Flying safely by bilevel programming

Martina Cerulli, Claudia D'Ambrosio, Leo Liberti

Abstract Preventing aircraft from getting too close to each other is an essential element of safety of the air transportation industry, which becomes ever more important as the air traffic increases. The problem consists in enforcing a minimum distance threshold between flying aircraft, which naturally results in a bilevel formulation with a lower-level subproblem for each pair of aircraft. We propose two single-level reformulations, present a cut generation algorithm which directly solves the bilevel formulation and discuss comparative computational results.

1 Introduction

In Air Traffic Management, the act of avoiding that two aircraft might collide is called *aircraft deconfliction*. More in general, it describes the set of methodologies for detecting and resolving aircraft conflicts. Aircraft are said to be potentially *in conflict* if their relative distance is smaller than a given safety threshold. Despite the importance of this kind of control, it is still widely performed manually on the ground by air traffic controllers, who essentially monitor the air traffic in a given, limited space on a radar screen, giving instructions to the pilots. With the increase of aircraft automation, there comes a need for integrating human ground control by algorithmic means.

The two crucial parameters of an aircraft flight that come into play in aircraft deconfliction are the trajectory and the speed. Typically, air traffic controllers change

Martina Cerulli
CNRS - LIX, École Polytechnique, Palaiseau, e-mail: mcerulli@lix.polytechnique.fr

Claudia D'Ambrosio
CNRS - LIX, École Polytechnique, Palaiseau, e-mail: dambrosio@lix.polytechnique.fr

Leo Liberti
CNRS - LIX, École Polytechnique, Palaiseau, e-mail: liberti@lix.polytechnique.fr

the trajectory, or use heading angle changes (HAC) in order to solve potential conflicts. In this paper, we focus instead on changing the aircraft speeds (which can be performed subliminally), while keeping the trajectories unchanged: we present a Mathematical Programming (MP) formulation for aircraft separation based on speed regulation. For a wider introduction to this problem, see [1].

We remark that altitude changes are not usually considered an acceptable way to solve conflicts since they consume more fuel and feel uncomfortable to passengers. We will therefore assume that all aircraft fly within a fixed altitude layer. This will allow us to model travelling aircraft as moving points in $\mathbb{R}^2$ (see 1 as an example).

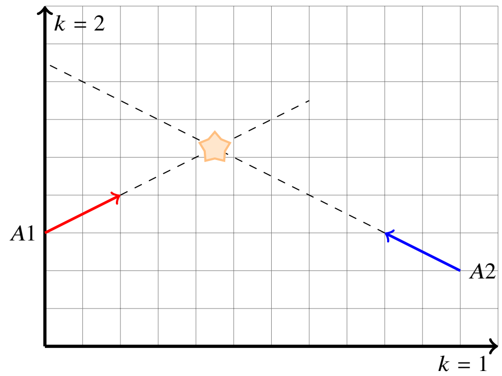

Fig. 1: Two conflicting aircraft

There is a large number and variety of approaches to the conflict detection and resolution problem. In this paper we compare our results to those obtained in [1] and [2]. These works propose Mixed-Integer Nonlinear Programming formulations for the deconfliction problem. Specifically, [1] also proposes a heuristic algorithm based on decomposing the problem into subproblems each of which only involves a small number of aircraft. The partial solutions are then combined to form a globally feasible but possibly sub-optimal solution of the original problem. A Feasibility Pump heuristic is proposed in [2]: this algorithm alternately solves two relaxed subproblems at each iteration, while minimizing the distance between the relaxed solutions.

Another approach based on aircraft HAC is proposed in [3]. First a MINLP formulation of the problem of minimizing heading angle changes satisfying the separation condition is presented. Then another mixed 0-1 nonlinear program is proposed: the number of aircraft conflicts that can be solved by speed regulation is maximized. These two MINLPs are solved using existing global solvers and then using a two-step methodology where the solution of the second MINLP is used as a pre-processing step for the first one.

Several papers consider conflicts involving more than two aircraft. In [4], for example, the planar multiple conflicts resolution problem is formulated as a nonconvex quadratically constrained quadratic program, where the objective function is chosen so as to minimize the speed deviations from the desired speed. The problem is then approximated by a convex semidefinite program, the optimal solution of which is used to randomly generate feasible and locally optimal conflict resolution maneuvers.

