MINLP emerging applications in Air Traffic Management

Sonia Cafieri

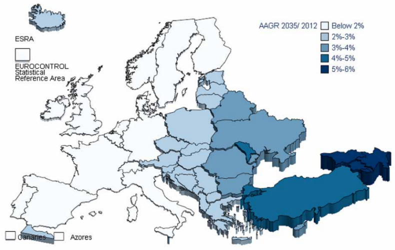
Laboratoire MAIAA
ENAC - École Nationale de l'Aviation Civile
University of Toulouse
France

Mixed-Integer Nonlinear Programming 2014
Carnegie Mellon University, Pittsburgh
June 2014

Air Traffic Management (ATM)

ATM : making sure that aicraft are safely guided in the skies and on the ground

Air Traffic growing
on the world scale



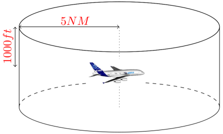
Eurocontrol forecast

BOEING long-term market forecast

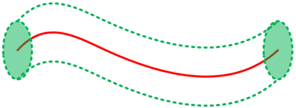
⇒ needs increasing automation

ATM applications

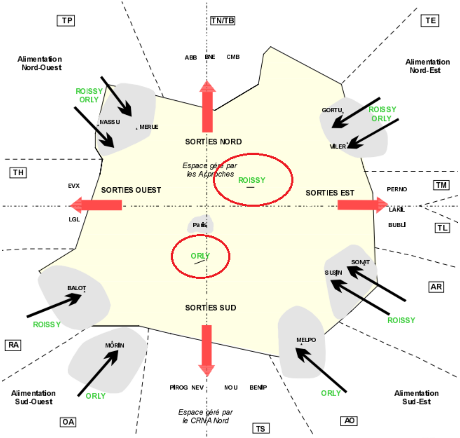
Aircraft separation for collision avoidance



Conflict-free trajectory planning



Design of TMAs: SID/STAR routes



ENAC

Outline

Outline

MINLP for ATM applications

ATM applications → MINLP

- Aircraft conflict avoidance
- Aircraft conflict-free trajectory planning
- Design of arrival and departure procedures (SID/STAR)
- ...

- *integer variables*:
  logical choices and possible different scenarios
- *nonlinearities*:
  complex ATM nonlinear processes:
  - aircraft separation
  - obstacle avoidance
  - noise restrictions
  - ...

Outline

Aircraft Conflict Avoidance

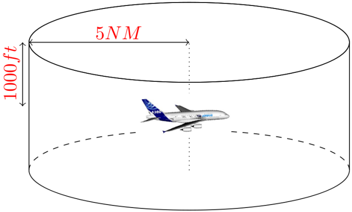
Aircraft i and j are **in conflict** if

their horizontal distance is less than d:

$||x_i - x_j|| \leq d \qquad (d = 5\text{NM})$

their altitude difference is less than h:

$||h_i - h_j|| \leq h \qquad (h = 1000\text{ft})$



1 NM (nautical mile)= 1852 m
1 ft (feet) = 0.3048 m

Aircraft Conflict Avoidance

- Resolution of conflicts currently still largely performed manually by air traffic controllers
- SESAR & NextGen projects: promote automation

Aircraft separation strategies

- *Heading angle deviation* or *altitude modification*

  $\Rightarrow$ most used

- *Speed adjustements*

  $\Rightarrow$ suggested by ERASMUS (En-Route Air Traffic Soft Management Ultimate System) project (2006-2009), allows one to perform a ***subliminal control***

Aircraft conflict resolution: literature

- Optimal Control (*Tomlin et al.* 2004, *Cellier et al.* 2012)
- Evolutionary computation (*Durand&Alliot* 1995, 1998, *Delahaye et al.* 1996)
- **Mathematical Programming based approaches:**
  Mixed-Integer Linear and Nonlinear Optimization
  - *Richards & How,* 2002 (MILP)
  - *Pallottino, Feron, Bicchi,* 2004 (MILP)
  - *Christodoulou & Costoulakis,* 2004 (MINLP)
  - *Vela et al.,* 2010 (MILP)
  - *Alonso-Ayuso, Escudero, Martín-Campo,* 2011, 2012 (MILP, MINLP)
  - *Rey et al.,* 2012 (MILP)
  - *Cafieri & Durand,* 2013 (MINLP)

  in general, subject to some hypoteses
  (constant speed, all maneuvers performed at the same time, etc.)

