A Comparative Study of Different Mathematical Models for the Aircraft Deconfliction Problem

Antonio Iglesias & Tatiana Tchemisova

University of Aveiro, Portugal

Abstract

Aircraft Deconflict Problem (ADP) consists of ensuring a safe distance between flying aircraft and is a critical challenge of air traffic management. Traditionally, it is the responsibility of human Air Traffic Controllers (TAC) to handle this task by adjusting aircraft altitudes, trajectories, directions, or speeds. However, there is a growing interest in introducing automatization to aircraft deconfliction.

One intriguing approach is the concept of subliminal speed control involving subtly and softly adjusting the given initial speed of each aircraft from that concentrated in a restricted area [1]. In [2], a mathematical model based on this approach and having a form of a Semi-Infinite Programming (SIP) problem was suggested. In this model, the variable is the speed variation, and the indexed by time (infinite) constraints are aimed to guarantee a safe distance between the aircrafts at any moment of time. Due to the infinite number of limitations and the lack of reliable SIP solvers, it is difficult to obtain a sufficiently accurate numerical solution to this model, and many authors are working to reduce the SIP problem to simpler models (see e.g. [2,4], and the references therein). It is well known that using the discretization approach ([4]), a SIP problem can be reformulated in the form of a Nonlinear Programming (NLP) problem with a finite number of constraints. In this research, we solve numerically the set of real problems from the database presented in [3] using the available SIP and NLP solvers. The results of the numerical experiences are compared with some other known approaches.

Introduction

Aviation is one of the most popular and effective means of redistributing people, goods, and materials. Being the fastest and safest mode of transportation around the world, it has become a major choice for travel. With the current high intensity of aviation traffic, expected to double in the next two decades, the prevention of conflicts between aircraft in shared airspace has become a top priority. Conflicts can occur when aircraft in the same airspace are too close to each other during flight and predetermined safety margins are breached. Aircraft potential conflicts can be modeled and solved in different ways, some of which lead to different mathematical modeling approaches.

The most commonly exploited way is based on the idea of achieving separation changing the trajectory (heading) or the flight level of the aircraft involved in the conflict. This kind of separation maneuver is the one usually exploited by ATCs when they detect a potential conflict. Another way is based on the idea of separating aircraft by slightly changing their speeds but keeping the predicted trajectories. A speed regulation that occurs in a reasonably small range (namely, from -6% to +3% of the original speed), allows a subliminal control as suggested in [1]. This control consists of slightly modifying aircraft speed in an imperceptible way for ATC, but in such a way that the number of contacts is reduced upstream of the control.

In [2], for solving the aircraft deconfliction problem via subliminal speed regulation, a "Natural problem formulation" in the form of a SIP model was proposed. In order to address the issue of uncountably many constraints, the authors solve it using bilevel optimization approach. Being aware of the fact that the relaxation of the infinite number of constraints and the use of the heuristic methods for solving the relaxed model leads to a loss of the correctness of the method, we focus our attention on the SIP model and reformulate it in the form of Nonlinear Programming (NLP) problem. We prove that our reformulation is equivalent to the original SIP problem which permits us to substitute the solution of the original SIP model with that of the NLP reformulation.

References

[1] Rey D., Rapine C., Rémy F., and Faouzi N.-E. *Subliminar Speed Control in Air Traffic Management: Optimization and Simulation*. Transportation Science. Vol. 50, No. 1, 2015.

[2] Cerulli M., d'Ambrosio C., Liberti L., and Pelegrín M. *Detecting and solving aircraft conflicts using bilevel programming*, J. Global Optim., vol. 81(2), p. 529-557, 2021.

[3] Hettich R., Kortanek K.O. *Semi-infinite programming: theory, methods and applications*, SIAM Review, Vol. 35, p.380-429, 1993.

[4] Cafieri S. and d'Ambrosio C. Feasibility pump for aircraft deconfliction with speed regulation. J.Global Optim., 2018, 71 (3), pp. 501-515.