An equity-oriented conflict-resolution (ECR) model is introduced in [5]. The ECR model combines three optimization stages, which attempt to: resolve a maximum number of potential conflicts; promote fair conflict-resolution maneuvers (airlines are equally affected by the trajectory adjustments); reduce the delay induced by the trajectory changes. The goal is to identify a set of conflicts that can be resolved altogether, reduce the deviation from total equity and eventually minimize the total delay in the system.

Another approach is presented in [6]. This approach uses the geometric characteristics of aircraft trajectories to obtain closed-form analytical solutions for optimal combinations of heading and speed changes for horizontal-plan conflict resolution, minimizing the magnitude of the velocity vector change. This closed forms can be used also to compute the solution for speed change alone and for heading change alone.

The rest of this paper is organized as follows. In Sect. 2 we introduce the parameters and decision variables of our formulations, the bilevel formulation, and two single-level reformulations. In Sect. 3 we present our cut generation algorithm for solving the bilevel problem. In Sect. 4 we discuss some computational results.

2 Mathematical Formulations

An "optimal deconfliction" must involve a minimal deviation from the original aircraft flight plan, subject to the distance between aircraft to exceed a given safety threshold. The objective function of our formulations will therefore aim at minimizing the sum of the speed change of each aircraft. Requiring that each aircraft pair respects the safety distance at each time instant t of a given interval $[0, T]$ involves the satisfaction of an uncountably infinite set of constraints.

We propose a MP formulation of the speed-change problem variant.

1. **Sets:**
   - $A = \{1, .., n\}$ is the set of aircraft (n aircraft move in the shared airspace)
   - $K = \{1, 2\}$ is the set of directions (the aircraft move in a Euclidean plane)
2. **Parameters:**
   - T is the length of the time horizon taken into account [hours]

- d is the minimum required safety distance between a pair of aircraft [Nautical Miles NM]
- x^0_{ik} is the k-th component of the initial position of aircraft i
- v_i is the initial speed of aircraft i [NM/h]
- u_{ik} is the k-th component of the direction of aircraft i
- q_i^{min} and q_i^{max} are the bounds on the potential speed modification for each aircraft

3. **Variables:**

- q_i is the possible increase or decrease of the original speed of aircraft i: $q_i = 1$ if the speed is unchanged, $q_i > 1$ if it is increased, $q_i < 1$ if it is decreased
- t_{ij} is the instant of time defined for the aircraft pair i and j for which the distance between the two aircraft is minimized

The terminology and symbols are taken from [1], where the problem is formulated by a MINLP since there are variables both continuous and integer and nonlinear constraints arise from the separation condition modeling.

2.1 Bilevel formulation of the problem

In order to address the issue of uncountably many constraints for each value in $[0, T]$, we propose to formulate the problem as a bilevel MP (for more details on bilevel programming, see [7]) with multiple lower-level problems. Each of these subproblems ensures that the minimum distance between each aircraft pair exceeds the safety distance threshold. Thus, each lower-level subproblem involves the lower-level variable t_{ij} and is parameterized by the upper-level variables q:

$$\min_{q,t} \quad \sum_{i\in A}(q_i - 1)^2 \tag{1}$$

$$\forall i \in A \quad q_i^{\mathsf{min}} \le q_i \le q_i^{\mathsf{max}} \tag{2}$$

$$\forall i < j \in A \quad d^2 \le \min_{t_{ij}\in[0,T]} \sum_{k\in\{1,2\}} ((x^0_{ik} - x^0_{jk}) + t_{ij}(q_i v_i u_{ik} - q_j v_j u_{jk}))^2. \tag{3}$$

The upper-level (convex) objective function is the sum of squared aircraft speed changes. This corresponds to finding the feasible solution with the minimum speed change, as mentioned before. It must be minimized w.r.t the variables t and q, with each q_i within the given range $[q_i^{\mathsf{min}}, q_i^{\mathsf{max}}]$.
The objective of each lower-level subproblem is to minimize over $t_{ij} \in [0, T]$ the relative Euclidean distance between the two aircraft it describes; note that this is also a convex function. This minimum distance, reached at t^*_{ij}, must be at least d^2. This corresponds to imposing the minimum safety distance d between aircraft i and j within $[0, T]$.

