A Flexible Framework for Solving the Air Conflict Detection and Resolution Problem using Maximum Cliques in a Graph

Thibault Lehouillier, Jérémy Omer, François Soumis, Guy Desaulniers
Group for Research in Decision Analysis
Polytechnique Montréal, QC, Canada
{thibault.lehouillier,jeremy.omer,francois.soumis,guy.desaulniers}@gerad.ca

***Abstract*—**
In this article, we present a new formulation for the air conflict detection and resolution problem. Given the current position, speed and acceleration of a set of aircraft, we identify the maneuvers required to avoid all possible conflicts and such that the fuel costs are minimized. To this end, we design a graph whose vertices correspond to discretized values of maneuvers and whose edges link conflict-free maneuvers. Finding a solution to the problem is equivalent to searching a clique of minimum weight and maximum cardinality in the graph. We formulate this search as a mixed integer linear program, since the weights of the vertices depend on the vertices in the clique. The significance of the presented algorithm relies on its flexibility, as changing hypotheses like the objective function, aircraft dynamics or the maneuvers considered do not change the method presented. Computational results highlight short solution times, where situations involving up to 20 aircraft in a complex situation are solved to optimality in less than 20 seconds.

***Index Terms*—Air Traffic Control, Conflict Resolution, Maximum Clique, Mixed Integer Linear Programming**

I. Introduction

A. Context: challenges of air traffic control

Developing advanced decision algorithms for the air traffic control (ATC) is of great importance for the overall safety and usage of the airspace, both in Europe and the United States. Such automated tools are recognized as key-components of future air traffic management (ATM) systems like the Single European Sky ATM Research (SESAR) [1] project in Europe and the Next Gen [2] program in the United States. Resolution algorithms for the air-traffic collision and detection problem are relevant especially in a context of growing traffic, where capacity and safety become an issue. Indeed, the latest long-term forecast published by EUROCONTROL states that the traffic demand will increase by 20% to 80% between 2012 and 2035 [3]. Besides, a simulation-based study performed by Lehouillier et al. [4] shows that the controllers in charge of the traffic in 2035, which will have increased by 50%, would have to solve on average 27 conflicts per hour in a busy sector.

B. Literature review

Maintaining separation between aircraft is usually referred to as the air conflict detection and resolution (CDR) problem. A conflict is a predicted loss of separation, i.e., when two aircraft are too close to each other regarding predefined horizontal and vertical separation distances. Typically, these distances define a safety cylinder of radius 5NM and height 2000ft around each aircraft, as displayed in Figure 1. A conflict happens when the predicted trajectory of an aircraft intersects the safety cylinder of another aircraft. To resolve a conflict, the controllers issue maneuvers that can consist of speed, heading or altitude changes. Given the current position, speed, acceleration and the predicted trajectory of a set of aircraft, the CDR problem corresponds to identifying the maneuvers required to avoid all conflicts and minimizing the costs induced.

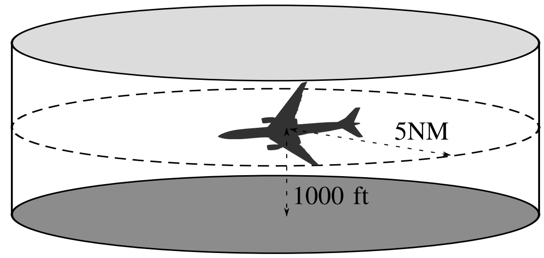


Fig. 1: Safety cylinder around an aircraft

The CDR problem is one of the most widely studied problems in air traffic management, since its time-continuous aspect and the hard separation constraints make it complex. For a comprehensive coverage of the existing literature, the reader may refer to the review performed in Martìn-Campo thesis [5]. The technique most adequate for modeling the problem is optimal control [6]. Coupled with non linear programming (NLP) techniques, it can yield interesting models, such as the one introduced by Raghunathan et al. [7]. The authors use a time discretization of the problem to derive solutions for simple instances with two aircraft. However, the sensitivity to the starting point of the resolution and the high computational time are considerable drawbacks.

In order to find a solution rapidly, several heuristics have been developed. Durand and Alliot [8] and Qi et al. [9] develop an ant colony algorithm, where maneuvers are chosen within a finite discrete set of heading changes performed at constant speed. Alonso-Ayuso et al. [10] adapt a variable neighborhood search algorithm considering only heading changes. Other fast methods include resolutions using the maneuvers extracted from a prescribed set [11], particle swarm optimization (see Gao et al. [12] for heading changes), or neural networks (see Durand et al. [13], Christodoulou and Kontegeorgous [14] for speed changes). Such methods present the asset of bringing fast solutions, but the hypotheses can be restrictive since they do not consider every type of maneuvers, and the convergence is not guaranteed.

Powerful theoretical frameworks for the study of the CDR problem are mixed integer linear and nonlinear programming. Pallottino et al. [15] exploit the geometry of the constraints of separation to develop two mixed integer linear programs (MILPs) that solve the problem using either speed changes with constant headings or heading changes with constant speeds. Christodoulou and Costoulakis [16] use the linear avoidance constraints of this article in a three dimensional nonlinear model using both speed and heading changes. Vela et al. [17] use the constraints of [15] to develop a MILP for the CDR problem by issuing a single command of speed and/or heading change to each aircraft. Alonso-Ayuso et al. [18] develop a MILP that considers both velocity and altitude changes. The authors also extend the model of [15] by introducing continuous instead of instantaneous speed changes in [19]. MILPs and MINLPs can be used to perform a time-based discretization of the optimal control, as in Schouwenaars' thesis [20] or more recently by Omer and Farges [21]. Omer [22] also develops a MILP with a space discretization using only the points of interest for the conflict resolution.

Graph theory has also been used in ATM, but mostly for the air traffic flow management (ATFM) problem (see the two articles by Bertsimas and Patterson [23], [24]). However, when it comes to ATC, graph theory is seldom used. Generally, conflicts between aircraft are modeled by a graph whose vertices represent the different aircraft and whose edges link pairs of conflicting aircraft. Vela [25] and Sherali et al. [26] use conflict graphs in their models. Resmerita et al. [27] study *a priori* conflict resolution by developing a multi-agent system where each aircraft has to choose a path in a resource graph whose vertices represent zones of the airspace and where chosen paths have to be conflict-free. Barnier and Brisset [28] assign different flight levels to aircraft with intersecting routes by looking for maximum cliques in a graph where a proper coloring of the vertices defines an assignment of all aircraft to a set of given flight levels.

Models tend to be restrictive to cope with technology limitations. More flexibility can be achieved if the scenario generation is separated from the resolution process itself. To this end Allignol et al. [29] present a new framework for conflict resolution that yields meaningful results on a detailed benchmark.

