7th International Conference on Research in Air Transportation

A Variable Neighborhood Search approach for the aircraft conflict resolution problem

F. Javier Martín Campo

javier.martin.campo@mat.ucm.es

Joint work with:

A. Alonso-Ayuso, L.F. Escudero, N. Mladenović

Outline

Introduction

Variable Neighborhood Search (VNS)

The Velocity, Turn and Altitude Changes model (VTAC)

Multi-objective methods

Computational experience

Conclusions and future research

Outline

Introduction

Variable Neighborhood Search (VNS)

The Velocity, Turn and Altitude Changes model (VTAC)

Multi-objective methods

Computational experience

Conclusions and future research

Motivation

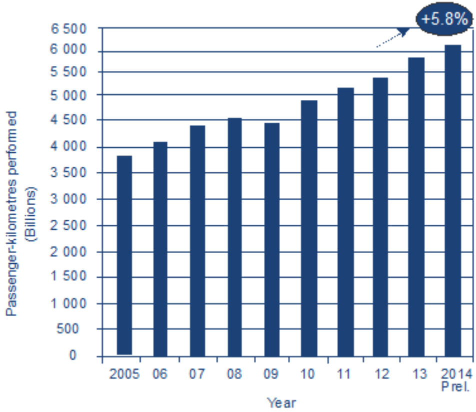
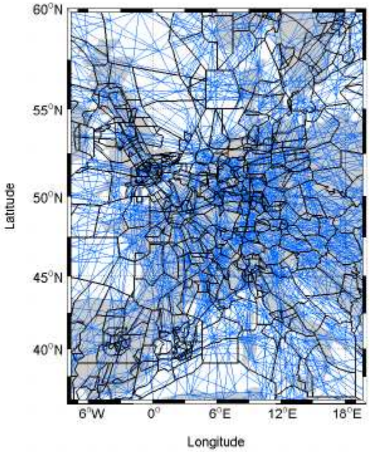


Figure 1. Total scheduled traffic
(revenue passenger-kilometres performed, 2005-2014)

International Civil Aviation Organization

- An air sector is considered.
- Enroute phase.
- Short term (up to 5 minutes).
- Static.



A. Lau, J. Berling, F. Linke, V. Gollnick, K. Nachtigall. Large-Scale Network Slot Allocation with Dynamic Time Horizons

Problem Objective

Given a set of flight configurations (waypoints, velocities, angles of motion, altitude level, etc.)

Problem Objective

Given a set of flight configurations (waypoints, velocities, angles of motion, altitude level, etc.)

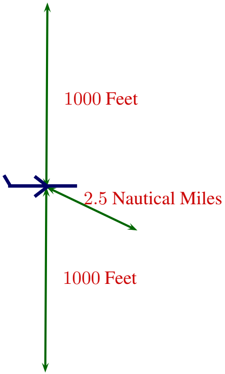
Problem objective

What control strategy should be followed by the pilots and the air traffic service provider to prevent the aircraft from coming too close to each other?

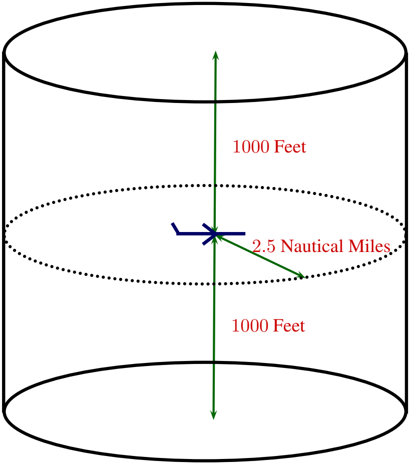
Conflict definition: It is an event in which two or more aircraft experience a loss of minimum separation.

Safety distances

Safety distances



Safety distances



How to avoid conflict situations?

In order to avoid conflict situations, maneuvers that an aircraft can perform:

How to avoid conflict situations?

In order to avoid conflict situations, maneuvers that an aircraft can perform:

Types of maneuvers

- Horizontal:
  - **Velocity** changes.
  - **Heading angle** changes.

How to avoid conflict situations?

In order to avoid conflict situations, maneuvers that an aircraft can perform:

Types of maneuvers

- Horizontal:
  - **Velocity** changes.
  - **Heading angle** changes.
- Vertical:
  - **Altitude** changes.

Literature based on Mathematical Optimization

Authors who have worked on the topic with mathematical optimization (among others):

- N. Durand, G. Granger and S. Cafieri.
- L. Pallottino, E. Feron and A. Bicchi.
- A.G. Richards and J.P. How.
- M.A. Christodoulou and C. Costoulakis.
- J. Omer, J. Farges and T. Lehouillier.
- M. Soler, M. Kamgarpour, J. Lloret and J. Lygeros
- C. Peyronne, A.R. Conn, M. Mongeau and D. Delahaye.
- D. Rey, C. Rapine, V. Dixit, S.T. Waller, R. Fondacci and N.E. El Faouzi.
- A. Vela and S. Solak.

