Mixed-integer nonlinear and continuous optimization formulations for aircraft conflict avoidance via heading and speed deviations

Sonia Cafieri, Andrew Conn, Marcel Mongeau

Mixed-integer nonlinear and continuous optimization formulations for aircraft conflict avoidance via heading and speed deviations*

Sonia Cafieri, Andrew R. Conn, Marcel Mongeau

Abstract

We introduce two new optimization models for the aircraft conflict avoidance problem that aims at issuing decisions on both speed and heading-angle deviations to keep aircraft pairwise separated by a given separation distance. The first model is a new mixed-integer nonlinear formulation. The second model is a continuous optimization formulation, less typical in aircraft conflict avoidance. The advantages of the two models are combined within a three-phase method that we propose to solve the problem to global optimality. Computational experiments on various instances from the literature yield very promising results, and show the effectiveness of the proposed models and of the three-phase solution approach.

1 Introduction

We address a real-world application arising in transportation engineering, and more specifically a traffic conflict avoidance problem, where it is essential to maintain vehicles pairwise separated by a threshold distance. The pairwise separation between moving objects arises as a fundamental component in a number of applications spanning from transportation engineering, to robotics, computational geometry, computer graphics, and computer-aided design ([5]). In transportation engineering, traffic conflict avoidance applies to various modes of transportation; recent applications concern autonomous vehicles ([1]). In robotics, motion planning optimization ([19]) must encompass collision avoidance among robots. This can be modeled considering a safety zone around each robot, to be avoided ([8]). In computational geometry, the problem we consider could be seen as a special case of collision detection of simple 2D geometrical objects with rectilinear motion as in [16].

In this paper we consider in particular the context of air traffic management and we address the aircraft conflict avoidance problem. The objective is to keep vehicles (in our case, aircraft) pairwise separated by a given standard distance all along their trajectories; the loss of such a separation leading to a

*In memory of our dearest friend Andy Conn

conflict. This separation is an essential safety issue in air traffic management (a hard constraint). The problem is in fact a feasibility problem. It is typically addressed as a proper optimization problem because there are secondary criteria (such as minimizing deviations with respect to initial trajectories), that are used by various authors to discriminate among possibly numerous feasible solutions. Aircraft separation is achieved in practice by modifying the aircraft headings, speeds or altitudes, or sometimes by combining some of these possible maneuvers. Decision variables of the optimization models from the literature are then related to the chosen maneuvers (see [9]).

Separation constraints depend continuously on time; the simplest way to deal with them is based on discretizing the time horizon and to impose pairwise separation at each time step. Some approaches discretize the decision variables as well (see [22] and references therein), while others rely on optimal control ([10]). Space discretization (where separation is built upon a set of relevant points of the 2D space) is also proposed in the literature, (see *e.g.*, [23]), where moreover a discrete set of turning angles is used. Such discretizations potentially yield very large combinatorial optimization problems. Other ways to model separation constraints rely on geometrical constructions or on analytical descriptions of aircraft motion. One such geometrical construction, proposed in [24] and extended later (*e.g.*, [2, 3]), is based on disks representing safety areas around aircraft (*shadows*), and on their interception. In this paper, we build on the analytical description of aircraft motion presented in [13], and exploited subsequently *e.g.*, in [12, 14, 15]. It is tailored to optimization models that do not rely on discretizing either time or decision variables. The resulting formulation of the separation constraints is however nonlinear.

Most mathematical optimization approaches of the literature involve mixed-integer (either linear or nonlinear) formulations ([12, 13, 14, 15, 24, 27, 28]). The only fully-continuous optimization approach, which is based on B-splines to build aircraft conflict-free trajectories, is proposed in [25]. It relies on transforming a semi-infinite programming formulation of the pairwise separation constraints into a single equality constraint. Only locally optimal solutions are obtained. Moreover, the authors could not achieve computing explicit derivatives, therefore resorting to finite-difference approximations.

As regards the possible types of separation maneuvers studied in the air-traffic conflict avoidance literature, most works consider either only speed regulation ([12, 13, 14, 15, 24, 27]), or only heading angle changes ([2, 4, 24, 25]). Combination of maneuvers is rarely considered, despite the interest of resorting to complementary maneuvers in an operational context. A single type of maneuver may indeed be insufficient to resolve all conflicts. Simply consider the (extreme) situation of face-to-face aircraft: speed regulation cannot evidently be enough. Moreover, it is possible to exhibit problem instances that sole heading-angle deviations will not resolve. Combining speed and heading angle deviations is considered in [4], leading to a formulation characterized by highly nonlinear trigonometric constraints. Multiple types of maneuvers are considered in [22], where the authors propose a maneuver-discretized approach, including altitude (flight level) changes. However, such altitude changes are rarely employed in

practice, as they are associated with several operational drawbacks. For this reason, the present paper does *not* consider altitude changes. Another approach based on discretization, combining speed and heading changes, is that of [23]. Speed and heading deviations are applied sequentially in a two-step solution approach in [14]. In [26], the authors propose an approach combining speed and heading angle deviations, that is extended in [17] to allow also flight level changes. This approach leads to mixed-integer quadratically-constrained problems, solved through a sequence of convex relaxations. As a result, the authors obtain global optima for several problem instances. In this paper, we consider the same maneuver combination as in [26] (speed and heading angle deviations); we propose two new models, allowing us to obtain globally optimal solutions for instances not solved in [17, 26].

Note finally that, in an operational context, each aircraft needs to recover its original trajectory after being deviated. This subsequent problem is out of the scope of the present paper; let us simply refer the reader to [4] for a simple approach to trajectory recovery (it involves solving a convex unconstrained quadratic problem for each pair of aircraft).

The main contribution of this paper is the introduction of two new models for aircraft conflict resolution relying on both aircraft heading and speed deviations, and stemming from two different ways to address the conditional separation constraints characterizing the problem. The inherent combinatorics in conflict avoidance problems leads naturally to conditional constraints: if two vehicles are converging spatially, then their smallest inter-distance must be bounded below at all time. Moreover, the separation constraint that must be imposed to avoid conflicts often involves nonconvex, nonlinear constraints. We first introduce a mixed-integer nonlinear programming (MINLP) formulation and a continuous optimization model that is based on an original penalty-function strategy introduced in [11] to deal with conditional constraints. The MINLP model has a continuous relaxation that is not convex, and the continuous-optimization formulation is a constrained non-convex problem. Both models involve a number of variables and constraints that is quadratic in the number of aircraft.

As a second contribution, we propose an effective three-phase solution approach whose first phase is the solution of the continuous optimization problem, and the other two phases rely on the MINLP, thus enabling us to obtain exact global solutions.

The remainder of this paper is structured as follows. Section 2 presents the aircraft conflict avoidance problem, and proposes a reformulation of the angle-related functions involved in the two models we are introducing. Sections 3 and 4 introduce respectively these two models: a mixed-integer nonlinear optimization formulation, and a continuous optimization model based on a penalty function. Section 5 reports numerical experiments on the aircraft conflict avoidance problem, performed using each of the two models. Section 6 proposes an effective global-optimization approach combining the advantages of the two models proposed above. Finally, conclusions are drawn in Section 7.

2 Aircraft conflict avoidance: Towards new models

This section first briefly defines the problem to be solved and explains the important separation constraints, then it introduces some linearizations that will be used in the two new optimization models that we are proposing.

2.1 Problem statement and separation constraints

This subsection defines the air traffic conflict avoidance problem, and presents how we model the aircraft separation constraints in a continuous time interval.

In the sequel, for any given vector, $\mathbf{x}$, the notation x_i represents its i^{th} component. We assume that we are given an index set, $A = \{1, 2, \ldots, n\}$, corresponding to aircraft flying at the same constant altitude (flight level), on straight-line segment trajectories. The aim is to keep aircraft pairwise separated during the time horizon these aircraft are monitored. Two aircraft are separated if their relative distance is greater than or equal to a given standard separation distance (commonly, 5 nautical miles (NM) for commercial en-route flights). A *conflict* corresponds to the loss of such a separation. For each aircraft $i \in A$, we know its initial position, $(x_i^0, y_i^0) \in \mathbb{R}^2$, and its velocity (*i.e.*, its heading angle, ϕ_i, and its speed, v_i). Given this configuration, aircraft conflict avoidance consists in issuing separation maneuvers (simultaneously, at time $t = 0$) so as to ensure that the aircraft will remain separated.