The model presented in this article uses the concept of a clique in a graph, which is a subset of the vertices where each pair of elements is linked by an edge. Finding a maximum clique in an arbitrary graph is a well-known optimization problem that is among the $\mathcal{NP}$-hard problems enumerated by Karp [30]. Due to its high complexity, the problem has been thoroughly studied and several methods, both exact and heuristic, have been developped. For a comprehensive coverage of the theoretical results, complexity study and existing methods overview, one can refer to Bomze et al. [31] and Hao et al. [32].

C. Contribution statement

In this article, we formulate the air-traffic collision detection and resolution problem as the search for a maximum clique of minimum weight in a specific graph linking conflict-free maneuvers. The first contribution of the model is its originality, both because graph theory is seldom used to model the CDR problem, and also because of the cost structure chosen for this article. Indeed, the cost of the vertices depends on the vertices belonging to a maximum clique. To our knowledge, our model is a new variant of the problem of finding a maximum clique of minimal weight in a general graph. Besides, this choice of modeling allows us to maintain a reasonable size for the graph built, hence preventing an additional computational effort. We formulate our problem as a MILP, which we test through extensive simulations on a benchmark composed of complex virtual instances. The variety of these instances challenges our model, and results highlight small solution times.

The design itself of our model makes it modular and adaptable. Indeed, changing the objective function, the types of maneuvers, or the aircraft dynamics is equivalent to modifying the characteristics of the graph built. Moreover, several sources of incertitude could also be taken into account, and this model serves us as a future basis for stochastic models. We consider this as the main contribution of our article, since flexibility is a critical point when studying the CDR problem. Indeed, the aforementioned MILPs and MINLPs are highly sensitive to the hypotheses of the problem. For instance, the constraints given in [15] are linear in almost all situations, except when heading changes are considered with aircraft flying at different speeds. In addition to being robust, our model will allow us to conduct meaningful comparisons with other existing models of the literature.

II. Problem Formulation

A. Modeling aircraft dynamics

To model the flight dynamics, we use a three-dimensional point-mass model for the aircraft, which is used in the majority of the literature on trajectory optimization. The model is given by the following equations:

$$\frac{dp_x}{dt} = V\cos\gamma\cos\chi \tag{1}$$
$$\frac{dp_y}{dt} = V\cos\gamma\sin\chi \tag{2}$$
$$\frac{dp_z}{dt} = V\sin\gamma \tag{3}$$
$$\frac{d\gamma}{dt} = \frac{g_0}{V}(n\cos\phi - \cos\gamma) \tag{4}$$
$$\frac{d\chi}{dt} = \frac{g_0}{V}\frac{n\sin\phi}{\cos\gamma} \tag{5}$$
$$\frac{dV}{dt} = \frac{F_T - F_D}{m} - g_0\sin\gamma \tag{6}$$

The position of the aircraft is given by the coordinates (p_x, p_y, p_z) of its center of gravity in a local coordinate system:

(p_x,p_y) being its coordinates in an horizontal plane and p_z its altitude. The aircraft flies at speed V and the angles χ, ϕ and γ correspond to its heading, roll and pitch respectively. F_T and F_D denote the norm of the thrust and drag forces respectively, m is the aircraft mass, n is the load factor and g_0 corresponds to the gravitational acceleration.

In this work, aircraft follow their planned 4D trajectory. This concept allows aircraft to follow an almost unrestricted trajectory in exchange of meeting time and space requirements over of a sequence of 4D points. The non-compliance with this contract costs penalty fees to companies. As a consequence, it is important to make sure that, after performing a maneuver, an aircraft recovers its initial 4D trajectory as soon as possible. We assume that the planned speed for an aircraft corresponds to its nominal speed, i.e., the speed minimizing the fuel burn rate per distance unit traveled using the model described in the BADA user manual [33] .

A maneuver performed by an aircraft is described as a triplet $(\delta V, \delta\chi, \delta FL)$ corresponding to a deviation in speed, heaeding and flight level respectively. We consider the maneuvers as non-instantaneous, and we follow the model described in [34]. The author of [34] states that the typical acceleration during a speed adjustment for commercial transport aircraft is in the order of 0.4kn/s or $0.02\ g$. This value is set to respect the comfort of passengers. Heading changes are approximated by a steady turn of constant rate and radius. The bank angle ϕ typically used is $35°$, which corresponds to the nominal bank angle for civil flights during the cruise phase [33]. The maximal turn rate $\omega_{\max}$ allowed is $\omega_{\max} = 0.89\text{rad.s}^{-1}$.

We also consider the altitude maneuvers to be dynamic. The changes of flight level are performed with a vertical speed which is a function of the thrust, the drag, and the true airspeed. Details on the computation of the vertical speed can be found in the BADA user manual [33].

B. On cliques

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected, simple graph with a vertex set $\mathcal{V}$ and an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$.

A *clique* in graph $\mathcal{G}$ is a vertex set $\mathcal{C}$ with the property that each pair of vertices in $\mathcal{C}$ is linked by an edge:

$$\mathcal{C} \subseteq \mathcal{V} \text{ is a clique} \Leftrightarrow \forall (u, v) \in \mathcal{C} \times \mathcal{C}, (u, v) \in \mathcal{E} \quad (7)$$

A *maximum* clique in $\mathcal{G}$ is a clique that is not a subset of any other clique in $\mathcal{G}$. The cardinality of a maximum clique of $\mathcal{G}$ is called *clique number* and is denoted by $w(\mathcal{G})$. Let $c : \mathcal{V} \to \mathbb{R}$ be a vertex-weight function associated with $\mathcal{G}$. A *maximum* clique of *minimum-weight* in $\mathcal{G}$ is a maximum clique $\mathcal{C}$ that minimizes $\sum_{v \in \mathcal{C}} c(v)$.

C. Graph construction

In this subsection, we introduce the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ used to model the CDR problem.

1) Defining the vertices: Let $\mathcal{F} = [\![1; n]\!]$ denote the set of the considered aircraft. We define $\mathcal{M} = \cup_{f=1}^{n} \mathcal{M}_f$ as the set of the all possible maneuvers, $\mathcal{M}_f$ being the set of maneuvers for aircraft $f \in \mathcal{F}$. We consider both horizontal and vertical maneuvers. Horizontal maneuvers can be of the following types:

- *NIL* refers to the *null* maneuver, i.e., when no maneuver is performed;
- H_θ corresponds to a heading change by an angle $\theta \in [-\pi; \pi]$[1];
- S_δ corresponds to a relative speed change of $\delta\%$. We use relative speed changes, because they have already been chosen in large scale projects such as ERASMUS [35].

We introduce altitude maneuvers corresponding to a flight level change. Formally, we denote $V_{\delta h}$ a change of δh flight levels (δh can be negative if the aircraft is descending).