ENAC

Outline

Outline

MINLP formulation overview

Decision variables

- $\forall k \in A \quad q_k$

  speed change of aircraft k in set A

Separation based on *speed regulation*

★ q_k = percentage of change (to be multiplied by v_k to obtain the new speed)

$q_k > 1$: acceleration, $q_k < 1$: deceleration, q_k=1: no speed change.

★ $\forall k \quad -6\% \, v_k \leq q_k \leq +3\% \, v_k \quad \Rightarrow$ subliminal control

- other variables, *both continuous and integer*

  used to model logical choices and as intermediate variables to express **separation**

Constraints: *nonlinear nonconvex*

used to model aircraft **separation**

and possible configurations that can occur

Aircraft separation

Separation condition for aircraft i and j:

$$\|\mathbf{x}^r_{ij}(t)\| \geq d \qquad \forall t \in (0, T)$$

* d = minimum required separation distance
* $\mathbf{x}^r_{ij}(t) = x_i(t) - x_j(t) = \mathbf{x}^{r0}_{ij} + \mathbf{v}^r_{ij}\, t$

  $\mathbf{x}^{r0}_{ij}$ = relative initial position of aircraft i and j

  $\mathbf{v}^r_{ij}$ = relative speed of aircraft i and j

Aircraft separation

Separation condition for aircraft i and j:

$$\|\mathbf{x}^r_{ij}(t)\| \geq d \qquad \forall t \in (0, T) \qquad \leftarrow \text{ depends on } t$$

* d = minimum required separation distance
* $\mathbf{x}^r_{ij}(t) = x_i(t) - x_j(t) = \mathbf{x}^{r0}_{ij} + \mathbf{v}^r_{ij}\, t$

  $\mathbf{x}^{r0}_{ij}$ = relative initial position of aircraft i and j

  $\mathbf{v}^r_{ij}$ = relative speed of aircraft i and j

$$\Downarrow$$

$$\|\mathbf{x}^{r0}_{ij} + \mathbf{v}^r_{ij}\, t\|^2 \geq d^2 \qquad \forall t \in (0, T)$$

i.e.

$$(v^r_{ij})^2\, t^2 + 2(\mathbf{x}^{r0}_{ij}\mathbf{v}^r_{ij})\, t + ((x^{r0}_{ij})^2 - d^2) \geq 0$$

Aircraft separation

By differentiation, the minimum occurs at

$$t_{ij}^{m} = -\frac{\mathbf{v}_{ij}^{r}\mathbf{x}_{ij}^{r0}}{(v_{ij}^{r})^2}$$

Substituting:

$$(x_{ij}^{r0})^2 - \frac{(\mathbf{v}_{ij}^{r}\mathbf{x}_{ij}^{r0})^2}{(v_{ij}^{r})^2} - d^2 \geq 0$$

Aircraft separation

By differentiation, the minimum occurs at

$$t_{ij}^{m} = -\frac{\mathbf{v}_{ij}^{r}\mathbf{x}_{ij}^{r0}}{(v_{ij}^{r})^2}$$

Substituting:

$$(x_{ij}^{r0})^2 - \frac{(\mathbf{v}_{ij}^{r}\mathbf{x}_{ij}^{r0})^2}{(v_{ij}^{r})^2} - d^2 \geq 0 \qquad \leftarrow \text{does not depend on } t$$

(separation condition)

Aircraft separation

By differentiation, the minimum occurs at

$$t_{ij}^m = -\frac{\mathbf{v}_{ij}^r \mathbf{x}_{ij}^{r0}}{(v_{ij}^r)^2}$$

Substituting:

$$(x_{ij}^{r0})^2 - \frac{(\mathbf{v}_{ij}^r \mathbf{x}_{ij}^{r0})^2}{(v_{ij}^r)^2} - d^2 \geq 0 \qquad \leftarrow \text{does not depend on } t$$