[5] Cerulli M. and Liberti L. *Polynomial programming prevents aircraft (and other) crashes*. 2020. hal-02971109v1f

[6] Database available at: public repository https: //github.com/MartinaCerulli/SRADP

[7] Software documentation at https://docs.scipy.org/doc/scipy/tutorial/optimize.html

SIP model for aircraft deconfliction

Consider the SIP model for solving the aircraft deconfliction via subliminal speed regulation introduced [1]. The decision variables q_i quantify the speed changes for each aircraft from the set A. The objective function consists of minimizing the total speed changes. An infinite number of constraints guarantee the safety distance d for each pair of aircrafts located in the airspace considered in the given time interval [0, T]. The following natural formulation follows.

$$min \sum_{i \in A} (q_i - 1)^2 \qquad (SIP)$$

$$s.t \sum_{i}^{3} [(x_{ik}^0 - x_{ij}^0) + t(q_i v_i u_{ik} - q_j v_j u_{jk})]^2 \geq d^2, t \epsilon [0, T],$$

$$q^{min} \leq q \leq q^{max}.$$

- $[\mathbf{0}, \boldsymbol{T}]$ time horizon measured in hours.
- $A = \{1,2 \ldots m\}$ represents the set of aircrafts.
- d safe distance between aircrafts, measured in Nautical Miles.
- x_{ik}^0 *k-th* component of the initial position of aircraft i.
- v_i initially planed speed of aircraft i in Nautical Miles per hour.
- u_{ik} *k-th* component of the direction of aircraft i.
- q^{min} and q^{max} minimal and maximal speed modification ratios.

The problem (SIP) can be solved using the discretization approach [3], where the infinite number of constraints' indices situated on the compact interval [0,T] is substituted by a finite grid consisting of a predefined number of points uniformly distributed on this interval, and a finite nonlinear discretized problem (SIP_D) is obtained. The discretized problem is solved using some NLP method and the optimal solution obtained is tested for the original problem (SIP). In the case the solution is not optimal, the greed is refined. To guarantee that this approach converges to an accurate solution, the model should satisfy several rather struct regularity assumptions.

NLP model for aircraft deconfliction

To avoid difficulties connected with exact solution of problem (SIP), an equivalent formulation in the form of the NLP problem with only two constraints, is suggested here. This approach is based on the following.

CONDITION 1: The following inequalities hold true for all $t \in [0, T]$:
$f(t) \coloneqq at^2 - 2bt + c \geq 0 \; for \; all \; t \in [0, T]$.
CONDITION 2: The following propositions hold true:
$f(0) = c \geq 0, \; f(T) = aT^2 - 2bT + c \geq 0, \; \sqrt{f(0)f(T)} \geq Tb - c.$
PROPOSITION 1: *Conditions 1 and 2 are equivalent.*

Based on **Proposition 1** and having represented the constraint function in (SIP) in the form $f(t) \coloneqq at^2 - 2bt + c$, where t is the time index, we can reformulate the **SIP** problem into the form of a **NLP** problem in the following manner:

$$min \sum_{i \in A} (q_i - 1)^2 \qquad (NLP)$$

$$s.t \quad f(T, q) \geq 0,$$

$$\sqrt{f(0, q) f(T, q)} \geq b(T, q) - c,$$

$$q^{min} \leq q \leq q^{max}.$$

This model has a finite number of nonlinear constraints, allowing not only to have a wider range of solution methods (NLP software), but also to reduce the compilation time.

Numerical Experiences

To solve problem (SIP) using the discretization approach, the continuous time interval of 2 hours was divided into 20 subintervals of 6 min each, and the obtained discretized problem was solved using two standard methods from [7]: ***Trust-constraint*** (a variant of ***Interior Point Method***) and the ***Differential Evolution methods***.

The results of the experiences are presented in Tables 1 and 2.

Instances		Results	
n	Radius	Obj.Val	Time(s)(min)
2	100	0.0042	36.40 (0.60)
3	200	0.0054	428.28 (7.13)
4	200	0.0092	1400.62 (23.34)
5	300	0.0052	3324.70 (55.41)
6	300	0.0058	6062.08 (101.03)
8	400	0.0178	12851.97 (214.19)
9	500	0.0186	18208.80 (303.48)
10	600	0.0201	28191.44 (406.87)
12	700	0.0196	44644.66 (744.07)

Table 1:Trust Constraint results

Instances		Results		
n	Radius	Obj.Val	Steps to convergence	Time(s)
2	100	0.00010	1	0.386
3	200	0.00015	1	0.83
4	200	0.0029	1	1.84
5	300	0.0134	1	3.16
6	300	0.0524	1	4.88
8	400	0.0773	1	11.51
9	500	0.0831	1	16.90
10	600	0.0943	1	22.58
12	700	0.144	1	69.36

Table 2: Differential Evolution results

TRUST CONSTRAINT
1. Unacceptable compilation times.
2. In 40% of cases safety is not verified.

DIFFERENTIAL EVOLUTION
1. Fast compilation times.
2. Verified safety at all instances

NLP RESULTS

To solve the NLP formulation, we used ***Trust-Constraint***, ***Differential Evolution, Nelder-Mead, TNC***, ***L-BFGS-B***, ***Powell*** and some other methods from [7].