The set of vertices is defined as $\mathcal{V} = [\![1; |\mathcal{M}|]\!]$[2]. We note $\mathcal{V}_f$ the set of vertices corresponding to aircraft f.
In emergency scenarios where the feasibility of the problem can be an issue, it is possible to introduce n vertices corresponding to costly emergency maneuvers. Such maneuvers have already been studied, and can for instance correspond to maneuvers implemented by the Terminal Collision Avoidance system [36], or to the maneuvers described by Schouwenaars [20]. However, since because the feasibility was not an issue for the tested instances, those vertices were not considered in this article.
The weight of the vertices correspond to the fuel consumption induced by the corresponding maneuvers. We give further detail in Subsection II-C5.

2) Defining the edges: Let $(i, j) \in \mathcal{V} \times \mathcal{V}$ be a pair of vertices representing maneuvers $(m_i, m_j) \in \mathcal{M} \times \mathcal{M}$ of aircraft $(f_i, f_j) \in \mathcal{F} \times \mathcal{F}$. For $i \neq j$, we write $m_i \square m_j$ when no conflict occurs if aircraft f_i follows maneuver m_i while aircraft f_j performs maneuver m_j. The set of edges $\mathcal{E}$ corresponds to the pairs of maneuvers performed by two different aircraft without creating conflicts:

$$\mathcal{E} = \{(i, j) \in \mathcal{V} \times \mathcal{V}, i \neq j : m_i \square m_j\} \quad (8)$$

It is important to note that there is no edge between two different maneuvers of a given aircraft, which yields Proposition 2.1.

Proposition 2.1: For all $f \in \mathcal{F}, \mathcal{V}_f$ is a stable set, i.e there is no edge linking two distinct vertices of $\mathcal{V}_f$.

Let $(i, j) \in \mathcal{V} \times \mathcal{V}$ be a pair of vertices representing maneuvers $(m_i, m_j) \in \mathcal{M} \times \mathcal{M}$ of aircraft $(f_i, f_j) \in \mathcal{F} \times \mathcal{F}$. The methodology used to compute if the edge (i, j) is added $\mathcal{G}$ is described with the following notations:

- $\mathcal{T}$: time horizon for the conflict resolution. In this article, we define $\mathcal{T}$ as a 15-minute interval, since it corresponds to the average time an aircraft takes to cross a sector.
- $\boldsymbol{p}_{f_i}(t) \in \mathbb{R}^3$: position vector of aircraft f_i at time t. $p_{f_i,x}(t)$ $p_{f_i,y}(t)$ and $p_{f_i,z}(t)$ denote respectively the abscissa, ordinate and altitude components of the position vector;

[1] positive angles correspond to counter-clockwise rotations
[2] $|\mathcal{M}|$ is the cardinality of set $\mathcal{M}$

- $\boldsymbol{s}_{f_i}(t) \in \mathbb{R}^3$: speed vector of aircraft f_i at time t. $s_{f_i,x}(t)$ $s_{f_i,y}(t)$ and $s_{f_i,z}(t)$ denote respectively the abscissa, ordinate and altitude components of the speed vector;
- $\boldsymbol{a}_{f_i}(t) \in \mathbb{R}^3$: acceleration vector of aircraft f_i at time t. $a_{f_i,x}(t)$ $a_{f_i,y}(t)$ and $a_{f_i,z}(t)$ denote respectively the abscissa, ordinate and altitude components of the acceleration vector;
- $\boldsymbol{p}_{f_j}(t), \boldsymbol{s}_{f_j}(t)$ and $\boldsymbol{a}_{f_j}(t)$ are also defined following the same notations

The definition of the maneuvers m_i and m_j applied to f_i and f_j is used to project the aircraft trajectory over time. Aircraft f_i and f_j are said to be separated at time t if and only if at least one of constraints (9) and (10) holds:

$$d^h_{f_if_j}(t)^2 = (p_{f_i,x}(t) - p_{f_j,x}(t))^2 + (p_{f_i,y}(t) - p_{f_j,y}(t))^2 \geq D^2_{\text{h,min}} \quad (9)$$

$$d^v_{f_if_j}(t)^2 = (p_{f_i,z}(t) - p_{f_j,z}(t))^2 \geq D^2_{\text{v,min}} \quad (10)$$

In this paper we choose $D_{\text{h,min}} = 5\text{NM}$ and $D_{\text{v,min}} = 1000\text{ft}$.

Consider f_i and $t_0 \in \mathcal{T}$ such that $\boldsymbol{p}_{f_i}(t_0), \boldsymbol{s}_{f_i}(t_0)$ and $\boldsymbol{a}_{f_i}(t_0)$ are known. If we assume a constant acceleration when the maneuver is applied, we obtain the position and the speed vector of f_i at time $t_0 + t$:

$$\boldsymbol{p}_{f_i}(t_0 + t) = \boldsymbol{p}_{f_i}(t_0) + (t - t_0)\boldsymbol{s}_{f_i}(t_0) + \frac{(t - t_0)^2}{2}\boldsymbol{a}_{f_i}(t_0) \quad (11)$$

$$\boldsymbol{s}_{f_i}(t_0 + t) = \boldsymbol{s}_{f_i}(t_0) + (t - t_0)\boldsymbol{a}_{f_i}(t_0) \quad (12)$$

Let $\boldsymbol{p}^h_{f_if_j}$ (respectively $\boldsymbol{s}^h_{f_if_j}$, $\boldsymbol{a}^h_{f_if_j}$) denote respectively the horizontal position, the speed and the acceleration of aircraft f_j relatively to aircraft f_i. We define

$$\begin{aligned} d^h_{f_if_j}(t+\tau) &= ||\boldsymbol{p}^h_{f_if_j}(t+\tau)|| \\ &= ||\boldsymbol{p}^h_{f_if_j}(t) + \tau \boldsymbol{s}^h_{f_if_j}(t) + \frac{\tau^2}{2}\boldsymbol{a}^h_{f_if_j}(t)|| \end{aligned}$$

where $\tau > 0$.

Let $\tau_{f_if_j} \in \underset{\tau>0}{\operatorname{argmin}}\, d^h_{f_if_j}(t+\tau)^2$ be defined by (13):

$$\frac{\partial}{\partial \tau} d^h_{f_if_j}(t + \tau_{f_if_j})^2 = 0 \quad (13)$$

Let $t^h_{f_if_j} \in \underset{t\in\mathcal{T}}{\operatorname{argmin}}\, d^h_{f_if_j}(t)^2$.