(separation condition)

Separation of i and j:

- (separation in $t^0 = 0$): $x_{ij}^{r0} \geq d$ ← checked in preprocessing step
- **if** $t_{ij}^m > 0$ **then** condition $(x_{ij}^{r0})^2 - \dfrac{(\mathbf{v}_{ij}^r \mathbf{x}_{ij}^{r0})^2}{(v_{ij}^r)^2} - d^2 \geq 0$

MINLP formulation: variables and objective

Variables:

- q_i, $(\geq q_{min}, \leq q_{max})$, $\forall i \in A$ percentage of velocity change of aircraft i *(continuous)*
- v_{ij}^{2r}, $\forall (i,j) \in A, i < j$ square of the relative velocity of i and j: $(v_{ij}^r)^2$ *(continuous)*
- p_{ij}, $\forall (i,j) \in A, i < j$ inner product $\mathbf{v}_{ij}^r \mathbf{x}_{ij}^{r0}$ *(continuous)*
- t_{ij}^m, $\forall (i,j) \in A, i < j$ *(continuous)*
- y_{ij}, $\forall (i,j) \in A, i < j$, used to check if $t_{ij}^m > 0$ *(binary)*

Objective:

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

MINLP formulation: constraints

Constraints:

- definition of v_{ij}^{2r} *(quadratic)*

$$v_{ij}^{2r} = \sum_{k=1}^{2} (q_i v_i u_{ik} - q_j v_j u_{jk})^2 \qquad \forall i,j \in A, i < j$$

- inner product in the separation condition *(linear)*

$$p_{ij} = \sum_{k=1}^{2} \left(x_i^0 u_{ik} - x_j^0 u_{jk}\right)\left(q_i v_i u_{ik} - q_j v_j u_{jk}\right) \qquad \forall i,j \in A, i < j$$

- definition of t_{ij}^m *(bilinear)*

$$t_{ij}^m v_{ij}^{2r} + p_{ij} = 0 \qquad \forall (i,j) \in A, i < j$$

- check sign of t_{ij}^m *(bilinear with binary var.)*

$$t_{ij}^m \,(2y_{ij} - 1) \geq 0 \qquad \forall (i,j) \in A, i < j$$

- separation *(quadratic + linear term, product with binary var.)*

$$y_{ij}\left((x_{ij}^{0r} v_{ij}^{2r}) - (p_{ij})^2 - ((d)^2\ v_{ij}^{2r})\right) \geq 0 \qquad \forall (i,j) \in A, i < j$$

MINLP modeling: further developments

- Mathematical Programming point of view:

  reformulate nonlinear terms

- ATM modeling point of view:

  get closer to a realistic situation

  avoiding separation maneuver performed simultaneously by all aircraft

MINLP model reformulation

- Compute bounds on variables v_{ij}^r, p_{ij} and t_{ij}^m ($\rightarrow$ obtain $(t_{ij}^m)_{min}$, $(t_{ij}^m)_{max}$) (taking into account bounds on q_i)
- Reformulate products of binary variables y_{ij} and continuous variables (exact reformulations):
  - reformulate

$$t_{ij}^m\,(2y_{ij}-1) \geq 0 \qquad \forall (i,j) \in A, i<j$$

to

$$\begin{aligned} t_{i,j}^m &\geq (t_{ij}^m)_{min}\,(1-y_{ij}) \qquad \forall (i,j) \in A, i<j \\ t_{i,j}^m &\leq (t_{ij}^m)_{max}\, y_{ij} \end{aligned}$$

  - reformulate

$$y_{ij}\left((x_{ij}^{0r} v_{ij}^{2r}) - (p_{ij})^2 - ((d)^2\; v_{ij}^{2r})\right) \geq 0 \qquad \forall (i,j) \in A, i<j$$

to

$$(x_{ij}^{0r} - d^2)\, v_{ij}^{2r} - (p_{ij})^2 \geq bigM_{ij}\,(1-y_{ij}) \qquad \forall (i,j) \in A, i<j$$