The main difficulty in aircraft conflict avoidance problems comes from the separation constraint that must hold for each pair $i, j \in A$ $(i < j)$ of aircraft:

$$||\mathbf{x}_{ij}(t)|| \geq d \qquad \forall\, t \geq 0, \tag{1}$$

where d is the minimum required separation distance, $\mathbf{x}_{ij}(t) = \mathbf{x}_i(t) - \mathbf{x}_j(t) \in \mathbb{R}^2$ is a vector representing the position of aircraft i *relative* to that of aircraft j (position of aircraft i in the Cartesian coordinate system whose origin is the position of aircraft j), and we use the Euclidean norm. This constraint represents in general a non-convex set. It is similar to the constraint requiring circles not to overlap in packing / cutting problems (with a norm $||\cdot||_A$, where A is a positive definite matrix A different from the identity matrix, for the ellipse case ([21])). A further difficulty here is that it must hold *for all* time t (our objects are vehicles, and move). As mentioned in Section 1, the latter difficulty has been addressed in various ways in the literature. Here we choose the simple "parabola approach" of [13] to address this continuum of constraints. More precisely, assuming uniform motion law, the relative distance of aircraft i and j is expressed as: $\mathbf{x}_{ij}(t) = \mathbf{x}_{ij}^0 + \mathbf{v}_{ij}t$, for all t, where $\mathbf{x}_{ij}^0$ is the aircraft relative initial position, and $\mathbf{v}_{ij}$ is their relative speed. Substituting into (1) and squaring, one obtains, introducing the notation $f_{ij}(t)$:

$$f_{ij}(t) = \|\mathbf{v}_{ij}\|^2 t^2 + 2\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij} t + \|\mathbf{x}_{ij}^0\|^2 - d^2 \geq 0, \qquad \forall\, t \geq 0. \tag{2}$$

Hence, for each pair of aircraft i and j, satisfying constraint (1) amounts to ensuring that the simple univariate quadratic *separation function*, $f_{ij}(t)$, is always

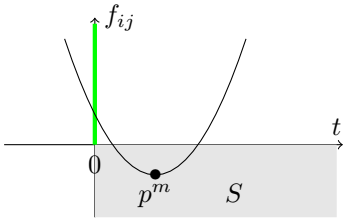

Figure 1: The separation function for two initially-separated, converging aircraft and the forbidden set S

non-negative. Remark that the vectors $\mathbf{x}_{ij}^0$ and $\mathbf{v}_{ij}$ will be straightforwardly computable explicitly in terms of the decision variables and of the input data of the problem. We assume that aircraft are separated at $t = 0$ (verified in a pre-processing stage). Without loss of generality, we suppose $\mathbf{v}_{ij} \neq 0$[1]. As in [13], since the quadratic function f_{ij} is strictly convex, one can easily verify that the condition (1) is equivalent to:

$$\textbf{if} \quad t_{ij}^m > 0, \quad \textbf{then} \quad f_{ij}(t_{ij}^m) \ \geq \ 0, \tag{3}$$

where t_{ij}^m is the minimizer of f_{ij}. As $t_{ij}^m = \dfrac{-\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij}}{\|\mathbf{v}_{ij}\|^2}$, finally the separation constraint (1) is equivalent to the following logical condition:

$$\textbf{if} \quad t_{ij}^m > 0, \quad \textbf{then} \quad \|\mathbf{v}_{ij}\|^2(\|\mathbf{x}_{ij}^0\|^2 - d^2) - (\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij})^2 \ \geq \ 0. \tag{4}$$

Figure 1 provides an intuitive interpretation of the separation condition (3): the parabola gives, at every time t, the value of the separation function f_{ij} (which is positive if and only if aircraft i and j are separated by a distance of at least d), *i.e.*, it gives the square of the distance between aircraft i and j plus d^2, at any time t. One observes that, in the displayed example, initially (at time $t = 0$) the aircraft are separated (the curve goes through the green vertical half-line) but are converging (their inter-distance is decreasing). The separation constraint therefore requires the "lowest" point of the parabola, $p^m := (t_{ij}^m, f_{ij}(t_{ij}^m))$, to lie outside the "forbidden" open fourth quadrant:

$$S = \{(t, f) \in \mathbb{R}^2 : t > 0 \text{ and } f < 0\}, \tag{5}$$

displayed in grey on Figure 1.

2.2 Towards new optimization models

We aim at proposing mathematical models involving decision variables controlling both speed and heading angle of each aircraft. We first present our main

[1] Otherwise, the fact that i and j are separated at $t = 0$ is equivalent to having i and j separated for all $t \geq 0$; equivalently, setting $\mathbf{v}_{ij}$=0 in (2) boils down to the initial separation $\|\mathbf{x}_{ij}^0\| \geq d$.

decision variables and the (nonlinear) way they are related to express the separation conditions. Then, we propose some reformulations to avoid explicitly handling trigonometric functions.

For each aircraft $i \in A$, let q_i be its speed variation, and let θ_i be its heading-angle variation. More precisely, q_i is the proportion of variation of the aircraft original speed, v_i, so that the actual aircraft speed is $q_i v_i$, and θ_i is added (its value may be negative) to the original angle, ϕ_i. Both speed and angle variations are assumed to be bounded. If we suppose that the speed can be decreased by no more than 6% and increased by no more than 3%, (common operational and cost constraints are even stricter), we have that the actual aircraft speed is $q_i v_i \in [0.94 v_i, 1.03 v_i]$. We shall assume that the heading angles, θ_i, are bounded in the interval $[\underline{\theta_i}, \overline{\theta_i}]$, where $\underline{\theta_i} = -\pi/6$ and $\overline{\theta_i} = \pi/6$, as it is common practice in air traffic control for operational reasons.

Recall from Subsection 2.1 that in order to express the separation condition for a pair of aircraft $i, j \in A$ $(i < j)$, we suppose that their relative initial position, $\mathbf{x}_{ij}^0 \in \mathbb{R}^2$, is given

$$\mathbf{x}_{ij}^0 := \begin{pmatrix} x_i^0 - x_j^0 \\ y_i^0 - y_j^0 \end{pmatrix}$$

and, taking into account possible speed and heading angle variations, their relative velocity, $\mathbf{v}_{ij} \in \mathbb{R}^2$, is:

$$\mathbf{v}_{ij} := \begin{pmatrix} \cos(\phi_i + \theta_i)(q_i v_i) - \cos(\phi_j + \theta_j)(q_j v_j) \\ \sin(\phi_i + \theta_i)(q_i v_i) - \sin(\phi_j + \theta_j)(q_j v_j) \end{pmatrix}. \tag{6}$$

Here again, this is similar to rotating objects in packing / cutting problems, when the speed variable q_i is set to the constant value 1.

To avoid explicit trigonometric functions in our optimization models, we define for each aircraft $i \in A$:

$$c_i := \cos(\phi_i + \theta_i) \qquad \text{and} \qquad s_i := \sin(\phi_i + \theta_i). \tag{7}$$

In order to linearize (6), we further define ω_i and π_i for each $i \in A$ as follows:

$$\omega_i := c_i q_i v_i, \tag{8}$$

$$\pi_i := s_i q_i v_i, \tag{9}$$

so that

$$\mathbf{v}_{ij} = \begin{pmatrix} \omega_i - \omega_j \\ \pi_i - \pi_j \end{pmatrix}, \quad i, j \in A : i < j. \tag{10}$$

In both optimization models to be introduced in Sections 3 and 4, our main decision-variable vectors will be: q, ω and π, (and $\mathbf{v}$ remains as an auxiliary decision-variable vector). Let us consider a particular aircraft $i \in A$. The variable q_i controls the variation of the speed of aircraft i. The other two variables, ω_i and π_i, although they both depend on q_i, account for variations in the heading angle of aircraft i. Note that, as $c_i^2 + s_i^2 = 1$, it follows that:

$$\omega_i^2 + \pi_i^2 = (q_i v_i)^2 (c_i^2 + s_i^2) = (q_i v_i)^2. \tag{11}$$

Thus, in fact we only have two degrees of freedom for each aircraft $i \in A$. Equation (11) is a nonlinear equality constraint that is included in our optimization models for each $i \in A$.

Remark: Going from the pair of variables (q_i, θ_i) to the triple (q_i, ω_i, π_i) is crucial. As already mentioned, it still involves only two degrees of freedom for each aircraft $i \in A$, taking into account equation (11). Expressing our optimization problem in terms of the above triple has the strong advantage of avoiding trigonometric functions, and of removing the bilinear product $c_i q_i$ and $s_i q_i$, thus linearizing (6) into (10), thereby enabling a more efficient use of optimization solvers.