We have: $t^h_{f_if_j} = \begin{cases} 0 & \text{if } \tau_{f_if_j} \leq 0 \\ |\mathcal{T}| & \text{if } \tau_{f_if_j} \geq |\mathcal{T}| \\ \tau_{f_if_j} & \text{otherwise} \end{cases}$

Aircraft f_i and f_j are horizontally separated during interval $\mathcal{T}$ if and only if (14) holds:

$$d^h_{f_if_j}(t^h_{f_if_j})^2 \geq D^2_{\text{h,min}} \quad (14)$$

By a similar reasoning, aircraft f_i and f_j are vertically separated during interval $\mathcal{T}$ if and only if (15) holds:

$$d^v_{f_if_j}(t^v_{f_if_j})^2 \geq D^2_{\text{v,min}} \quad (15)$$

If either (14) or (15) holds when aircraft f_i and f_j apply maneuvers m_i and m_j, then an edge is created between i and j. As explained in II-A, it is important that every aircraft initiates a safe return towards its initial trajectory once the conflict is avoided. For each edge, we compute the minimum time necessary before one or both aircraft can recover their initial trajectories. The cost of the recovery of a trajectory is detailed in Subsection II-C4.

3) Application to the CDR problem: As mentioned in Section I, given the current position, speed, acceleration and the planned trajectories of a set of aircraft, solving the CDR problem consists in finding a conflict-free set of maneuvers that minimizes the costs. Proposition 2.2 links the cliques in $\mathcal{G}$ to the CDR problem:

Proposition 2.2: Let $\mathcal{C}$ be a clique in graph $\mathcal{G}$. Then $\mathcal{C}$ represents a set of conflict-free maneuvers for a subset of $\mathcal{F}$ of cardinality $|\mathcal{C}|$.

Proposition 2.2 shows that finding a set of conflict-free maneuvers for $\mathcal{F}$ is equivalent to finding a clique of $\mathcal{G}$ of cardinality $|\mathcal{F}|$. We derive the following theorem:

Theorem 2.3: If a conflict-free solution exists, then $\omega(\mathcal{G}) = |\mathcal{F}|$. Otherwise, $\omega(\mathcal{G})$ is the maximum number of flights involved in a conflict-free situation.

We define the problem $\text{CDR}_{\mathcal{M}}$ as the restriction of the CDR problem to the set of maneuvers $\mathcal{M}$. Using both Proposition 2.2 and Theorem 2.3, we can state anew the $\text{CDR}_{\mathcal{M}}$ problem as follows: solving the $\text{CDR}_{\mathcal{M}}$ problem consists in finding a clique of maximum cardinality and minimal cost in graph $\mathcal{G}$. In fact, we consider a new variant of a clique problem where the weight associated with a vertex is not known a priori and rather depends on the edges induced by the clique. Indeed, the cost associated with a maneuver depends on the duration that this maneuver will be performed before returning towards the planned trajectory. Because this duration depends on the maneuvers selected for the other aircraft, it cannot be determined a priori and must be computed as the maximum duration needed to avoid a loss of separation with all other aircraft given their chosen maneuvers. To handle such vertex costs, we will first define edge costs.

4) Computing the cost of the edges: The cost measure chosen for this article corresponds to the extra fuel consumption induced by the maneuvers, i.e., the additional fuel required to return to the 4D trajectory after the maneuver is performed. We use the model given in the BADA manual [33]. For a jet commercial aircraft f, the fuel consumption by time and distance unit is given by (16) and (17):

$$C_{t,f}(t) = c_{1,f}\left(1 + \frac{V_f(t)}{c_{2,f}}\right) F_{T,f}(t) \quad (16)$$

$$C_{d,f}(t) = \frac{C_{t,f}(t)}{V_f(t)} \quad (17)$$

where $c_{1,f}$ and $c_{2,f}$ are numerical constants depending on the type of aircraft f.

We compute the cost of an edge $e = (i, j)$ linking two vertices representing two maneuvers of aircraft f_i and f_j, denoted m_i and m_j, as a pair constituted of the extra fuel costs for both f_i and f_j, denoted $C^{(i,j)}_i$ and $C^{(i,j)}_j$.

Depending on the type of the maneuver, a different approach is followed. Given an aircraft f, we compute its nominal speed V_f. For a change of speed $V'_f = V_f(1 + \delta)$ during a period δt, we consider that a recovery period of δt at speed $V''_f = V_f(1 - \delta)$ is performed to recover the initial trajectory. After a change of direction by an angle θ during a period δt, the aircraft performs a turn with an angle θ_r in order to recover its trajectory. We denote V^r_f the speed when aircraft f is recovering its original trajectory. Typically, we choose

$V_f^r = 1.06 \times V_f$ because it represents the maximum change pilots usually do. Let t^r be the recovery time necessary for aircraft f to catch up with the initial 4D trajectory. The cost required to recover the trajectory is the extra fuel burnt when the aircraft flies at the recovery speed during t_r and the extra distance induced by the maneuver. For a flight level change, we compute the extra cost as the difference of consumption between the different flight levels, along with the cost of changing twice of flight level. The distance flown is also longer, and this extra distance is also accounted for.

5) Computing the cost of the vertices: Several techniques can be followed in order to determine the vertices cost. The basic one would be to discretize the duration of the maneuver, and to create the vertices accordingly. In this situation, computing the costs would be straight-forward. However, the drawback of this method is that the graph built is huge, which could result in a difficult resolution. We choose to follow another structure of cost because it is more compact in terms of graph size.

Let us consider a vertex i which corresponds to a maneuver m_i for an aicraft f_i. The cost of each edge linking i to one of its neighbors j, associated to a maneuver m_j for aircraft f_j, corresponds to f_i applying m_i during a time t_i^j, which depends on m_j. Time t_i^j is the minimum time during which f_i must apply m_i in order to avoid any conflict if one or both aircraft return to their initial trajectory. Following maneuver m_i for a duration t_i^j induces a cost $C_i^{(i,j)}$. If i is part of the maximum clique $\mathcal{C}$ to be determined, we need to establish the time t_i during which maneuver m_i is actually applied in order to determine its cost c_i. t_i is obtained by:

$$t_i = \max_{j \in \mathcal{V} \cap \mathcal{C}} t_i^j \tag{18}$$

As a consequence, we have that c_i is the cost of aircraft f_i applying m_i during t_i. If i is not part of the maximum clique $\mathcal{C}$, then no constraint is imposed on the cost c_i. As detailed in Section III, the optimization model will automatically force the value of c_i to 0. To conclude, we have that for any $i \in \mathcal{V}$:

$$c_i = \begin{cases} \max\limits_{j \in \mathcal{V} \cap \mathcal{C}} C_i^{(i,j)} & \text{if } i \in \mathcal{C} \\ 0 & \text{otherwise} \end{cases}$$

D. Illustrative example

For the sake of clarity, an illustrative example with three aircraft is given in Figure 2.

If each aircraft follows its planned trajectory, conflicts will happen between the blue aircraft and the two others. For this example, we assume that, in addition to the null maneuver, only two heading changes ($\pm 30°$) are allowed. We build the CDR graph shown in Figure 3. Solving the CDR is then equivalent to searching for a minimum-weight clique of 3 vertices, i.e., a triangle.

