsymbol{v''_o} - \boldsymbol{v_i} \ \wedge \\
\boldsymbol{s'} = \boldsymbol{s} + t'\boldsymbol{v'} \ \wedge \ 0 < t' < t'' \ \wedge \\
\boldsymbol{v'_o} \neq \boldsymbol{0} \\
\Rightarrow \\
\quad \mathit{separation?}(\boldsymbol{s'},\boldsymbol{v''}) \ \wedge \\
\quad \boldsymbol{s'} + (t'' - t')\boldsymbol{v''} = \boldsymbol{s''}.
\end{array}
$$

Proof (Sketch). We proof that the time of switch t' defines a point $\boldsymbol{s'} = \boldsymbol{s} + t'\boldsymbol{v'}$ that lies on both the tangent line L and the tangent line that passes through the target point $\boldsymbol{s''}$. We conclude by using Theorem 1 and the fact that a recovery maneuver is a resolution maneuver for the conflicting situation described by the target point $\boldsymbol{s''}$, relative velocity vector $-\boldsymbol{v}$, and arrival time t''. □

4.5 Coordinated Maneuvers

Let $\boldsymbol{v'_o}$ and $\boldsymbol{v'_i}$ be the resolutions computed by `kb2d` for the ownship and traffic aircraft, respectively. If neither the resolution velocity vector $\boldsymbol{v'_o}$ nor $\boldsymbol{v'_i}$ are null, then they are coordinated, i.e., they verify Formula 5.

Theorem 4 (`kb2d_coordinated`). *For all* $\boldsymbol{s}$, $\boldsymbol{v} = \boldsymbol{v_o} - \boldsymbol{v_i}$, $T > 0$, $D > 0$, $\boldsymbol{v'} = \boldsymbol{v'_o} - \boldsymbol{v'_i}$, $e = \pm 1$,

$$
\begin{array}{l}
\mathit{conflict?}(\boldsymbol{s},\boldsymbol{v},T) \ \wedge \\
s_x^2 + s_y^2 > D^2 \ \wedge \\
\boldsymbol{v'_o} = \mathit{kb2d}(s_x,s_y,v_{ox},v_{oy},v_{ix},v_{iy},e) \ \wedge \\
\boldsymbol{v'_i} = \mathit{kb2d}(-s_x,-s_y,v_{ix},v_{iy},v_{ox},v_{oy},e) \ \wedge \\
\boldsymbol{v'_o} \neq \boldsymbol{0} \ \wedge \ \boldsymbol{v'_i} \neq \boldsymbol{0} \\
\Rightarrow \\
\quad \neg \ \mathit{conflict?}(\boldsymbol{s},\boldsymbol{v'},T).
\end{array}
$$

Proof (Sketch). Notice that the time of loss of separation is the same for both aircraft and that the relative positions computed by each aircraft are opposite, that is; the relative position of ownship is $\boldsymbol{s}$ and that of traffic is $-\boldsymbol{s}$. Hence, it is not difficult to see that KB2D always selects the same e to compute a tangent trajectory for both aircraft. Then, we show that $\boldsymbol{v}'_{\boldsymbol{o}}$ and $\boldsymbol{v}'_{\boldsymbol{i}}$ are line solutions for the same e, i.e., they satisfy Formula 12. We conclude by using Lemma 2 (`coordinated_line`). □

5 Conclusions and Future Work

We proposed a pairwise resolution and recovery algorithm for two dimensional air traffic conflicts. The algorithm, which we call KB2D, computes maneuvers that are correct, i.e., they guarantee traffic separation, and coordinated, i.e, the separation is guaranteed even if both aircraft simultaneously maneuver to solve the conflict. This coordination is achieved without explicit exchange of intent information.