We introduce in the following two sections two new models for aircraft conflict avoidance based on the above setting: a mixed-integer non-linear formulation and a continuous penalty-based nonlinear model. Both formulations are constructed using the main decision variables, q_i, ω_i and π_i, $i \in A$, and the quadratic equality constraints (11) linking the above variables (together with the auxiliary variables $\mathbf{v}_{ij}$, $i, j \in A$ $(i < j)$, and their defining constraints). The two formulations address differently the crucial issue of the logical separation condition

$$\textbf{if} \quad t_{ij}^m > 0, \quad \textbf{then} \quad \|\mathbf{v}_{ij}\|^2(\|\mathbf{x}_{ij}^0\|^2 - d^2) - (\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij})^2 \ \geq \ 0 \tag{12}$$

that one must consider for every possible pair of aircraft, $i, j \in A, i < j$.

3 MINLP formulation

The first model that we introduce for aircraft conflict avoidance is an MINLP, in which the separation constraint (12), for each $i, j \in A, i < j$, is modelled via the *complementary formulation* (see, *e.g.*, [6]) of logical constraints:

$$t_{ij}^m(y_{ij} - 1) \geq 0, \tag{13}$$

$$y_{ij} \left(\ \|\mathbf{v}_{ij}\|^2 \left(\|\mathbf{x}_{ij}^0\|^2 - d^2\right) - (\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij})^2 \ \right) \geq 0, \tag{14}$$

$$y_{ij} \in \{0, 1\}, \tag{15}$$

where y_{ij} is an extra auxiliary binary decision variable. Remark by the way that constraint (12) is equivalent to the following implication (without a strict inequality in the condition on the sign of t_{ij}^m):

$$\textbf{if} \quad t_{ij}^m \geq 0, \quad \textbf{then} \quad \|\mathbf{v}_{ij}\|^2(\|\mathbf{x}_{ij}^0\|^2 - d^2) - (\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij})^2 \ \geq \ 0, \tag{16}$$

since by assumption, aircraft i and j are separated when $t_{ij}^m = 0$ (it is rather the form (16) that is seen in the literature on aircraft conflict avoidance).

Equations (13)-(15) model the crucial separation condition that characterizes the conflict resolution problem. An optimization formulation comprises also an appropriate objective to be defined. A natural objective is to minimize the angle and speed variations with respect to the given initial conditions ϕ_i and

v_i ($i \in A$). However, this requires expliciting each angle change in the objective function through trigonometric functions, losing thereby the computational benefit of our model for angle change. Note first that, as in air-traffic practice the allowed angle deviation, θ_i, is a bounded variable, one has for each $i \in A$:

$$\underline{a} \leq \cos(\theta_i) \leq 1 \qquad \text{and} \qquad -\bar{b} \leq \sin(\theta_i) \leq \bar{b}, \tag{17}$$

for some values of $\underline{a} \geq 0$ and $\bar{b} \leq 1$. In practice, each $|\theta_i|$ is typically bounded by $\pi/6$; so that one generally sets $\underline{a} = \frac{\sqrt{3}}{2}$ and $\bar{b} = \frac{1}{2}$. Let us now introduce, for each $i \in A$, an auxiliary variable, b_i, defined as an upper bound on $|\sin(\theta_i)|$:

$$-\bar{b} \;\leq\; -b_i \leq \sin(\theta_i) \leq b_i \;\leq\; \bar{b}. \tag{18}$$

In order to minimize angle deviations, one then aims at minimizing the b_i's. Thus, in order to minimize both speed and angle deviations, one may consider the following objective function

$$(1-\lambda) \sum_{i \in A} (1-q_i)^2 + \lambda \sum_{i \in A} b_i, \tag{19}$$

where $\lambda \in [0,1]$ is a user-defined weighting parameter that can be adjusted to reach a good compromise between speed and angle deviations.

Linking constraints

Introducing the auxiliary-variable vector b requires defining constraints to link these new variables to the original decision-variable vectors q, ω and π. To that aim, let $i \in A$, and let us consider the trigonometric identity:

$$\cos(\phi_i + \theta_i) = \cos\phi_i \cos\theta_i - \sin\phi_i \sin\theta_i,$$

together with $\underline{a} \leq \cos(\theta_i) \leq 1$ and $-b_i \leq \sin(\theta_i) \leq b_i$. Considering the four cases that can arise, depending on the signs of $\sin\phi_i$ and $\cos\phi_i$, and recalling that each (given) heading angle, ϕ_i, satisfies $-\pi \leq \phi_i < \pi$, one obtains the following relationship:

$$\min\left(\underline{a}\cos\phi_i, \cos\phi_i\right) - b_i\,|\sin\phi_i| \;\leq\; c_i \;\leq\; \max\left(\underline{a}\cos\phi_i, \cos\phi_i\right) + b_i\,|\sin\phi_i| \tag{20}$$

Similarly, the trigonometric identity:

$$\sin(\phi_i + \theta_i) = \sin\phi_i \cos\theta_i + \cos\phi_i \sin\theta_i$$

yields:

$$\min\left(\underline{a}\sin\phi_i, \sin\phi_i\right) - b_i\,|\cos\phi_i| \;\leq\; s_i \;\leq\; \max\left(\underline{a}\sin\phi_i, \sin\phi_i\right) + b_i\,|\cos\phi_i|, \tag{21}$$

where c_i and s_i are the intermediate cosine and sine variables defined by (7). From the definitions (8) and (9) of ω_i and π_i, we then obtain the following

linking constraints:

$$\begin{aligned}
\omega_i &\geq (\min(\underline{a}\cos\phi_i, \cos\phi_i) - b_i\,|\sin\phi_i|)\; q_i v_i, && (22)\\
\omega_i &\leq (\max(\underline{a}\cos\phi_i, \cos\phi_i) + b_i\,|\sin\phi_i|)\; q_i v_i, && (23)\\
\pi_i &\geq (\min(\underline{a}\sin\phi_i, \sin\phi_i) - b_i\,|\cos\phi_i|)\; q_i v_i, && (24)\\
\pi_i &\leq (\max(\underline{a}\sin\phi_i, \sin\phi_i) + b_i\,|\cos\phi_i|)\; q_i v_i. && (25)
\end{aligned}$$

Valid (bound-constraint) inequalities

Let us now determine lower and upper bounds for the decision variables ω_i and π_i, $i \in A$, in order to render a branch-and-bound solution procedure more efficient. Let us introduce the following notation, related to inequalities (20) and (21):

$$\begin{aligned}
\underline{c_i}(b_i) &:= \min(\underline{a}\cos\phi_i, \cos\phi_i) - b_i\,|\sin\phi_i|, && (26)\\
\overline{c_i}(b_i) &:= \max(\underline{a}\cos\phi_i, \cos\phi_i) + b_i\,|\sin\phi_i|, && (27)\\
\underline{s_i}(b_i) &:= \min(\underline{a}\sin\phi_i, \sin\phi_i) - b_i\,|\cos\phi_i|, && (28)\\
\overline{s_i}(b_i) &:= \max(\underline{a}\sin\phi_i, \sin\phi_i) + b_i\,|\cos\phi_i|. && (29)
\end{aligned}$$

Using the bounds on the variable q_i, $\underline{q_i} \leq q_i \leq \overline{q_i}$ (where, as mentioned above, commonly $\underline{q_i} = 0.94$ and $\overline{q_i} = 1.03$), and following the possible cases for the signs of (the given data) $\cos\phi_i$ and $\sin\phi_i$ in the expressions of $\underline{c_i}(b_i)$ and $\overline{c_i}(b_i)$ above, we obtain, for each aircraft $i \in A$, the bound-constraint valid inequality:

$$\underline{\omega_i} \leq \omega_i \leq \overline{\omega_i}, \tag{30}$$

where

$$\underline{\omega_i} = \begin{cases} \underline{c_i}(\bar{b})\,\underline{q_i}\,v_i, & \text{if } \underline{c_i}(\bar{b}) \geq 0, \\ \underline{c_i}(\bar{b})\,\overline{q_i}\,v_i, & \text{otherwise,} \end{cases} \quad \text{and} \quad \overline{\omega_i} = \begin{cases} \overline{c_i}(\bar{b})\,\underline{q_i}\,v_i, & \text{if } \overline{c_i}(\bar{b}) \leq 0, \\ \overline{c_i}(\bar{b})\,\overline{q_i}\,v_i, & \text{otherwise.} \end{cases}$$

Recall that v_i, $\underline{c_i}(\bar{b})$ and $\overline{c_i}(\bar{b})$ are constants (the latter will be denoted simply $\underline{c_i}$ and $\overline{c_i}$ respectively in the sequel), and the speed bounds, $\underline{q_i}$ and $\overline{q_i}$, are always positive.