More general MINLP model

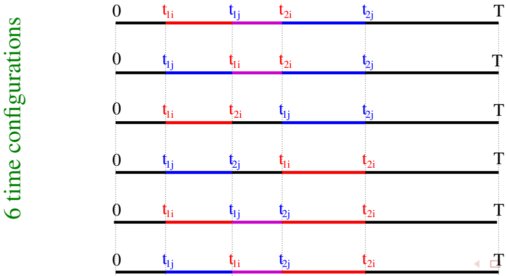
Main idea:

- each aircraft k can change its speed at time t_k^1 and go back to its original speed at t_k^2
- for a pair of aircraft, no order *a priori* on instant times $\Rightarrow$ **several configurations** possible
- 4 instant times to be handled for a pair of aircraft $\Rightarrow$ 5 **time windows**
  where aircraft fly with their original speed v or with a changed speed $v\ q$

- t_i^1, t_j^1 instant times when i and j *start flying* with changed speed
- t_i^2, t_j^2 instant times when i and j *end flying* with changed speed

$\Downarrow$

5 time intervals $\forall$ configuration



More general MINLP model

Key novelty in the model: handle time configurations and time intervals, for each of them impose separation for each pair of aircraft (velocity piecewise constant)

$\Rightarrow$ for each time "segment", consider

- relative initial position of aircraft
- relative distance
- relative velocity
- velocity of each aircraft

} many more variables and constraints

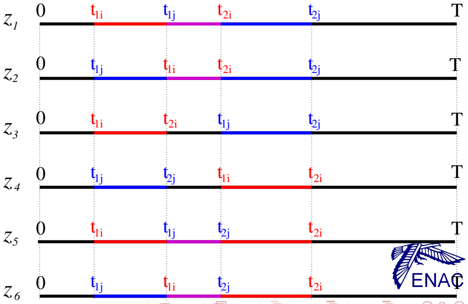
Binary variables: $z_{ij}^{\ell},\ \ell \in \{1, \ldots, 6\}$

state what is the *order of instant times* for each time configuration

Example:

$z_{ij}^1 = 1 \quad \Leftrightarrow \quad t_i^1 \leq t_j^1$ and $t_j^1 \leq t_i^2$ and $t_i^2 \leq t_j^2$

$\rightarrow$ bigM constraints



Outline

Numerical solution

Solution approaches

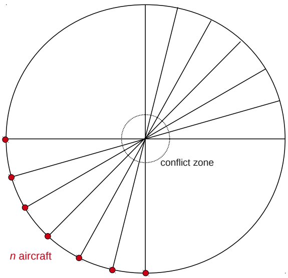
- Global exact solution
- Matheuristic approach using clusters
- Global exact solution on the reformulated problem
- Feasibility Pump

Numerical solution

Solution approaches

- Global exact solution
- Matheuristic approach using clusters
- Global exact solution on the reformulated problem
- Feasibility Pump

Test problems



$n \in \{1, \ldots, 10\}$ aircraft

$n(n-1)/2$ conflicts

$d = 5$ NM, $v = 400$ NM/h

Global exact solution

deterministic Global Optimization: *spatial Branch-and-Bound*

COUENNE software for MINLP (Belotti et al., 2008)

n	r	COUENNE time (sec)
2	1×10^2	0.11
3	2×10^2	0.98
4	2×10^2	8.43
5	3×10^2	469.86
6	3×10^2	46707.03

Global exact solution

deterministic Global Optimization: *spatial Branch-and-Bound*

`COUENNE` software for MINLP (Belotti et al., 2008)

n	r	COUENNE time (sec)
2	1×10^2	0.11
3	2×10^2	0.98
4	2×10^2	8.43
5	3×10^2	469.86
6	3×10^2	46707.03

- 2 to 4 aircraft: efficient solution
- $>$ 4 aircraft: higly time and memory demanding

S. Cafieri & N. Durand, JOGO 58(4):613-629, 2014

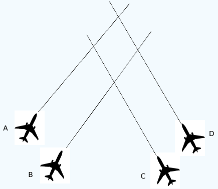
Matheuristic approach

★ *Decompose the problem* in subproblems:

→ aircraft clusters (up to η aircraft at a time ($\eta < 4$))

★ Solve on clusters using an exact solver (COUENNE)