KB2D is strongly inspired on KB3D [2], a three dimensional algorithm designed and formally verified by the Formal Methods Group at NIA and NASA Langley Research Center. In contrast to KB3D, resolution maneuvers computed by KB2D are geometrically optimal for simultaneous changes of heading and ground speed. KB2D, like KB3D, uses the geometrical analysis proposed by K. Bilimoria in [1]. However, KB2D and Bilimoria's geometric optimization algorithm are completely different solutions to the same problem. In particular, we intentionally avoid the use of trigonometric functions in KB2D. Therefore, we conjecture that our algorithm is less susceptible to numerical instability due to floating point errors.

Using computer algebra tools, F. Kirchner has solved the optimality problem for a 3D geometry [6]. A purely geometrical optimization for a 3D airspace may not be practical since horizontal and vertical velocity changes compare differently in terms of fuel efficiency, passenger comfort, and aircraft performance. Future work includes the development of a verification framework for optimal CD&R for a parametric cost function.

Finally, we stress the fact that KB2D has been mechanically verified in PVS [11]. The PVS development consists of 37 lemmas and 800 lines of specification. Given the critical nature of conflict detection and resolution systems, we believe that formal verification of CD&R algorithms is a necessary step toward the safety analysis of new air traffic management concepts.

References

1. Bilimoria, K.: A geometric optimization approach to aircraft conflict resolution. In: Guidance, Navigation, and Control Conference, vol. AIAA 2000-4265, Denver, CO (August 2000)
2. Dowek, G., Geser, A., Muñoz, C.: Tactical conflict detection and resolution in a 3-D airspace. In: Proceedings of the 4th USA/Europe Air Traffic Management R&DSeminar, ATM 2001, Santa Fe, New Mexico, 2001. A long version appears as report NASA/CR-2001-210853 ICASE Report No. 2001-7 (2001)

3. Dowek, G., Muñoz, C., Carreño, V.: Provably safe coordinated strategy for distributed conflict resolution. In: Dowek, G. (ed.) Proceedings of the AIAA Guidance Navigation, and Control Conference and Exhibit 2005, San Francisco, California, AIAA-2005-6047 (2005)
4. Geser, A., Muñoz, C., Dowek, G., Kirchner, F.: Air Traffic Conflict Resolution and Recovery. ICASE Report 2002-12, ICASE, Langley Research Center (2002)
5. Hoekstra, J., Ruigrok, R., van Gent, R., Visser, J., Gijsbers, B., Valenti, M., Heesbeen, W., Hilburn, B., Groeneweg, J., Bussink, F.: Overview of NLR free flight project 1997-1999. Technical Report NLR-CR-2000-227, National Aerospace Laboratory (NLR) (May 2000)
6. Kirchner, F.: Optimal unconstrained solution to conflict resolution in 3-d airspace. Manuscript (2001)
7. Kuchar, J., Yang, L.: A review of conflict detection and resolution modeling methods. IEEE Transactions on Intelligent Transportation Systems 1(4), 179–189 (2000)
8. Maddalon, J., Butler, R., Geser, A., Muñoz, C.: Formal verification of a conflict resolution and recovery algorithm. Technical Report NASA/TP-2004-213015, NASA Langley Research Center, NASA LaRC, Hampton VA 23681-2199, USA (April 2004)
9. Muñoz, C., Mayero, M.: Real automation in the field. ICASE Interim Report 39 NASA/CR-2001-211271, NASA Langley Research Center, NASA Langley Research Center (December 2001)
10. NASA: Concept definition for Distributed Air/Ground Traffic Management (DAG-TM), version 1.0. Advanced Air Transportation Technologies (AATT) Project. NASA Ames and Langley Research Centers (1999)
11. Owre, S., Rushby, J.M., Shankar, N.: PVS: A Prototype Verification System. In: Kapur, D. (ed.) Automated Deduction - CADE-11. LNCS, vol. 607, pp. 748–752. Springer, Heidelberg (1992)
12. RTCA: Final report of the RTCA board of directors' select committee on free flight. Technical Report Issued 1-18-95, RTCA, Washington, DC (1995)
13. Di Vito, B.L.: Manip User's Guide, Version 1.1. NASA Langley Research Center, Hampton, Virginia (February 18, 2003)