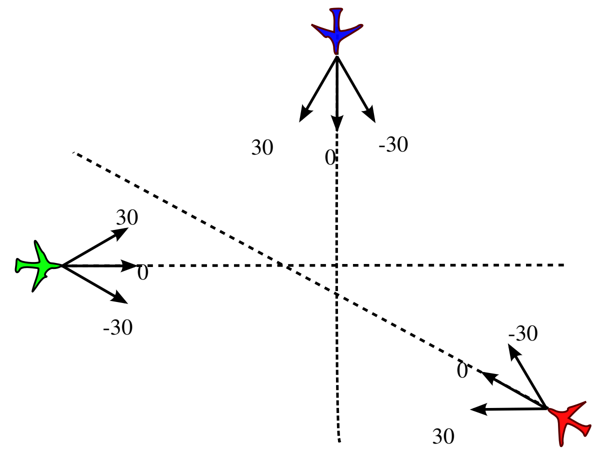


Fig. 2: Illustrative example with three aircraft

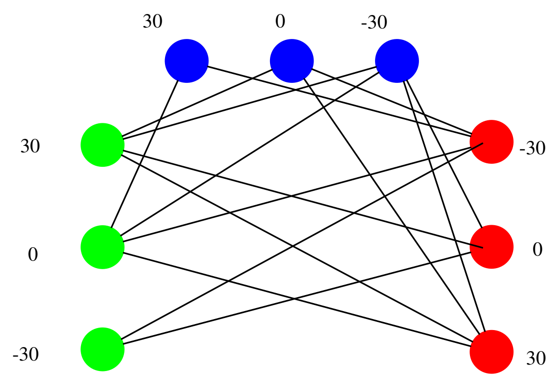


Fig. 3: Graph $\mathcal{G}$ associated with the illustrative example

III. Formulation as a Mixed Integer Linear Program

Determining the cost of a vertex i is very specific, since it is correlated to whether or not i belongs to a maximum clique $\mathcal{C}$. As a consequence, the dedicated algorithms usually in graph theory librairies cannot be used for our model. We propose to formulate the problem as a mixed-integer linear program using the following variables:

- $x_i = \begin{cases} 1 & \text{if vertex } i \text{ is part of the maximum clique} \\ 0 & \text{otherwise} \end{cases}$
- $c_i \in \mathbb{R}_+$ is the cost of vertex i.

We describe the clique search by the following integer program:

$$\text{minimize} \sum_{i \in \mathcal{V}} c_i \tag{19}$$

$$\text{subject to } x_i + x_j \leq 1, \forall (i,j) \notin \mathcal{E} \tag{20}$$

$$\sum_{i \in \mathcal{V}} x_i = |\mathcal{F}| \tag{21}$$

$$c_i \geq C_i^{(i,j)}(x_i + x_j - 1), \forall (i,j) \in \mathcal{E} \tag{22}$$

$$x_i \in \{0;1\}, \forall i \in \mathcal{V} \tag{23}$$

$$c_i \in \mathbb{R}_+, \forall i \in \mathcal{V} \tag{24}$$

The objective function (19) minimizes the cost of the maneuvers. Constraints (20) are clique constraints stating that two non-adjacents vertices must not be part of the clique. In terms of conflict resolution, it means that two maneuvers in conflict must not be part of the solution. Constraint (21) exploits Theorem 2.3 defining the cardinality of the maximum clique. Constraints (22) are used to compute the cost of the vertices: if a vertex is in the maximum clique, then its cost must be greater than the cost on every edge connecting it to other vertices in the clique. Otherwise, no particular constraint is imposed on the vertex.

IV. Results

In this section, the proposed model is validated with a benchmark of virtual instances known in the literature as complex to solve, both two-dimensional and three-dimensional. All tests were performed on a computer equipped with the following hardware: Intel Core i7-3770 processor, 3.4 GHz, 8-GB RAM. The algorithms were implemented in C++ and relies on CPLEX 12.5.1.0 [37] with default options to solve every instance.

The tables presented in this section will be of two types: dimension tables and computational results. The headings are given as follows:

- *Case*: case configuration;
- $|\mathcal{F}|$: number of aircraft;
- $|\mathcal{V}|$: number of vertices;
- $|\mathcal{E}|$: number of edges;
- $d = \frac{2|\mathcal{E}|}{|\mathcal{V}|(|\mathcal{V}|-1)}$: graph density;
- n: number of variables;
- m: number of constraints;
- z_{ip}: value of the objective function for the optimal solution of the problem (in kilograms of fuel);
- n_{nodes}: number of branch-and-bound nodes;
- t_{lp}: time (in seconds) to solve the continuous relaxation of the MILP;
- t_{ip}: time (in seconds) to obtain the z_{ip} value.

A. Two-dimensional benchmark

To test the model, a benchmark gathering virtual instances is defined. The first set of instances is roundabout instances, noted $\mathcal{R}_n$, where n aircraft are distributed on the circumference of a 100NM radius and fly towards the center at the same speed and altitude. The second set is crossing flow instances, denoted $\mathcal{F}_{n,\theta,d}$, where two trails of n aircraft separated by d nautical miles intersect each other with an angle θ. The last type of instance is a grid, denoted $\mathcal{G}_{n,d}$ constituted of two crossing flow instances $\mathcal{F}_{n,\frac{\pi}{2},d}$ intersecting with a 90° angle, one instance being translated 15NM North-East from the other. An example of these instances is given on Figure 4. Subfigure 4(a) denotes the instance $\mathcal{R}_8$ with a roundabout of 8 aircraft, Subfigure 4(b) depicts the instance $\mathcal{F}_{3,\frac{\pi}{4},10}$ and Subfigure 4(c) describes the instance $\mathcal{G}_{3,10}$. In all the instances of this benchmark, the aircraft are Airbus A320, originally flying at flight level FL330 (33000 feet). Horizontal speed maneuvers correspond to relative changes of $\pm3\%$ and $\pm6\%$. Horizontal heading changes authorized are $\pm5°, \pm10°, \pm15°$.The graph remains small when one considers this set of maneuvers, and their small magnitude makes them less costly. Nevertheless, if these values were to be inefficient to solve all the conflicts, we could introduce maneuvers of larger magnitude.

Solutions for the instances described in Figure 4 are displayed on Figure 5. Subfigure 5(a) depicts the optimal solution for the instance $\mathcal{R}_8$ where all aircraft perform a right turn of 5° and avoid each other in a roundabout fashion before returning to their initial trajectory. Instance $\mathcal{F}_{3,\frac{\pi}{4},10}$ is solved in a symmetric fashion: each trail of aircraft perform the same set of heading changes. The first aircraft of each trail turns right by 5°, while the second and third aircraft turn left by 5° and 10° respectively. Instance $\mathcal{G}_{3,10}$ is also solved symmetrically, where the horizontal trails follow the same set of maneuvers, as well as the vertical trails.