2.2 KKT reformulation

We follow standard practice and replace each convex lower-level subproblem by its Karush-Kuhn-Tucker (KKT) conditions. Assuming some regularity condition (e.g. the Slater's condition) holds, this yields a single-level MP with complementarity constraints. Given the KKT multipliers μ_{ij} and λ_{ij} defined for each lower-level problem, we have:

$$\min_{q,t,\mu,\lambda} \quad \sum_{i\in A}(q_i-1)^2 \tag{4}$$

$$\text{s.t.} \quad \forall i \in A \quad q_i^{min} \leq q_i \leq q_i^{max} \tag{5}$$

$$\forall i<j \in A \quad \sum_{k\in\{1,2\}} (2t_{ij}(q_i v_i u_{ik} - q_j v_j u_{jk})^2 + \\ + 2(x_{ik}^0 - x_{jk}^0)(q_i v_i u_{ik} - q_j v_j u_{jk}) - \mu_{ij} + \lambda_{ij}) = 0 \tag{6}$$

$$\forall i<j \in A \quad \mu_{ij},\ \lambda_{ij} \geq 0 \tag{7}$$

$$\forall i<j \in A \quad \mu_{ij}\, t_{ij} = 0 \tag{8}$$

$$\forall i<j \in A \quad \lambda_{ij}\, t_{ij} - \lambda_{ij}\, T = 0 \tag{9}$$

$$\forall i<j \in A \quad -t_{ij} \leq 0,\ t_{ij} \leq T \tag{10}$$

$$\forall i<j \in A \quad \sum_{k\in\{1,2\}} ((x_{ik}^0 - x_{jk}^0) + t_{ij}(q_i v_i u_{ik} - q_j v_j u_{jk}))^2 \geq d^2 \tag{11}$$

Constraints (6)–(10) correspond to stationarity, primal and dual feasibility conditions and complementary slackness. The last constraint Eq. (11) is necessary to ensure that each KKT solution t_{ij}^* respects the safety distance.

2.3 Dual reformulation

We propose another closely related reformulation of the bilevel problem (1)-(3), which arises because the lower-level subproblems are convex Quadratic Programs (QP). Specifically, their duals are also QPs which only involve dual variables [8, 9]. In particular, an upper-level constraint such as Eq. (3) has the form

$$\text{const} \leq \min\{\frac{1}{2}\, x^\top Q x + p^\top x \mid Ax \geq b \wedge x \geq 0\}$$

with Q positive semidefinite. By strong duality it can be written as follows:

$$\text{const} \leq \max\{-\frac{1}{2}\, y^\top Q y + b^\top z \mid A^\top z - Qy \leq p \wedge z \geq 0\}, \tag{12}$$

where the maximization QP on right hand side is the dual of the previous minimization one [9].

Proposition 1 *Eq.* (12) *can be replaced by*

$$\mathsf{const} \le -\frac{1}{2}\, y^\top Q y + b^\top z \wedge A^\top z - Qy \le p \wedge z \ge 0 \; (*)$$

in Eq. (1)-(3).

Proof If Eq. (12) is active, then the maximum objective function value of the QP is const. Because of the max operator, the objective function of the QP cannot attain any larger value. This means that $(*)$ can only be feasible when $-\frac{1}{2}\, y^\top Q y + b^\top z$ attains its maximum over $A^\top z - Qy \le p$ and $z \ge 0$. If Eq. (12) is inactive, it has no effect on the optimum. Since $(*)$ is a relaxation of Eq. (12), the same holds. □

Given the dual variables y and z of the lower-level subproblems, prop. 1 yields the following reformulation of 1-(3).

$$\min_{q,y,z} \quad \sum_{i \in A} (q_i - 1)^2 \tag{13}$$

$$\forall i \in A \quad q_i^{min} \le q_i \le q_i^{max} \tag{14}$$

$$\forall i < j \in A \quad -\sum_{k=1}^{2} (q_i v_i u_{ik} - q_j v_j u_{jk})^2\, y_{ij}^2 + (-T) z_{ij} \ge d^2 - \sum_{k=1}^{2} (x_{ik}^0 - x_{jk}^0)^2 \tag{15}$$

$$\forall i < j \in A \quad -\frac{z_{ij}}{2} - \sum_{k=1}^{2} (q_i v_i u_{ik} - q_j v_j u_{jk})^2\, y_{ij} \le \sum_{k=1}^{2} (x_{ik}^0 - x_{jk}^0)(q_i v_i u_{ik} - q_j v_j u_{jk}) \tag{16}$$

$$\forall i < j \in A \quad z_{ij} \ge 0 \tag{17}$$

3 Cut generation algorithm

We introduce a solution algorithm for the bilevel formulation (1)-(3), using a cutting plane approach. We iteratively define the feasible set of the upper-level problem by means of quadratic inequalities in the upper-level variables q_i, q_j.
The algorithm is as follows:

CP algorithm

1. Let $h = 1$; initialize the relaxation R_h of the bilevel program, obtained considering only the upper-level problem;

2. Solve R_h and get q^*;
3. For each aircraft pair (i, j), compute the instant τ_{ij}^h in $[0, T]$ for which the distance between i and j is minimum and check if this distance is greater than or equal to the safety value d;
4. If for all the pairs the safety threshold is respected, the algorithm ends (q^* is an optimal solution of (1)-(3)).
   Else, for all the pairs (i, j) violating the inequality, add the cut:

$$\sum_{k \in \{1,2\}} ((x_{ik}^0 - x_{jk}^0) + \tau_{ij}^h (q_i v_i u_{ik} - q_j v_j u_{jk}))^2 \geq d^2 \tag{18}$$

5. Put $h = h + 1$ and go back to 1.

Note that we set R_1 to (1)–(2).

4 Computational experiments

We consider the set of instances proposed in [1], where n aircraft are placed on a circle of given radius r, with initial speed v_i and a trajectory defined by a heading angle such that aircraft fly toward the center of the circle (or slightly deviating with respect to such direction). See 2 from [1].

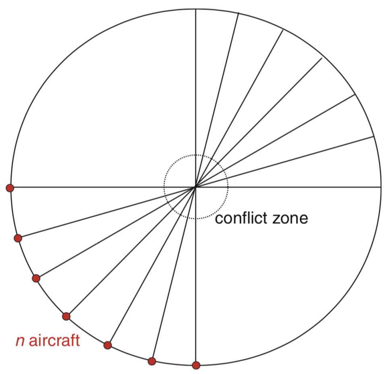

Fig. 2: n conflicting aircraft flying towards the center of a circle

Then we also consider instances always proposed in [1] in which aircraft move along straight trajectories intersecting in n_c conflict points.
We set: $T = 2$ hours, $d = 5\,\text{NM}$, $v_i = 400\,\text{NM/h}$ for each $i \in A$. For the "*circle*

instances" the heading angles cap_i are randomly generated and parameters x^0_{ik} and u_{ik} are given by

$$u_{i1} = \cos(\mathsf{cap}_i), \quad u_{i2} = \sin(\mathsf{cap}_i), \quad x^0_{ik} = -r\, u_{ik}$$

The bounds q_i^{min} and q_i^{max} are set to 0.94 and 1.03 respectively.

We implement the proposed formulations using the AMPL modeling language [10] and solve the bilevel one with the Cutting Plane algorithm before presented (CP in the Table 1) and the others with the global optimization solver Baron [11] (B in the Table 1) or, when Baron was not successful (exceeding the time-limit set to 15000 sec), with a Multistart algorithm (MS in the Table 1).

The Multistart method for the KKT reformulation uses SNOPT [12] at each iteration (1000 iterations in total), while the one for the Dual reformulation uses IPOPT [13]. Also for the Cutting Plane algorithm (CP), at each iteration we solve the relaxed formulation R_h using IPOPT, that implements an Interior-Point method, for the NLP relaxation. In the last case, to guarantee that the solution is globally optimal, when the algorithm stops because it is not necessary to add other cuts, we perform another iteration solving the last version of R_h using Baron. If the solution returned by this global solver is always the same, we can stop; otherwise other cuts are added and the algorithm keeps going.

All the solvers are run with their default settings. The tests are performed on a 2.7 GHz Intel Core i7 processor with 16GB of RAM and macOS Mojave Operating System.
Our results are reported in Table 1, and compared with those that are the best among the ones obtained with different methods in [1] and [2], that not always guarantee optimal final solutions, using a *matheuristic* approach.