→ *local exact solutions*

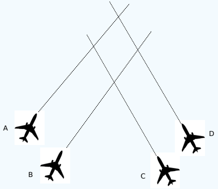


Cluster: ABCD

Matheuristic approach

★ *Decompose the problem* in subproblems:
→ aircraft clusters (up to η aircraft at a time ($\eta < 4$))

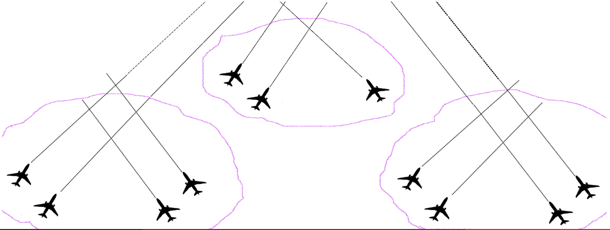
★ Solve on clusters using an exact solver (COUENNE)
→ *local exact solutions*



Cluster: ABCD

Iteratively (until all conflicts are solved):

- get local exact solutions
- re-initialize aircraft speeds keeping acceleration/deceleration from local solutions
- perform local search to update the aircraft speeds and take into account bounds from ERASMUS



Matheuristic approach solution

n	r	COUENNE	Matheuristic	
		time (sec)	ncl	time (sec)
2	1×10^2	0.11	1	-
3	2×10^2	0.98	1	-
4	2×10^2	8.43	2	1.02
5	3×10^2	469.86	2	3.32
6	3×10^2	46707.03	2	48.97
7	3×10^2	-	2	88.67
8	3×10^2	-	2	121.88

promising results hybridizing heuristics and mathematical programming, but optimality only on subproblems

Matheuristic approach solution

n	r	COUENNE	Matheuristic	
		time (sec)	ncl	time (sec)
2	1×10^2	0.11	1	-
3	2×10^2	0.98	1	-
4	2×10^2	8.43	2	1.02
5	3×10^2	469.86	2	3.32
6	3×10^2	46707.03	2	48.97
7	3×10^2	-	2	88.67
8	3×10^2	-	2	121.88

promising results hybridizing heuristics and mathematical programming, but optimality only on subproblems

S. Cafieri & N. Durand, JOGO 58(4):613-629, 2014

Global exact solution - reformulated problem

n	r	COUENNE	Matheuristic		COUENNE ref.
		time (sec)	ncl	time (sec)	time (sec)
2	1×10^2	0.11	1	-	0.12
3	2×10^2	0.98	1	-	0.94
4	2×10^2	8.43	2	1.02	8.48
5	3×10^2	469.86	2	3.32	167.03
6	3×10^2	46707.03	2	48.97	7031.51
7	3×10^2	-	2	88.67	-
8	3×10^2	-	2	121.88	-

significantly faster than solution of the original problem,
exact solution

Feasibility Pump

General scheme of Feasibility Pump

for MINLPs

$$\begin{aligned} \min\ & f(x,y) \\ g(x,y) & \leq 0 \\ x & \in X \\ y & \in Y \cap \mathbb{Z}^n \end{aligned}$$

it=0;

while $(((\hat{x}^{it}, \hat{y}^{it}) \neq (\bar{x}^{it}, \bar{y}^{it}))$ **or** time limit) **do**

  Solve $(P1)$ (the problem obtained relaxing integrality requirements and minimizing a “distance” with respect to $(\hat{x}^{it}, \hat{y}^{it})$);

  Solve $(P2)$ (the problem obtained relaxing “complicated” constraints and minimizing a “distance” with respect to $(\bar{x}^{it}, \bar{y}^{it})$);

  it++;

end while

(D’Ambrosio, Frangioni, Liberti, Lodi, 2012)

Feasibility Pump for conflict avoidance (1/3)

- $(P1)$: NLP - continuous relaxation of the (reformulated) MINLP
- $(P2)$: MIQP - quadratic objective, linear relaxation of the constraints
  - McCormick's relaxation for bilinear terms
  - Linear relaxation by tangents and secant for quadratic terms

with C. D'Ambrosio, work in progress

Feasibility Pump for conflict avoidance (2/3)