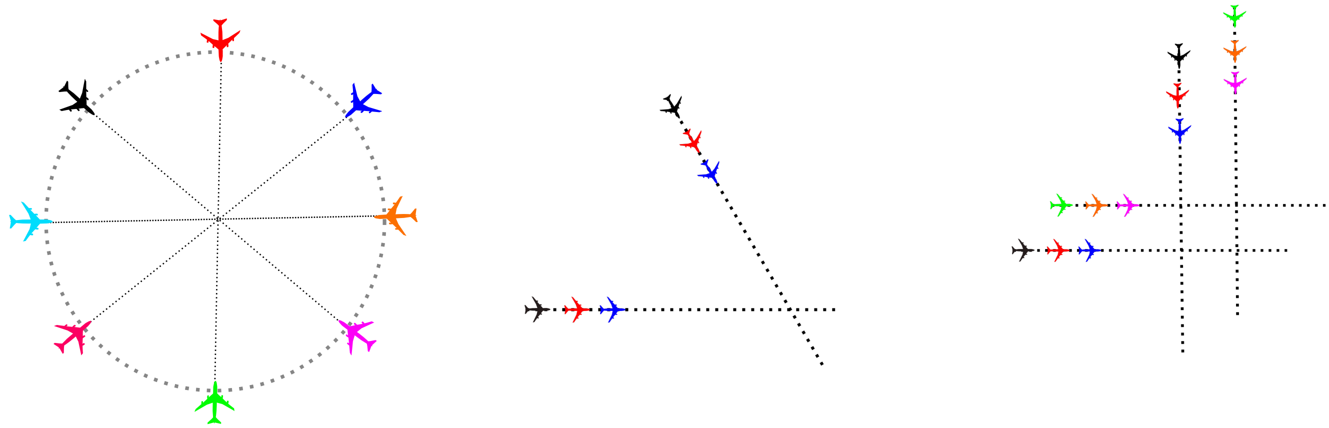
Next, some computational results for the MILP model are reported. Table I shows the dimensions of the model, whereas Table II gives the most important results. In Table II, the solution time for the continuous relaxation is very small, but the quality of the relaxation is medriocre. The optimal value is always 0, inducing a gap of 100%. This results comes from the fact that the fractional solution of the linear relaxation chooses two maneuvers for each aircraft with a value of 0.5. Constraints (22) force the cost of each vertex to be 0, hence the result. Results also display short solution times: problems known to be complex with 20 aircraft are solved to optimality in less than 15 seconds. This result is very satisfying since the density of the graph is high.

TABLE I: Dimensions table for the two-dimensional benchmark

Case	$\lvert\mathcal{F}\rvert$	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{E}\rvert$	d	m	n
$\mathcal{R}_2$	2	26	116	0.36	52	285
$\mathcal{R}_4$	4	52	832	0.63	104	1769
$\mathcal{R}_6$	6	78	2076	0.69	156	4309
$\mathcal{R}_8$	8	104	3840	0.72	208	7889
$\mathcal{R}_{10}$	10	130	6080	0.73	260	12421
$\mathcal{R}_{12}$	12	156	9096	0.75	312	18505
$\mathcal{R}_{14}$	14	182	12208	0.74	364	24781
$\mathcal{R}_{16}$	16	208	16416	0.76	416	33249
$\mathcal{R}_{18}$	18	234	20772	0.76	468	42013
$\mathcal{R}_{20}$	20	260	25760	0.77	520	52041
$\mathcal{F}_{5,30,10}$	10	110	4522	0.75	220	9265
$\mathcal{F}_{5,45,10}$	10	110	4518	0.75	220	9257
$\mathcal{F}_{5,60,10}$	10	110	4478	0.75	220	9177
$\mathcal{F}_{5,75,10}$	10	110	4492	0.75	220	9205
$\mathcal{F}_{5,90,10}$	10	110	4528	0.76	220	9277
$\mathcal{G}_{2,3,10}$	12	132	6645	0.78	264	13555
$\mathcal{G}_{2,5,10}$	20	220	19724	0.82	440	39889

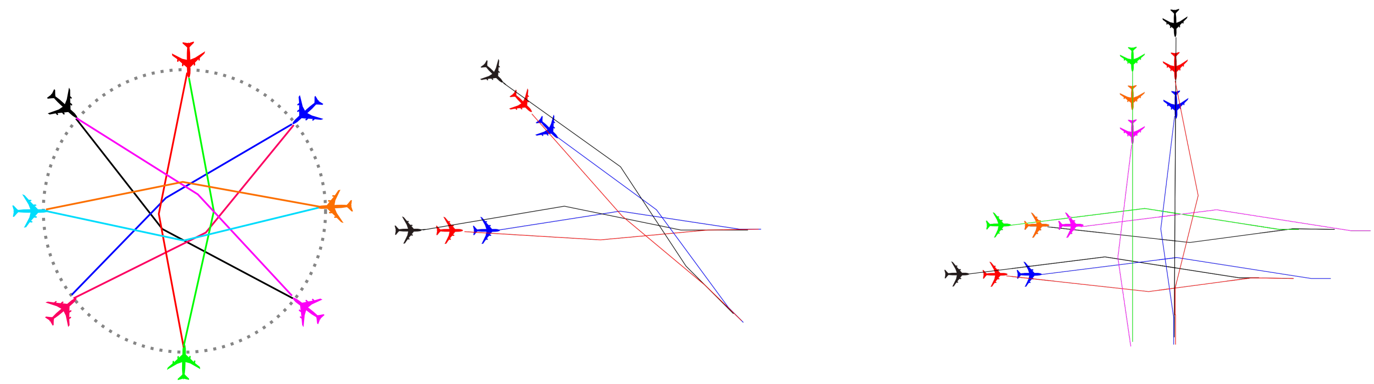
TABLE II: Computational results for the two-dimensional benchmark

Case	z_{ip}	n_{nodes}	t_{lp}	t_{ip}
$\mathcal{R}_2$	3.71	6	0	0.05
$\mathcal{R}_4$	14.98	73	0	0.07
$\mathcal{R}_6$	22.7	0	0.01	0.19
$\mathcal{R}_8$	31.05	47	0.01	0.83
$\mathcal{R}_{10}$	112.7	208	0.05	1.4
$\mathcal{R}_{12}$	189.27	581	0.09	3.11
$\mathcal{R}_{14}$	224.75	183	0.1	6.46
$\mathcal{R}_{16}$	261.44	162	0.15	9.08
$\mathcal{R}_{18}$	636.7	257	0.21	12.1
$\mathcal{R}_{20}$	740.6	210	0.27	6.5
$\mathcal{F}_{5,30,10}$	49.08	405	0.02	1.5
$\mathcal{F}_{5,45,10}$	41.29	535	0.02	1.52
$\mathcal{F}_{5,60,10}$	34.49	238	0.02	1.39
$\mathcal{F}_{5,75,10}$	30.66	496	0.02	1.34
$\mathcal{F}_{5,90,10}$	28.28	269	0.02	1.41
$\mathcal{G}_{2,3,10}$	57.65	564	0.01	3.64
$\mathcal{G}_{2,5,10}$	121.92	2740	0.2	12.7



(a) Roundabout (b) Crossing Flow (c) Grid

Fig. 4: Examples of instances



(a) Roundabout (b) Crossing Flow (c) Grid

Fig. 5: Solutions of the examples

B. Three-dimensional benchmark

In this final benchmark, we introduce altitude maneuvers: aircraft are allowed to move to an adjacent flight level. We test our model on the same instances as for the two-dimensional case. We report the computational results in Table III.