In an analogous manner, we obtain for π_i:

$$\underline{\pi_i} \leq \pi_i \leq \overline{\pi_i}, \tag{31}$$

where

$$\underline{\pi_i} = \begin{cases} \underline{s_i}\,\underline{q_i}\,v_i, & \text{if } \underline{s_i} \geq 0, \\ \underline{s_i}\,\overline{q_i}\,v_i, & \text{otherwise,} \end{cases} \quad \text{and} \quad \overline{\pi_i} = \begin{cases} \overline{s_i}\,\underline{q_i}\,v_i, & \text{if } \overline{s_i} \leq 0, \\ \overline{s_i}\,\overline{q_i}\,v_i, & \text{otherwise,} \end{cases} \tag{32}$$

with the notation: $\underline{s_i} := \underline{s_i}(\bar{b})$, and $\overline{s_i} := \overline{s_i}(\bar{b})$.

The **MINLP formulation** can be summarized as follows:

$$\min_{q,\omega,\pi,t^m,\mathbf{v},b,y} (1-\lambda)\sum_{i\in A}(1-q_i)^2 + \lambda\sum_{i\in A} b_i \tag{33}$$

s.t.

$$y_{ij}\left(\|\mathbf{v}_{ij}\|^2\left(\|\mathbf{x}_{ij}^0\|^2 - d^2\right) - (\mathbf{x}_{ij}^0\cdot\mathbf{v}_{ij})^2\right) \geq 0, \qquad i,j\in A: i<j \tag{34}$$

$$t_{ij}^m\|\mathbf{v}_{ij}\|^2 = -\mathbf{x}_{ij}^0\cdot\mathbf{v}_{ij}, \qquad i,j\in A: i<j \tag{35}$$

$$t_{ij}^m(y_{ij}-1)\geq 0, \qquad i,j\in A: i<j \tag{36}$$

$$\mathbf{v}_{ij} = \begin{pmatrix}\omega_i-\omega_j\\ \pi_i-\pi_j\end{pmatrix}, \qquad i,j\in A: i<j \tag{37}$$

$$\omega_i^2+\pi_i^2 = (q_i v_i)^2, \qquad i\in A \tag{38}$$

$$\omega_i \geq \left(\min\left(\underline{a}\cos\phi_i, \cos\phi_i\right) - b_i\,|\sin\phi_i|\right)\, q_i v_i, \qquad i\in A \tag{39}$$

$$\omega_i \leq \left(\max\left(\underline{a}\cos\phi_i, \cos\phi_i\right) + b_i\,|\sin\phi_i|\right)\, q_i v_i, \qquad i\in A \tag{40}$$

$$\pi_i \geq \left(\min\left(\underline{a}\sin\phi_i, \sin\phi_i\right) - b_i\,|\cos\phi_i|\right)\, q_i v_i, \qquad i\in A \tag{41}$$

$$\pi_i \leq \left(\max\left(\underline{a}\sin\phi_i, \sin\phi_i\right) + b_i\,|\cos\phi_i|\right)\, q_i v_i, \qquad i\in A \tag{42}$$

$$\underline{q_i}\leq q_i\leq \overline{q_i}, \qquad i\in A \tag{43}$$

$$\underline{\omega_i}\leq \omega_i\leq \overline{\omega_i}, \qquad i\in A \tag{44}$$

$$\underline{\pi_i}\leq \pi_i\leq \overline{\pi_i}, \qquad i\in A \tag{45}$$

$$0\leq b_i\leq \bar{b}, \qquad i\in A \tag{46}$$

$$y_{ij}\in\{0,1\}, \qquad i,j\in A: i<j. \tag{47}$$

Remarks:

1. The number of variables is $\mathcal{O}(\frac{3n^2}{2})$ (among which $\frac{n(n-1)}{2}$ variables are binary), and the number of constraints is $\mathcal{O}(2n^2)$.

2. Constraints (39) to (42) are bilinear constraints since, as $\cos(\phi_i)$ and $\sin(\phi_i)$ are known input data, their sign, and thereby the maxima and minima involved, are known in advance.

3. Replacing constraints (35) and (36) with the equivalent constraints:

$$(\mathbf{x}_{ij}^0\cdot\mathbf{v}_{ij})(y_{ij}-1)\leq 0 \qquad i,j\in A: i<j, \tag{48}$$

   one can eliminate the optimization variables t_{ij}^m's and the highly non-linear constraints (35) from the formulation.

4. In our implementation, we do not use the decision variables $\mathbf{v}_{ij}$, but rather auxiliary variables representing $\|\mathbf{v}_{ij}\|^2$ and $\mathbf{x}_{ij}^0\cdot\mathbf{v}_{ij}$, $i,j\in A: i<j$, and we adapt thereby constraints (37).

4 Penalty continuous-optimization formulation

This section introduces our second formulation, named NLP-Penalty in the sequel, of the aircraft conflict avoidance problem. In the NLP-Penalty model,

the logical separation constraint (12) is formulated in a different way from that of the MINLP formulation. Whereas the latter models this logical constraint via the complementary formulation, the NLP-Penalty model addresses it via the continuous quadrant penalty formulation of logical constraints introduced in [11].

The formulation we are proposing involves the decision variables:

$$q_i, \omega_i, \pi_i,\ i \in A \qquad \text{and} \qquad (t_{ij}^m, f_{ij}^m), \mathbf{v}_{ij}, \quad i,j \in A : i < j, \tag{49}$$

where $t_{ij}^m = \text{argmin} f_{ij}(t)$, using the notation (2), and letting $f_{ij}^m := f_{ij}(t_{ij}^m)$. The corresponding feasible domain is a subset of

$$\mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \times (\mathbb{R}^2 \setminus S)^{\binom{n}{2}} \times \mathbb{R}^{\binom{n}{2}}. \tag{50}$$

Letting i, j be a pair of aircraft ($i, j \in A,\ i < j$), remark that the separation condition (12) then simply reads:

$$\textbf{if} \quad t_{ij}^m > 0, \quad \textbf{then} \quad f_{ij}^m \ \geq\ 0. \tag{51}$$

We reformulate the logical constraint (51) using the continuous quadrant penalty function $g_\beta : (t, f) \in \mathbb{R}^2 \to \mathbb{R}$ introduced in [11]:

$$g_\beta(t,f) = \begin{cases} 0, & \text{if} \quad t \leq 0 \ \text{ or } \ f \geq 0, \\ t^2, & \text{if} \quad 0 < t \leq -\dfrac{f}{3}, \\ \dfrac{-(t^2 + 6tf + f^2)}{8}, & \text{if} \quad -\dfrac{f}{3} < t < -3f, \\ f^2, & \text{if} \quad -\dfrac{t}{3} \leq f < 0, \end{cases} \tag{52}$$

(setting $\beta = 3$ as proposed by the authors), and illustrated on Figure 2. Summing up over all pairs $i, j \in A : i < j$, this yields the penalty function

$$\sum_{i,j \in A : i<j} g_\beta(t_{ij}^m, f_{ij}^m) \tag{53}$$

to be minimized.

Each of the term functions g_β is shown in [11] to be smooth. To give a geometrical interpretation of the function (53), remark first that the logical constraint (51) is equivalent to requiring: $p^m = (t_{ij}^m, f_{ij}^m) \in \mathbb{R}^2 \setminus S$, as illustrated by Figure 1, and where S is defined by (5). Moreover, since $\mathbf{v}_{ij}$, t_{ij}^m, and f_{ij}^m depend on q_i, ω_i, π_i and q_j, ω_j, π_j, these six optimization variables control the location of the point p^m on the graph of f_{ij}. Each term of the objective function (53) therefore guides a continuous optimization descent method towards the subset of the feasible set where the corresponding separation constraint is satisfied.

The NLP-Penalty formulation we are proposing penalizes constraint (51), for $i, j \in A : i < j$, and preserves the other constraints.