We present here the results of the numerical experiences that presented the best results***:*** ***Trust-Constraint, Quase Newton method BFGS-B***, and the ***Powell method***.

Instances		Results
n	Radius	Obj.Val
2	100	$1.15 \cdot 10^{-11}$
3	200	$1.24 \cdot 10^{-9}$
4	200	$8.29 \cdot 10^{-9}$
5	300	$6.85 \cdot 10^{-9}$
6	300	$7.80 \cdot 10^{-9}$
7	400	$3.18 \cdot 10^{-7} (*)$
8	400	$7.21 \cdot 10^{-7}$
9	500	$2.75 \cdot 10^{-6}$
10	600	$3.07 \cdot 10^{-6}$
12	700	$3.24 \cdot 10^{-5}$

Instances		Results	
n	Radius	Obj.Val	Time(s)
2	100	0.0072*	0.010
3	200	0.0090*	0.043
4	200	0.0144	0.020
5	300	0.0180	0.059
6	300	0.0216*	0.070
7	400	0.0252*	0.028
8	400	0.0288	0.081
9	500	0.0324*	0.105
10	600	0.0360	0.183
12	700	0.0432	0.290

Table 3: Trust-Constraint and L-BFGS-B method results

Trust Constraint
- Much better numerical results than the **SIP version** in terms of Objective Value**.**
- Acceptable computation time.
- Feasible solutions (verifies safety at every point).

L-BFGS-B method
- Satisfactory numerical results.
- Extremely fast computational times.
- Poor results in terms of safety (50% safe deconflictions).

Instances		Results	
n	Radius	Obj.Val	Time(s)
2	100	0.0072*	0.039
3	200	0.0097*	0.472
4	200	0.0143	0.452
5	300	0.0179	0.801
6	300	0.0215	1.492
7	400	0.0251	2.409
8	400	0.0287	3.587
9	500	0.0323	5.389
10	600	0.0359	7.400
12	700	0.0431	12.221

Table 4: Powell's method results

Powell's method
- Satisfactory numerical results.
- Fast computational times.
- Good results in terms of safety (80% of safe deconflictions).

Software, Hardware and DataBase

For the numerical experiences, the ***Matlab (fmincon)*** and ***Python (Scipy.minimize)*** were used, as both are recommended tools for the resolution of optimization problems.

The hardware used was:

ASUS H97M-E i7-4790 3.6GHz SK H3 1150 for the resolution of the **SIP** problem, as this kind of problem requires larger computational power and a ***ZenBook UX325 ,11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz*** for the **NLP** problem.

The data (the angles defining the movement of each aircraft) was extracted from the **Github** repository [6].

Discussion

In Table 6, we compare behavior of the tested NLP solvers for the Aircraft Deconflict Problem in the form (NLP). Note that the Matlab solver **fmincon** was not able to find feasible solutions, and its performance is marked by **x** in the Table.

Solvers (NLP)	Numerical Results	Convergence Times (s)	Safe deconflictions (%)
L-BFGS-B (NLP)	Satisfactory	Best	50
Trust-Constraint (NLP)	Best	Worst	100
Trust-Constraint (SIP)	Satisfactory	Worst	60
Differential Evolution (NLP)	Satisfactory	Bad	80
Differential Evolution (SIP)	Satisfactory	Average	100
Nelder-Mead	Satisfactory	Average	80
Powell	Satisfactory	Good	80
TNC	Satisfactory	Average	50
fmincon	x	x	x

Table 5: Comparison of methods

Conclusion

Dealing with aircraft deconfliction is an intricate challenge demanding hard research efforts to discover optimal problem-solving methods. We have suggested a NLP formulation that is equivalent to the original SIP model.

The numerical experiences show that the use of this formulation can lead to a significant reduction in compilation time. The most efficient method for the NLP formulation is the ***Powell*** algorithm

Future work

In the field of plane deconfliction, future work should focus on:
1. **UAV Integration**: To study the impact of autonomous and unmanned aerial vehicles on deconfliction protocols.
2. **Communication**: To improve data sharing between aircraft and air traffic control for real-time decision-making.
3. **Human Factors**: To investigate pilot decision-making and workload management to enhance performance during deconfliction scenarios.
4. **Regulatory Frameworks**: To assess and update regulations to improve deconfliction practices.

Acknowledgements

This work was supported by Portuguese funds through CIDMA - Center for Research and Development in Mathematics and Applications, University of Aveiro, Portugal, and FCT - Portuguese Foundation for Science and Technology, within the project UIDB/04106/2020.