The values of the optimal solutions for the roundabout instances remain the same, highlighting that it is optimal to make simple turns instead of changing flight levels. For the crossing flows and the grid instances, it is more efficient for some aircraft to change their flight level instead of turning or changing their speed. As a consequence, the solutions are less expensive.

TABLE III: Computational results for the three-dimensional benchmark

Case	z_{ip}	n_{nodes}	t_{lp}	t_{ip}
$\mathcal{R}_2$	3.71	6	0	0.05
$\mathcal{R}_4$	14.98	153	0	0.41
$\mathcal{R}_6$	22.7	440	0.01	0.25
$\mathcal{R}_8$	31.05	245	0.01	1.12
$\mathcal{R}_{10}$	112.7	375	0.05	1.41
$\mathcal{R}_{12}$	189.27	648	0.09	3.42
$\mathcal{R}_{14}$	224.75	256	0.1	6.88
$\mathcal{R}_{16}$	261.44	210	0.15	11.45
$\mathcal{R}_{18}$	636.7	289	0.21	14.12
$\mathcal{R}_{20}$	740.6	223	0.27	8.12
$\mathcal{F}_{5,30,10}$	46.12	401	0.02	1.58
$\mathcal{F}_{5,45,10}$	40.12	588	0.02	2.13
$\mathcal{F}_{5,60,10}$	31.69	324	0.02	1.96
$\mathcal{F}_{5,75,10}$	30.11	542	0.02	1.78
$\mathcal{F}_{5,90,10}$	26.12	287	0.02	1.45
$\mathcal{G}_{2,3,10}$	45.18	612	0.01	4.12
$\mathcal{G}_{2,5,10}$	108.12	2910	0.2	16.7

V. Conclusions

In this article we studied the air-traffic conflict detection and resolution problem. We designed a graph whose vertices correspond to discretized values of maneuvers and whose edges link conflict-free maneuvers. A solution to the problem corresponds to a maximum clique of minimum cost in the graph. To build the graph we used a three dimensional point-mass model for the dynamics of the aircraft, and considered maneuvers to be performed with constant acceleration. Possible maneuvers are speed changes, heading changes and altitude changes. The objective function reflects the extra fuel consumption induced by the performed maneuvers. The cost structure used in the model is specific, since the cost of the vertices depends on the vertices belonging to the maximum clique. This specifity makes our model an original variant of the search for a maximum clique of minimum weight. We developed a mixed-integer linear programming formulation to handle the clique search. We performed tests for our model on virtual instances known to be complex, with and without altitude maneuvers. Results exhibit small solution times (less than 15 seconds). These results are promising for the pursuit of further research, since the instances considered are more complex than real-life instances.

The main advantage of our model is its flexibility. Indeed, every similar problem using discretized values for the maneuvers could be studied with the same mathematical framework. For instance, different maneuvers could be considered, inducing a change in the vertices. Other objective functions could be considered, depending on the focus of interest. One could try to minimize the magnitude of the maneuvers performed, or use the cost-index of companies to maximize the utility of the aircraft. Another model for the aircraft dynamics could also be used for the problem. All these changes can be handled by our model, which is interesting for the community since in the future we will be able to compare this model to other existing models.

Along with this comparison, another point of investigation will be to introduce uncertainties in our model. These uncertainties could be of different types: we can consider errors in the trajectory prediction, or introduce wind to have a more realistic footing for our study. Real life instances would also be valuable to validate the performance of our model.

Acknowledgments

The research conducted for this paper is part of the project OPR-601 funded by the CRIAQ. The authors are indebted to the different stakeholders involved in the project.

References

[1] SESAR Joint Undertaking, “European ATM master plan, edition 2,” Tech. Rep., 2012.

[2] Joint Planning and Development Office, “Next gen air transportation system integrated work plan, technical report,” Tech. Rep., 2008.

[3] EUROCONTROL, “Eurocontrol long-term forecast: IFR flight movements 2013-2035,” Eurocontrol - STATFOR, Tech. Rep., 2013.

[4] T. Lehouillier, J. Omer, F. Soumis, and C. Allignol, “Interactions between operations and planning in air traffic control,” Proceedings of the 2nd International Conference of Research in Air Transportation, Istanbul, 2014.

[5] M. Campo and F. Javier, “The collision avoidance problem: methods and algorithms,” Ph.D. dissertation, 2010.

[6] K. Zhou, J. C. Doyle, K. Glover et al., Robust and optimal control. Prentice Hall New Jersey, 1996, vol. 40.

[7] A. U. Raghunathan, V. Gopal, D. Subramanian, L. T. Biegler, and T. Samad, “Dynamic optimization strategies for three-dimensional conflict resolution of multiple aircraft,” Journal of guidance, control, and dynamics, vol. 27, no. 4, pp. 586–594, 2004.

[8] N. Durand, J.-M. Alliot, and J. Noailles, “Automatic aircraft conflict resolution using genetic algorithms,” Proceedings of the Symposium Applied Computing, Philadelphia, 1996.

[9] M. Qi, Fei, “Flight conflict resolution for civil aviation based on ant colony optimization,” in Computational Intelligence and Design (ISCID), 2012 Fifth International Symposium on, vol. 1. IEEE, 2012, pp. 239–241.

[10] A. Alonso-Ayuso, L. F. Escudero, F. J. Martín-Campo, and N. Mladenović, “A vns metaheuristic for solving the aircraft conflict detection and resolution problem by performing turn changes,” Journal of Global Optimization, pp. 1–14, 2014.

[11] R. Vivona, D. Karr, and D. Roscoe, “Pattern-based genetic algorithm for airborne conflict resolution,” in AIAA Guidance, Navigation and Control Conference and Exhibit, Keystone, Colorado, 2006.

[12] Y. Gao, X. Zhang, and X. Guan, “Cooperative multi-aircraft conflict resolution based on co-evolution,” in Instrumentation & Measurement, Sensor Network and Automation (IMSNA), 2012 International Symposium on, vol. 1. IEEE, 2012, pp. 310–313.