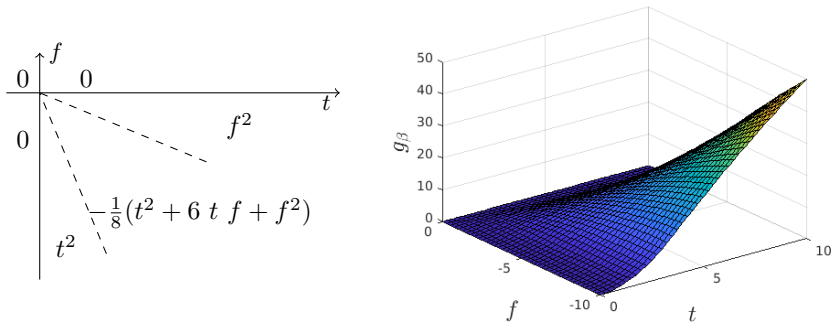

Figure 2: The piecewise quadratic penalty function g_β (left), and its 3D graph (right), when $\beta = 3$ (from [11]).

Similarly to the MINLP formulation introduced in Section 3, a minimization of the weighted function representing the deviations to aircraft speeds and headings, can be considered. Combining such a function with the penalty function (53) would lead to a weighted function involving a further weight (penalty) parameter to be tuned by the user. We choose in this study to focus solely on the penalty function term (53), in order to investigate more easily the behaviour of the penalty function in the context of conflict resolution. Moreover, the NLP-Penalty model that we propose will constitute one of the phases of resolution of the optimization problem minimizing maneuver deviations to global optimality presented in Section 6.

To summarize, the (continuous) **NLP-Penalty formulation** we are proposing reads as:

$$\min_{q,\omega,\pi,t^m,f^m,\mathbf{v}} \sum_{i,j\in A:i<j} g_\beta(t_{ij}^m, f_{ij}^m) \tag{54}$$

s.t.

$$f_{ij}^m \|\mathbf{v}_{ij}\|^2 = \|\mathbf{v}_{ij}\|^2(\|\mathbf{x}_{ij}^0\|^2 - d^2) - (\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij})^2, \qquad i,j \in A : i < j \tag{55}$$

$$t_{ij}^m \|\mathbf{v}_{ij}\|^2 = -\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij}, \qquad i,j \in A : i < j \tag{56}$$

$$\mathbf{v}_{ij} = \begin{pmatrix} \omega_i - \omega_j \\ \pi_i - \pi_j \end{pmatrix}, \qquad i,j \in A : i < j \tag{57}$$

$$\omega_i^2 + \pi_i^2 = (q_i v_i)^2, \qquad i \in A \tag{58}$$

$$\underline{q_i} \le q_i \le \overline{q_i}, \qquad i \in A$$

$$\underline{\omega_i} \le \omega_i \le \overline{\omega_i}, \qquad i \in A \tag{59}$$

$$\underline{\pi_i} \le \pi_i \le \overline{\pi_i}, \qquad i \in A. \tag{60}$$

Remarks:

1. The number of (continuous) variables is $\mathcal{O}(\frac{3n^2}{2})$, and the number of constraints is $\mathcal{O}(\frac{3n^2}{2})$.

2. In our implementation, the f_{ij}^m's and the t_{ij}^m's do not in fact appear as auxiliary variables: they are explicitly replaced in the model using constraints (55) and (56). As for the MINLP model, we also do not use the decision variables $\mathbf{v}_{ij}$ (but rather auxiliary variables representing $\|\mathbf{v}_{ij}\|^2$ and $\mathbf{x}_{ij}^0 \cdot \mathbf{v}_{ij}$, $i, j \in A : i < j$), and we adapt thereby constraints (57).

3. Since the objective function of the NLP-penalty formulation does not involve angle-related deviations, the vector, b, of extra decision variables introduced for the MINLP does not intervene in the calculation of the bound constraints (59) and (60), but only its extreme values through $\bar{b}$, the upper bound on the $|\sin(\theta_i)|$'s.

4. This problem is bounded below (by zero). Moreover, it is feasible. Indeed, for each $i \in A$ one can choose $q_i = 1$, and ω_i, π_i satisfying (58) and corresponding to a null heading change ($\theta_i = 0$) in (7), (8) and (9). Then, for each pair $i, j \in A$, constraints (55) and (56) simply define the lowest separation-function value, f_{ij}^m, and its corresponding time, t_{ij}^m (or they yield the trivial equality $0 = 0$ in the case where $\mathbf{v}_{ij} = 0$).

As demonstrated in [11], the penalty function, g_β, features several desirable properties in view of its use with a state-of-the-art local optimization solver. However, the fact that each term function, $g_\beta : \mathbb{R}^2 \to \mathbb{R}$, *leans outwards* the (open fourth quadrant) set S (which corresponds to a non-separated pair of aircraft) does not imply that the objective function $\sum_{i,j \in A : i<j} g_\beta(t_{ij}^m, f_{ij}^m) : \mathbb{R}^{2\binom{n}{2}} \to \mathbb{R}$ leans outwards the Cartesian-product set $S^{\binom{n}{2}}$ ($\subseteq \mathbb{R}^{2\binom{n}{2}}$). Indeed, remark first that this objective function is not separable: each of the speed and angle-related variables q_i, ω_i, π_i corresponding to *one* aircraft $i \in A$ intervenes in *several* terms of the objective function (each aircraft i is involved in several pairs of aircraft). Moreover, this objective function is not convex, and may thereby feature local optima that do not satisfy the constraint(s) the penalty function aims at modelling. Finally, one must also take into account the remaining constraints (all the ones that are not penalized), which include nonlinear equality constraints such as (58), yielding thereby a nonconvex feasible domain.

The next two sections show that good results are obtained despite the above-mentioned potential difficulties.

5 Numerical results

This section presents the performance of the MINLP and the NLP-Penalty formulations of Sections 3 and 4, through numerical experiments.

We test our models on two types of benchmark instances: the Circle Problem (CP), and the Random Circle Problem (RCP). In both cases, instances are constructed as follows. Aircraft are uniformly positioned on the circumference of a circle, representing the observed airspace portion, and are given an initial speed and a heading to fly towards the opposite side of the circle. This kind

of instances are widely used in the literature [13, 14, 15, 17, 26]. We use the instances proposed by (and available via) [17, 26]. The radius of the circle is 200 nautical miles (NM). The CP instances are generated with initial identical speeds ($v_i = 500$ NM/h, $i \in A$), and with all aircraft heading towards the center of the circle, thus resulting in a highly symmetric problem. The RCP instances are such that aircraft initial speeds and headings are randomly deviated within specified ranges (486–594 NM/h for the speed, and $\pm\pi/6$ with respect to the heading towards the center, for the angle), resulting thereby in more realistic and less structured instances. The standard separation distance, d, to be respected between aircraft, is equal to 5 NM. Aircraft speeds and heading angles are bounded due to operational constraints. For each aircraft $i \in A$, we set the bounds on variables q_i to: $\underline{q_i} = 0.94$ and $\bar{q_i} = 1.03$, that correspond to a so-called *subliminal speed change* (the original aircraft speed is allowed to vary only between -6% and $+3\%$). We allow heading angles, θ_i, to vary between $-\pi/6$ and $+\pi/6$, and compute bounds on variables ω_i and π_i accordingly ($i \in A$). The β parameter of the continuous quadrant penalty function, g_β, is set to $\beta = 3$.

The proposed models are implemented using the `AMPL` ([20]) modeling language. Numerical tests are performed on a 2.66 GHz Intel Xeon (octo core) processor with 32 GB of RAM and Linux operating system.

We first present numerical results obtained by considering our MINLP model (Section 3). The problem is solved by running the MINLP solver `COUENNE` ([7]), version 0.5, with all its default values, except for its feasibility tolerance which we increase to 10^{-5}. As this is recommended only for small problems, we disable both the optimality-based bound tightening as well as the "aggressive" feasibility-based bound tightening options, that are generally computationally expensive. Tests are run with a time limit of 600 seconds.