At iteration *it* of FP:

- $(P1)^{it}$: nonconvex NLP

  A feasible solution is computed minimizing the Hamming distance wrt to solution of $(P2)^{it}$:

  Objective: $$\min \sum_{i,j \in A, i<j:\ \hat{y}_{ij}=1} (1 - y_{ij}) + \sum_{i,j \in A, i<j:\ \hat{y}_{ij}=0} y_{ij}$$

  If no feasible solution with y variables assuming values $\hat{y}^{it}$,
  solved to local optimality multiple times, using randomly generated starting points $\rightarrow$ IPOPT

  When an optimal solution is found, try to improve the objective of the original problem:

  Optimality cut: $$\sum_{i \in A} (1 - q_i)^2 \leq \sum_{i \in A} (1 - \bar{q}_i)^2 - \epsilon$$

Feasibility Pump for conflict avoidance (3/3)

At iteration it of FP:

- $(P2)^{it}$: considering integrality requirements

  Solved using CPLEX

  Objective: $$\min \sum_{i \in A} (q_i - \bar{q}_i)^2$$

  minimizing the distance wrt solution of $(P1)^{it-1}$

Feasibility Pump solution

n	r	COUENNE	Matheuristic		COUENNE ref.	Feas. Pump
		time (sec)	ncl	time (sec)	time (sec)	time (sec)
2	1×10^2	0.11	1	-	0.12	0.69
3	2×10^2	0.98	1	-	0.94	8.86
4	2×10^2	8.43	2	1.02	8.48	7.08
5	3×10^2	469.86	2	3.32	167.03	15.12
6	3×10^2	46707.03	2	48.97	7031.51	54.53
7	3×10^2	-	2	88.67	-	81.24 *
8	3×10^2	-	2	121.88	-	315.25

feasible solution found quickly on large problems,
no guarantee of optimality

Feasibility Pump solution

n	r	COUENNE	Matheuristic		COUENNE ref.	Feas. Pump
		time (sec)	ncl	time (sec)	time (sec)	time (sec)
2	1×10^2	0.11	1	-	0.12	0.69
3	2×10^2	0.98	1	-	0.94	8.86
4	2×10^2	8.43	2	1.02	8.48	7.08
5	3×10^2	469.86	2	3.32	167.03	15.12
6	3×10^2	46707.03	2	48.97	7031.51	54.53
7	3×10^2	-	2	88.67	-	81.24 *
8	3×10^2	-	2	121.88	-	315.25

feasible solution found quickly on large problems,
no guarantee of optimality

S. Cafieri & C. D'Ambrosio, work in progress

Solution: comparison

n	r	COUENNE	Matheuristic		COUENNE ref.	Feas. Pump
		time (sec)	ncl	time (sec)	time (sec)	time (sec)
2	1×10^2	0.11	1	-	0.12	0.69
3	2×10^2	0.98	1	-	0.94	8.86
4	2×10^2	8.43	2	1.02	8.48	7.08
5	3×10^2	469.86	2	3.32	167.03	15.12
6	3×10^2	46707.03	2	48.97	7031.51	54.53
7	3×10^2	-	2	88.67	-	81.24 *
8	3×10^2	-	2	121.88	-	315.25

Time to find a feasible solution

n	COUENNE	COUENNE ref.
2	0.00	0.00
3	0.00	0.00
4	6.44	8.29
5	0.00	0.00
6	41649.91	3934.24
7	-	-
8	-	-

Objective function value

n	COUENNE	Feas. Pump
2	0.002531	0.002572
3	0.001667	0.001744
4	0.004022	0.004030
5	0.003035	0.003112
6	0.006015	0.006091
7	-	-
8	-	0.008749

Outline

Conclusions and Perspectives

- MINLP modeling and solution approaches interesting in the ATM context
- Future work:
  - extension of the computational approach to the more general MINLP model
  - testing on different kinds of instances
  - testing with other MINLP solvers (BARON, MINOTAUR, IBBA)
  - in FP, solving $(P1)$ with a different technique, defining $(P2)$ as a MILP
  - MINLP modeling: trajectory planning and optimal departure/arrival route design

The end

Thank you!