[13] N. Durand, J.-M. Alliot, and F. Médioni, “Neural nets trained by genetic algorithms for collision avoidance,” Applied Intelligence, vol. 13, no. 3, pp. 205–213, 2000.

[14] M. A. Christodoulou and S. G. Kodaxakis, “Automatic commercial aircraft-collision avoidance in free flight: the three-dimensional problem,” Intelligent Transportation Systems, IEEE Transactions on, vol. 7, no. 2, pp. 242–249, 2006.

[15] L. Pallottino, E. M. Feron, and A. Bicchi, “Conflict resolution problems for air traffic management systems solved with mixed integer programming,” Intelligent Transportation Systems, IEEE Transactions on, vol. 3, no. 1, pp. 3–11, 2002.

[16] M. Christodoulou and C. Costoulakis, “Nonlinear mixed integer programming for aircraft collision avoidance in free flight,” in Electrotechnical Conference, 2004. MELECON 2004. Proceedings of the 12th IEEE Mediterranean, vol. 1. IEEE, 2004, pp. 327–330.

[17] A. Vela, S. Solak, J.-P. 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, vol. 12, no. 1, pp. 47–57, 2011.

[18] A. Alonso-Ayuso, L. F. Escudero, F. J. Martin-Campo, and F. Javier, “Collision avoidance in air traffic management: a mixed-integer linear optimization approach,” IEEE Transactions on Intelligent Transportation Systems, vol. 12, no. 1, pp. 47–57, 2011.

[19] A. Alonso-Ayuso, L. F. Escudero, and F. J. Martin-Campo, “A mixed 01 nonlinear optimization model and algorithmic approach for the collision avoidance in ATM: Velocity changes through a time horizon,” IEEE Transactions on Intelligent Transportation Systems, vol. 39, pp. 3136–3146, 2012.

[20] T. Schouwenaars, “Safe trajectory planning of autonomous vehicles,” Ph.D. dissertation, Massachusetts Institute of Technology, 2006.

[21] J. Omer and J.-L. Farges, “Hybridization of nonlinear and mixed-integer linear programming for aircraft separation with trajectory recovery,” IEEE Transactions on Intelligent Transportation Systems, vol. 14, no. 3, pp. 1218–1230, 2013.

[22] J. Omer, “A space-discretized mixed-integer linear model for air-conflict resolution with speed and heading maneuvers,” 2014.

[23] D. Bertsimas and S. S. Patterson, “The air traffic flow management problem with enroute capacities,” Operations research, vol. 46, no. 3, pp. 406–422, 1998.

[24] ——, “The traffic flow management rerouting problem in air traffic control: A dynamic network flow approach,” Transportation Science, vol. 34, no. 3, pp. 239–255, 2000.

[25] A. E. Vela, “Understanding conflict-resolution taskload: implementing advisory conflict-detection and resolution algorithms in an airspace,” 2011.

[26] H. D. Sherali, J. Cole Smith, and A. A. Trani, “An airspace planning model for selecting flight-plans under workload, safety, and equity considerations,” Transportation Science, vol. 36, no. 4, pp. 378–397, 2002.

[27] S. Resmerita, M. Heymann, and G. Meyer, “A framework for conflict resolution in air traffic management,” in Decision and Control, 2003. Proceedings. 42nd IEEE Conference on, vol. 2. IEEE, 2003, pp. 2035–2040.

[28] N. Barnier and P. Brisset, “Graph coloring for air traffic flow management,” Annals of operations research, vol. 130, no. 1-4, pp. 163–178, 2004.

[29] C. Allignol, N. Barnier, N. Durand, and J.-M. Alliot, “A new framework for solving en-routes conflicts,” in ATM 2013, 10th USA/Europe Air Traffic Management Research and Development Seminar, 2013, pp. pp–1.

[30] R. M. Karp, Reducibility among combinatorial problems. Springer, 1972.

[31] I. M. Bomze, M. Budinich, P. M. Pardalos, and M. Pelillo, "The maximum clique problem," in Handbook of combinatorial optimization. Springer, 1999, pp. 1–74.

[32] Q. Wu and J.-K. Hao, "A review on algorithms for maximum clique problems," European Journal of Operational Research, 2014.

[33] "User manual for the Base of Aircraft Data (BADA)," Eurocontrol, Tech. Rep. 11/03/08-08, 2011.

[34] R. A. Paielli, "Modeling maneuver dynamics in air traffic conflict resolution," Journal of guidance, control, and dynamics, vol. 26, no. 3, pp. 407–415, 2003.

[35] ERASMUS, "Erasmus final report v1.1. technical report d4.6," Tech. Rep., 2009.

[36] F. A. Administration, "Introduction to tcas ii - version 7.1," Federal Aviation Administration, Tech. Rep., 2011.

[37] "CPLEX v12.5. User's manual for CPLEX," IBM ILOG, Tech. Rep. 11/03/08-08, 2014.

About the Authors

Thibault Lehouillier is a Ph.D. candidate at Polytechnique Montreal, Canada, and a research assistant at the Group on Research in Decision Analysis. He received his M.Sc. in computer science and applied mathematics from Ensimag, France, in 2011, and his M.Sc. in operations research from Joseph Fourier University, France, in 2011. His research insterests include mathematical programming applied to air-traffic management and logistics.

Jérémy Omer is a postdoctoral fellow at Polytechnique Montreal, Canada. He received a B.Sc. degree in electrical engineering from Supelec, France, and an M.Sc. degree in applied mathematics from Polytechnique Montreal. He received a Ph.D. in 2013 for research in the optimization of air traffic conducted at Onera, the French Aerospace Lab. His research interests include decomposition algorithms for linear and stochastic programming and optimization techniques applied to air-traffic systems, logistics and robotics.

François Soumis is a professor in the department of Mathematics and Industrial Engineering at Polytechnique Montreal, Canada, and the chairholder of the Canada Research Chair in Large Transportation Network Optimization. He received his B.Sc., M.Sc. and Ph.D. in mathematics from University of Montreal. His research interests are in the areas of large-scale optimization (in particular, column generation), integer programming, combinatorial optimization, and constrained shortest path problems with applications to vehicle routing and crew scheduling in land, air, rail, and maritime transportation, as well as to personnel scheduling.

Guy Desaulniers received his B.Sc., M.Sc. and Ph.D. in mathematics from University of Montreal, Canada, in 1987 and 1989, and from Polytechnique Montreal, Canada, in 1993, respectively. He is a professor in the department of Mathematics and Industrial Engineering at Ecole Polytechnique. His main research interests are in the areas of large-scale optimization (in particular, column generation), integer programming, combinatorial optimization, and constrained shortest path problems with applications to vehicle routing and crew scheduling in land, air, rail, and maritime transportation, as well as to personnel scheduling.