Note that, in contrast with [26], whose objective function involves a fixed, arbitrary nonlinear combination of the speed and angle deviations, our model allows the user to decide a particular speed-angle trade off, expressed through the weighting parameter, λ in (33), or to compensate any unfortunate choice of units, for instance to scale the angle deviation (expressed in radiants, degrees, *etc.*) with the speed deviation (which can be in percentage as here, in mile per hour, or in knots). In our experiments, we set $\lambda = 10^{-6}$. The recourse to such a small value of λ is not surprising when one considers the difference in the scales of the two criteria, speed and angle deviations, that constitute the objective function (19). Indeed, a typical extreme speed deviation for an aircraft i corresponds to accelerating it up to a fraction $q_i = 1.03$ of its initial speed, which yields a corresponding penalty term value of $(1 - q_i)^2 = \frac{9}{10{,}000}$; whereas an extreme angle deviation involves a heading angle change of $|\theta_i| = \frac{\pi}{6}$ which yields a corresponding penalty term value of $b_i = \frac{1}{2}$: the latter is 3 to 4 orders of magnitude larger than the former.

Table 1 reports the results obtained solving our MINLP problem. The headings of the table are as follows: *Name*, name of the instance; n, number of aircraft; n_c, number of initial conflicts (conflicts resulting from no speed change and no heading deviation); n_{hthc}, number of initial head-to-head conflicts; *time*,

				MINLP ($\lambda = 10^{-6}$)	
Name	n	n_c	n_{hth}	time (s)	speed dev.
CP_4	4	6	2	0.056	3.4e-18
CP_5	5	10	0	0.080	4.5e-15
CP_6	6	15	3	0.232	4.5e-15
CP_7	7	21	0	0.256	4.5e-11
CP_8	8	28	4	0.248	8.5e-12
CP_9	9	36	0	0.224	4.1e-11
CP_10	10	45	5	0.408	2.5e-10
CP_11	11	55	0	0.420	9.2e-13
CP_12	12	66	6	0.372	4.0e-10
CP_13	13	78	0	1.096	2.8e-11
CP_14	14	91	7	1.012	1.9e-11
CP_15	15	105	0	2.764	1.4e-09
CP_16	16	120	8	1.700	5.1e-10
CP_17	17	136	0	2.304	8.1e-10
CP_18	18	153	9	2.788	4.7e-10
CP_19	19	171	0	3.672	2.7e-10
CP_20	20	190	10	tlim	0.00084

				MINLP ($\lambda = 10^{-6}$)	
Name	n	n_c	n_{hth}	time (s)	speed dev.
RCP_10_1	10	2	1	0.368	6.1e-12
RCP_10_2	10	3	1	0.276	4.6e-12
RCP_10_3	10	2	1	0.276	7.4e-13
RCP_10_4	10	1	1	1.172	1.2e-12
RCP_10_5	10	5	1	0.268	3.6e-11
RCP_10_6	10	4	1	0.488	5.6e-12
RCP_10_7	10	4	1	0.324	4.2e-12
RCP_10_8	10	4	1	0.512	8.4e-12
RCP_10_9	10	3	1	0.624	4.7e-12
RCP_10_10	10	0	0	0	0
RCP_20_1	20	8	2	2.320	1.1e-09
RCP_20_2	20	9	2	2.244	2.0e-10
RCP_20_3	20	13	2	3.860	3.5e-10
RCP_20_4	20	9	2	5.712	8.9e-06
RCP_20_5	20	12	3	2.264	5.5e-11
RCP_20_6	20	13	2	2.840	1.0e-11
RCP_20_7	20	9	2	2.476	2.8e-10
RCP_20_8	20	9	2	2.552	2.9e-11
RCP_20_9	20	19	4	2.100	7.4e-11
RCP_20_10	20	15	3	2.088	1.3e-10
RCP_30_1	30	35	1	25.02	5.1e-09
RCP_30_2	30	38	1	11.59	3.1e-09
RCP_30_3	30	46	1	47.13	8.5e-06
RCP_30_4	30	39	1	301.6	0.000214
RCP_30_5	30	36	2	10.31	3.6e-10
RCP_30_6	30	32	2	tlim	2.1e-05
RCP_30_7	30	18	1	13.89	8.4e-10
RCP_30_8	30	40	1	21.55	2.5e-10
RCP_30_9	30	41	2	24.42	6.5e-09
RCP_30_10	30	46	1	tlim	0.000776
RCP_30_11	30	34	2	tlim	0.000154
RCP_30_12	30	36	1	tlim	0.000914
RCP_30_13	30	30	1	tlim	7.7e-05
RCP_30_14	30	39	2	tlim	6.8e-05
RCP_30_15	30	30	1	tlim	0.000281

Table 1: Results on CP and RCP instances: MINLP, with $\lambda = 10^{-6}$. Global exact solutions.

computing time in seconds, and *speed dev.*, value of the speed deviation in the computed optimal solution (this later value will be useful in the tests of Section 6). One observes that our MINLP model allows us to find global optima for all the classical CP instances except for the largest one (n=20), including the ones involving $11 \leq n \leq 19$ aircraft for which only local optima could be found in [17, 26]. The computing time is always less than 4 seconds for these academic and highly symmetric instances, even when a large number of conflicts are to be solved. Moreover, all the 10- and 20-aircraft RCP instances are solved to global optimality, the computing time varying from 0 to a little more than 5 seconds (note that among the randomly-generated instances of [26], RCP_10_10 features no conflicts to be solved). For the, larger, 30-aircraft RCP instances, 7 out of 15 instances are solved to global optimality, while our MINLP ends with a time limit (*tlim* in Table 1) for the remaining cases. In such cases, a feasible solution is however always computed (no guarantee of global optimality).

Let us now concentrate on the solution of the asymmetrical, and more realistic, RCP instances using our NLP-Penalty model. We choose the `IPOPT` solver for NLPs ([29]), version 3.12, to solve the optimization problem. The inner linear algebra solver is MA57 ([18]), and we set the feasibility tolerance to 10^{-5}, exactly as for the MINLP solver. Note that if this local optimization approach attains the lower bound zero, then global optimality is achieved with respect to the penalty objective function (53). Such solutions correspond however only to feasible solutions for the MINLP problem (which aims at deciding angle and speed for each aircraft so as to minimize their deviations from original values). Recall moreover that our NLP-Penalty problem is a nonconvex constrained NLP; it can thereby feature multiple local minima. Further, the penalization approach is likely to yield local minima that violate the (penalized) separation constraints (referred to as *local infeasibility*). The strategy we propose to address these issues is to solve the problem (to local optimality), using randomly-generated starting points through a simple multistart heuristic. We first run the solver from the starting guess corresponding to no deviation in speed and no deviation in heading angle (*i.e.*, setting for all $i \in A$: $q_i = 1$, and values of ω_i and π_i corresponding to $\theta_i = 0$). Then, within a maximum of $N = 5$ trials, we generate new starting guesses choosing randomly, for each $i \in A$: q_i uniformly in $[0.94, 1.03]$, and ω_i, π_i such that the value of θ_i varies uniformly in $[-\pi/6, \pi/6]$. A trial converging towards an optimum with zero penalty value is said a *successful run*.

Table 2 reports the numerical results obtained using our NLP-Penalty model. The first four columns report the characteristics of the test instances as for Table 1; the following four columns report respectively the computing time, in seconds, for the successful run (*time success*); the total time, in seconds, obtained by summing up the computing times of all trials (*total time*); the number of trials (*nb trials*); and the speed deviation that corresponds to the computed solution (*speed dev.*). The speed deviation is computed *a posteriori*. One first observes that this local-optimization approach always yields, on all the considered test cases, global optimality with respect to the penalty objective function (53) (since the lower bound zero is attained). Moreover, for 32 instances out of 34 the solution is computed in one run only, while for the other ones no more than 2 runs are needed. Computing times are always below 6 seconds, except for one instance whose multiple starting-point runs yielded a total of 20 seconds. These results show the interest of the proposed continuous optimization approach.