Table 1

Instances			Cafieri	Bilevel			KKT reformulation			Dual reformulation		
n	n_c	r	*Best obj*	*obj*	*time(s)*	solver	*obj*	*time(s)*	solver	*obj*	*time(s)*	solver
Circle												
2	-	100	2.531e-3	2.531e-3	**0.2**	CP	**2.524e-3**	0.3	B	2.526e-3	0.4	B
3	-	200	1.667e-3	1.667e-3	1.6	CP	1.664e-3	**1.5**	B	**1.663e-3**	3.7	B
4	-	200	4.009e-3	4.029e-3	**26.3**	CP	4.025e-3	65.4	B	**4.017e-3**	184.4	B
5	-	300	**3.033e-3**	3.056e-3	**2084.2**	CP	3.052e-3	12511.1	B	3.050e-3	13978.3	B
6	-	300	**6.033e-3**	6.557e-3	1549.2	CP	6.088e-3	32.0	MS	6.096e-3	**7.8**	MS
Non-circle												
6	5	-	1.295e-3	**1.249e-3**	533.2	CP	1.254e-3	53.3	MS	1.254e-3	**14.9**	MS
7	4	-	1.617e-3	**1.571e-3**	500.1	CP	1.591e-3	238.8	MS	1.591e-3	**31.2**	MS
7	6	-	1.579e-3	**1.562e-3**	1028.3	CP	1.566e-3	86.9	MS	1.566e-3	**33.2**	MS
8	4	-	2.384e-3	**2.375e-3**	835.7	CP	2.384e-3	1163.4	MS	2.384e-3	**39.5**	MS
10	10	-	1.470e-3	**1.394e-3**	500.6	CP	1.469e-3	835.2	MS	1.397e-3	**78.9**	MS

Looking at the solutions obtained on these instances, it appears that they are comparable. The value of the objective function is always very low, given the nature of the problem (q must be in $[0.94, 1.03]$). We report in bold the best value and the minimum time required for each instance. In terms of computational performances, for the bigger instances, the reformulation which ensures the minimum solving time is the Dual one.

Acknowledgements The project leading to this application has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement MINOA No 764759.

References

1. S. Cafieri, N. Durand: Aircraft Deconfliction with Speed Regulation: New Models from Mixed-integer Optimization, *Journal of Global Optimization* **58(4)**, 613–629 (2014)
2. S. Cafieri, C. D'Ambrosio: Feasibility pump for aircraft deconfliction with speed regulation, *Journal of Global Optimization* **71(3)**, 501–515 (2018)
3. S. Cafieri, R. Omheni: Mixed-integer nonlinear programming for aircraft conflict avoidance by sequentially applying velocity and heading angle changes, *European Journal of Operational Research* **260(1)**, 283–290 (2017)
4. E. Frazzoli, Z.-H. Mao, J.-H. Oh, E. Feron: Resolution of Conflicts Involving Many Aircraft via Semidefinite Programming, *Journal of Guidance, Control and Dynamic* **24(1)**, 79–86 (2001)
5. D. Rey, C. Rapine, V.V. Dixit, S. T. Waller: Equity-Oriented Aircraft Collision Avoidance Model, *IEEE Transactions on Intelligent Transportation Systems* **16(1)**, 172–183 (2015)
6. K. D. Bilimoria: A Geometric Optimization Approach to Aircraft Conflict Resolution, *18th Applied Aerodynamics Conference* p. 4265 (2000)
7. S. Dempe, V. Kalashnikov, G.A. Prez-Valds, N. Kalashnykova: Bilevel Programming Problems: Theory, Algorithms and Applications to Energy Networks, *Springer Publishing Company* (2015)
8. W. S. Dorn: Self-Dual Quadratic Programs, *Journal of the Society for Industrial and Applied Mathematics* **9(1)**, 51–54 (1961)
9. W. S. Dorn: Duality in quadratic programming, *Quarterly of Applied Mathematics* **18(2)**, 155–162 (1960)
10. R. Fourer, D. M. Gay, B. W. Kernighan: *AMPL: A Modeling Language for Mathematical Programming* (2002)
11. N.V. Sahinidis and M. Tawarmalani: *BARON 7.2.5: Global Optimization of Mixed-Integer Nonlinear Programs*, User's Manual (2005)
12. P.E. Gill: *User's guide for SNOPT version 7.2* (2006)
13. A. Wächter and L. Biegler: On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming, *Mathematical Programming* (2006)