6 Three-phase solution method

The proposed NLP-Penalty and MINLP formulations yield solutions that are not comparable: the former yields a locally-optimal solution while the latter ends with a global minimizer and this, for objective functions that are different. In this subsection, we propose an effective global optimization approach that attempts at exploiting the advantages of both models. It is a three-phase method

				NLP-Penalty			
Name	n	n_c	n_{hth}	time success (s)	total time (s)	nb trials	speed dev.
RCP_10_1	10	2	1	0.132	0.132	1	0.00088
RCP_10_2	10	3	1	0.160	0.160	1	0.00077
RCP_10_3	10	2	1	0.104	0.104	1	0.00155
RCP_10_4	10	1	1	0.464	0.464	1	0.00144
RCP_10_5	10	5	1	0.936	0.936	1	0.00081
RCP_10_6	10	4	1	0.048	0.048	1	0.00063
RCP_10_7	10	4	1	0.100	0.100	1	0.00479
RCP_10_8	10	4	1	0.264	0.264	1	0.00105
RCP_10_9	10	3	1	0.108	0.108	1	0.00065
RCP_10_10	10	0	0	0	0	0	0
RCP_20_1	20	8	2	3.092	3.092	1	0.00432
RCP_20_2	20	9	2	0.904	1.708	2	0.00057
RCP_20_3	20	13	2	0.244	0.244	1	0.00115
RCP_20_4	20	9	2	0.372	0.372	1	0.00060
RCP_20_5	20	12	3	0.492	0.492	1	0.00097
RCP_20_6	20	13	2	0.192	0.192	1	0.00059
RCP_20_7	20	9	2	1.456	1.456	1	0.00283
RCP_20_8	20	9	2	0.520	0.520	1	0.00060
RCP_20_9	20	19	4	0.340	0.340	1	0.00418
RCP_20_10	20	15	3	2.136	2.136	1	0.01182
RCP_30_1	30	35	1	3.108	3.108	1	0.00703
RCP_30_2	30	38	1	1.072	1.072	1	0.00354
RCP_30_3	30	46	1	0.896	0.896	1	0.01184
RCP_30_4	30	39	1	2.624	2.624	1	0.00297
RCP_30_5	30	36	2	1.480	1.480	1	0.00583
RCP_30_6	30	32	2	2.204	2.204	1	0.00647
RCP_30_7	30	18	1	1.632	1.632	1	0.01872
RCP_30_8	30	40	1	1.424	1.424	1	0.00631
RCP_30_9	30	41	2	4.940	4.940	1	0.01273
RCP_30_10	30	46	1	5.912	5.912	1	0.00598
RCP_30_11	30	34	2	0.996	0.996	1	0.00665
RCP_30_12	30	36	1	1.556	1.556	1	0.00354
RCP_30_13	30	30	1	2.540	2.540	1	0.00925
RCP_30_14	30	39	2	10.09	20.06	2	0.00965
RCP_30_15	30	30	1	4.724	4.724	1	0.01018

Table 2: Results on RCP instances: NLP-Penalty problem.

whose first phase solves the NLP-Penalty problem; the other two phases rely on the MINLP to obtain an exact global minimizer.

The solution method we are proposing applies to the case of the minimization of aircraft speed changes only (heading deviation changes remain however as optimization variables). This choice is based on the observation that, in practice, air traffic controllers (ATC) prefer handling aircraft heading deviation changes rather than (subliminal) speed changes which cannot be easily visualized on air situation displays. Thus, we consider $\lambda = 0$ in our MINLP model. This yields an optimization problem where decisions are taken on both speed and heading angle changes, and the objective is to minimize $\sum_{i \in A}(1 - q_i)^2$.

Here is a summary of our approach:

First, we solve the NLP-Penalty problem by a state-of-the-art local optimization solver in order to find a conflict-free solution. In an attempt to avoid convergence towards a point of local infeasibility, the same simple multistart heuristic described above is applied. From the obtained solution, we then compute the corresponding (angle-related) values of b_i, for all aircraft $i \in A$, and of the binary variables y_{ij}, for all pairs i, j, $i < j$ (recall that the b_i's and y_{ij}'s are not optimization variables of the NLP-Penalty problem). We also compute, *a posteriori*, the value of the speed deviation $\sum_{i \in A}(1 - q_i)^2$. A speed deviation that is zero up to a small tolerance, denoted *tol* and set to $tol = 10^{-6}$, allows us to stop the process: the computed feasible solution is globally optimal.

Second, we fix the variables y_{ij} for all i, j, $i < j$, to the values computed in the first phase, and solve the problem (referred to, in the following, as MINLP$_{fix}$) starting from the initial solution corresponding to the values of $q_i, \omega_i, \pi_i, b_i$ and y_{ij} computed in the first phase. The value of the speed deviation computed at the end of the first phase is used as an upper bound (*cutoff* value) in the branch-and-bound addressing MINLP$_{fix}$, in order to remove sub-optimal regions from the feasible set. This second phase consists therefore in a global optimization on a *continuous* nonconvex problem. If the computed solution corresponds to a null (up to the tolerance *tol*) speed deviation (which is the objective function to be minimized, since we set $\lambda = 0$), then it is a globally optimal solution, and we stop.

Third, if the speed deviation is positive (larger than *tol*), the value of the binary variables y_{ij} are unfixed, and the MINLP is solved, starting from the last computed solution, and using the corresponding speed deviation as an upper bound.

This three-phase method is described in Algorithm 1.

We apply this three-phase solution method on the 30-aircraft RCP instances. We use again `IPOPT` to solve the NLP-Penalty problem, and `COUENNE` to solve the problems MINLP$_{fix}$ and MINLP. The solvers' setting is the same as in the previous tests, with again a time limit of 600 seconds. The values of the other parameters involved in our models are also the same as in the previous tests.

Table 3 reports the results. The first four columns are the same as in previous tables, followed by two columns for each of the second and the third phases, reporting the computing time in seconds, and the corresponding speed deviations. The last column displays the total time in seconds, *i.e.*, the sum of the

Algorithm 1 Aircraft conflict avoidance: Three-phase method

Require: n, number of aircraft;
for each aircraft i: v_i, initial speed; ϕ_i initial heading; (x_i^0, y_i^0) initial position
tol, speed-deviation stopping criterion;
N, number of local-optimization starting points

- If (no initial pairwise conflicts), then **stop**
- Current iterate: $q^c := 1$, and ω^c, π^c such that $\theta = 0$; upper_bound $:= +\infty$

Phase 1:

- Solve NLP-Penalty by local optimization, trials := 1
- While (non-zero penalty value and trials $\leq N$) do
  - Solve NLP-Penalty by local optimization, trials := trials+1
  - If (zero penalty value), then update (q^c, ω^c, π^c) {*conflict-free solution*}
    else {*local infeasibility*} re-initialize $q^c := 1$, and ω^c, π^c such that $\theta = 0$
- If (trials $> N$) then compute b^c, y^c and go to Phase 3
- Compute b^c, y^c, and $q_{dev} := \sum_{i \in A}(1 - q_i^c)^2$
- If ($q_{dev} \leq tol$) then **stop**, else upper_bound := q_{dev}

Phase 2:

- Solve MINLP$_{fix}$ by continuous global optimization {*with fixed y (= y^c)*}
  using upper_bound and starting from $(q^c, \omega^c, \pi^c, b^c, y^c)$,
  with $\lambda = 0$ {*minimizing the speed deviation*}
  to get new $(q^c, \omega^c, \pi^c, b^c, y^c)$. Compute $q_{dev} := \sum_{i \in A}(1 - q_i^c)^2$
- If ($q_{dev} \leq tol$) then **stop**, else upper_bound := q_{dev}

Phase 3:

- Solve MINLP by mixed-integer global optimization, using upper_bound
  and starting from $(q^c, \omega^c, \pi^c, b^c, y^c)$,
  with $\lambda = 0$ {*minimizing the speed deviation*}

computing time for solving the problem through the three phases of computation. The computing time for the first phase is the one reported in Table 2 and is therefore not repeated in Table 3. A dash ("-") in the 3rd-phase columns indicates that global optimality was already proved when exiting the 2nd phase.

One observes that all the instances, except RCP_30_5, are solved to global optimality. At the exception of three instances, one does not need to go beyond the second phase to obtain an optimal solution. For all tests, the speed deviation computed through the second phase is at most 3.3 $\times 10^{-6}$. The third phase is run for the three instances for which the speed deviation is not sufficiently small (larger than tol). In two cases out of these three instances, the third phase then reduces the speed deviation to 10^{-17}, whereas for the remaining instance the time limit is attained. The total computing time to obtain a globally-optimal solution is lower than one minute for 8 out of the 15 instances. This is very promising in view of the aircraft conflict avoidance application.

				2nd phase: MINLP$_{fix}$		3rd phase: MINLP		Total
Name	n	n_c	n_{hth}	time (s)	speed dev.	time (s)	speed dev.	time (s)
RCP_30_1	30	35	1	34.93	1.4e-06	7.900	4.22e-17	45.94
RCP_30_2	30	38	1	59.69	6.5e-15	–	–	60.76
RCP_30_3	30	46	1	2.920	1.7e-16	–	–	3.816
RCP_30_4	30	39	1	62.98	2.6e-16	–	–	65.60
RCP_30_5	30	36	2	43.98	2.2e-06	tlim	2.24e-06	tlim
RCP_30_6	30	32	2	43.99	1.5e-16	–	–	46.20
RCP_30_7	30	18	1	105.2	1.2e-16	–	–	106.8
RCP_30_8	30	40	1	2.700	1.3e-17	–	–	4.124
RCP_30_9	30	41	2	51.23	3.3e-06	20.38	1.89e-17	76.56
RCP_30_10	30	46	1	67.59	7.3e-07	–	–	73.46
RCP_30_11	30	34	2	51.79	4.4e-16	–	–	52.79
RCP_30_12	30	36	1	86.90	4.8e-15	–	–	88.45
RCP_30_13	30	30	1	4.092	3.6e-16	–	–	6.632
RCP_30_14	30	39	2	3.752	8.8e-18	–	–	23.81
RCP_30_15	30	30	1	79.32	9.9e-16	–	–	84.05

Table 3: Results for large RCP instances: three-phase method. Global exact solutions (except for RCP_30_5)

7 Conclusion

This paper proposes two new mathematical optimization models for the air-traffic conflict avoidance problem involving finding aircraft speed *and* heading (angle) deviations. One model is a mixed-integer nonlinear optimization formulation. The other model constitutes one of the only two continuous-optimization formulations of the aircraft conflict avoidance problem; it relies on the continuous quadrant penalty formulation of logical constraints introduced in [11]. From the modeling point of view, both models benefit from reformulations of trigonometric terms and valid bound inequalities. From the air traffic control operational point of view, combining the speed and heading deviation maneuvers makes the introduced models more realistic than most models found in the literature.

The three-phase approach that we propose takes advantage of the strengths of both continuous local optimization (efficient computation of bounds), and of the combinatorial branch-and-bound method (proof of global optimality). In particular, the proposed NLP-Penalty model yields efficient computation of good quality upper bounds. Since efficient exact global optimization relies on bound computation, the NLP-Penalty turns out to have a significant impact on the overall resolution. Ultimately, the efficiency of the three-phase approach we introduce allows us to solve difficult instances of the complex problem of aircraft conflict avoidance.

Future tracks of research include traffic conflict problems dealing with autonomous vehicles involving for instance separation maneuvers applied not simultaneously, in a dynamic environment.

Acknowledgements S. Cafieri and M. Mongeau thank Barbara Conn for

proofreading the manuscript.

References

[1] H. Ahn and D. Del Vecchio. Safety verification and control for collision avoidance at road intersections. *IEEE Transactions on Automatic Control*, 63(3):630–642, 2017.

[2] A. Alonso-Ayuso, L. F. Escudero, and F. J. Martín-Campo. Collision avoidance in air traffic management: A mixed-integer linear optimization approach. *IEEE Transactions on Intelligent Transportation Systems*, 12(1):47–57, 2011.

[3] A. Alonso-Ayuso, L. F. Escudero, and F. J. Martín-Campo. A mixed 0-1 nonlinear optimization model and algorithmic approach for the collision avoidance in ATM: Velocity changes through a time horizon. *Computers & Operations Research*, 39(12):3136–3146, 2012.

[4] A. Alonso-Ayuso, L. F. Escudero, and F. J. Martín-Campo. Exact and approximate solving of the aircraft collision resolution problem via turn changes. *Transportation Science*, 50:263–274, 2016.

[5] J. Basch, J. Erickson, L. J. Guibas, J. Hershberger, and L. Zhang. Kinetic collision detection between two simple polygons. In *Proceedings of the tenth annual ACM-SIAM symposium on Discrete algorithms*, pages 102–111. SIAM, 1999.

[6] P. Belotti, P. Bonami, M. Fischetti, A. Lodi, M. Monaci, A. Nogales-Gómez, and D. Salvagnin. On handling indicator constraints in mixed integer programming. *Computational Optimization and Applications*, 65(3):545–566, 2016.

[7] P. Belotti, J. Lee, L. Liberti, F. Margot, and A. Wächter. Branching and bounds tightening techniques for non-convex MINLP. *Optimization Methods & Software*, 24(4-5):597–634, 2009.

[8] N. Bezzo, R. Fierro, A. Swingler, and S. Ferrari. A disjunctive programming approach for motion planning of mobile router network. *International Journal of Robotics & Automation*, 3:227–252, 2010.

[9] S. Cafieri. MINLP in Air Traffic Management: Aircraft conflict avoidance. In T. Terlaky, M. Anjos, and S. Ahmed, editors, *Advances and Trends in Optimization with Engineering Applications*, MOS-SIAM Series on Optimization. SIAM, 2017.

[10] S. Cafieri, L. Cellier, F. Messine, and R. Omheni. Combination of optimal control approaches for aircraft conflict avoidance via velocity regulation. *Optimal Control Applications and Methods*, 39:181–203, 2018.

[11] S. Cafieri, A. R. Conn, and M. Mongeau. The continuous quadrant penalty formulation of logical constraints. Technical Report hal-03623407, HAL open archive, 2022. https://hal.archives-ouvertes.fr/hal-03623407.

[12] S. Cafieri and C. D'Ambrosio. Feasibility pump for aircraft deconfliction with speed regulation. *Journal of Global Optimization*, 71:501–515, 2018.

[13] S. Cafieri and N. Durand. Aircraft deconfliction with speed regulation: New models from mixed-integer optimization. *Journal of Global Optimization*, 58(4):613–629, 2014.

[14] S. Cafieri and R. Omheni. Mixed-integer nonlinear programming for aircraft conflict avoidance by sequentially applying velocity and heading angle changes. *European Journal of Operational Research*, 260:283–290, 2017.

[15] S. Cafieri and D. Rey. Maximizing the number of conflict-free aircraft using mixed-integer nonlinear programming. *Computers & Operations Research*, 80:147–158, 2017.

[16] S. Cameron. Collision detection by four-dimensional intersection testing. *IEEE Transactions on Robotics and Automation*, 6(3):291–302, 1990.

[17] F. H. Dias, H. L. Hijazi, and D. Rey. Disjunctive linear separation conditions and mixed-integer formulations for aircraft conflict resolution. *European Journal of Operational Research*, 296(2):520–538, 2022.

[18] I. S. Duff. MA57—A code for the solution of sparse symmetric definite and indefinite systems. *ACM Transactions on Mathematical Software*, 30(2):118–144, 2004.

[19] J. G. Ecker, M. Kupferschmid, and S. P. Marin. Performance of several optimization methods on robot trajectory planning problems. *SIAM Journal on Scientific Computing*, 15(6):1401–1412, 1994.

[20] R. Fourer, D. M. Gay, and B. W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*. Brooks/Cole, 2nd edition, 2002.

[21] J. Kallrath and S. Rebennack. Cutting ellipses from area-minimizing rectangles. *Journal of Global Optimization*, 59:405–437, 2014.

[22] T. Lehouillier, J. Omer, F. Soumis, and G. Desaulniers. Two decomposition algorithms for solving a minimum weight maximum clique model for the air conflict resolution problem. *European Journal of Operational Research*, 256(3):696–712, 2017.

[23] J. Omer. A space-discretized mixed-integer linear model for air-conflict resolution with speed and heading maneuvers. *Computers & Operations Research*, 58:75–86, 2015.

[24] L. Pallottino, E. M. Feron, and A. Bicchi. Conflict resolution problems for air traffic management systems solved with mixed integer programming. *IEEE Transactions on Intelligent Transportation Systems*, 3(1):3–11, 2002.

[25] C. Peyronne, A. R. Conn, M. Mongeau, and D. Delahaye. Solving air traffic conflict problems via local continuous optimization. *European Journal of Operational Research*, 241(2):502–512, 2015.

[26] D. Rey and H. L. Hijazi. Complex number formulation and convex relaxations for aircraft conflict resolution. *2017 IEEE 56th Annual Conference on Decision and Control (CDC), Melbourne, Australia*, pages 88–93, 2017.

[27] D. Rey, C. Rapine, R. Fondacci, and N.-E. El Faouzi. Subliminal speed control in air traffic management: Optimization & simulation. *Transportation Science*, 50(1):240–262, 2015.

[28] A. E. Vela, S. Solak, J.-P. B. Clarke, W. E. Singhose, E. R. Barnes, and E. L. Johnson. Near real-time fuel-optimal en route conflict resolution. *IEEE Transactions on Intelligent Transportation Systems*, 11(4):826–837, Dec 2010.

[29] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.