Principles of VNS

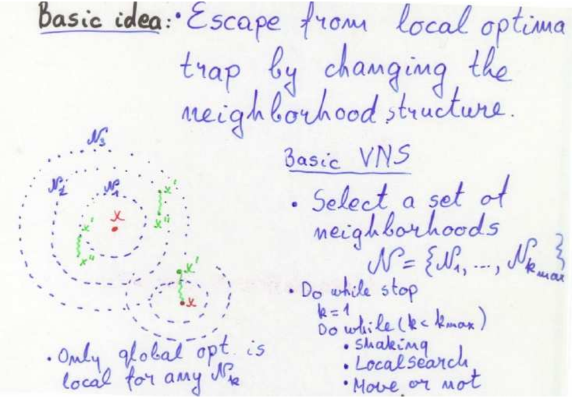
- Introduced by Mladenović and Hansen (1995).
- It has been successfully applied to many combinatorial optimization problems among others.

Principles of VNS

- Introduced by Mladenović and Hansen (1995).
- It has been successfully applied to many combinatorial optimization problems among others.

Facts

- A local minimum with respect to one neighborhood structure is not necessary so for another.
- A global minimum is a local minimum with respect to all possible neighborhood structures.
- For many problems local minima with respect to one or several neighborhoods are relatively close to each other.



Extracted from a VNS tutorial (N. Mladenović)

Outline

The Velocity Changes (VC) model

Pallottino, Feron and Bicchi (2002), "*Conflict resolution problems for air traffic management systems solved with mixed integer programming*", **IEEE, Transactions on Intelligent Transportation Systems 3(1), 3–11**.

Pallottino, Feron and Bicchi (2002), "*Conflict resolution problems for air traffic management systems solved with mixed integer programming*", **IEEE, Transactions on Intelligent Transportation Systems 3(1), 3–11**.

Two different MILO models are studied: Velocity (VC) and Heading Angle Changes (HAC).

The Velocity Changes (VC) model

Pallottino, Feron and Bicchi (2002), "*Conflict resolution problems for air traffic management systems solved with mixed integer programming*", **IEEE, Transactions on Intelligent Transportation Systems 3(1), 3–11**.

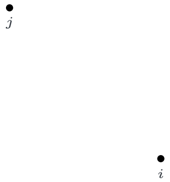
Two different MILO models are studied: Velocity (VC) and Heading Angle Changes (HAC).

Features of the VC model

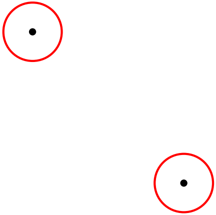
- It does not allow neither **altitude** nor **heading angle** changes causing several infeasible situations.
- It is incomplete due to specific **cases that are not solved**.
- It is based on **geometric constructions**.

Geometric Construction I

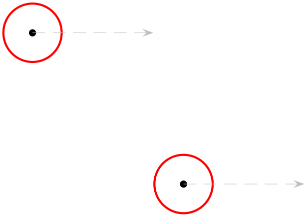
Geometric Construction I



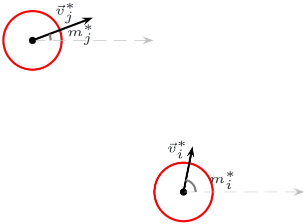
Geometric Construction I



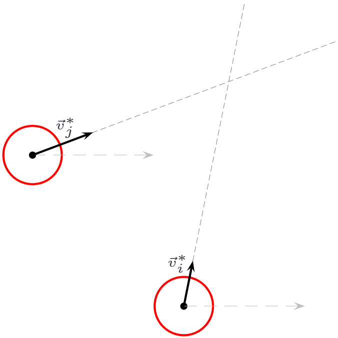
Geometric Construction I



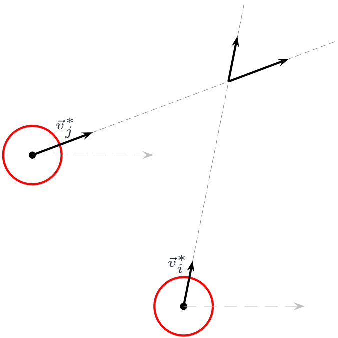
Geometric Construction I



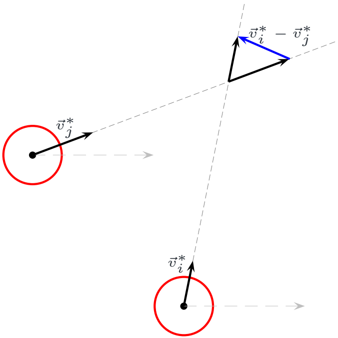
Geometric Construction I



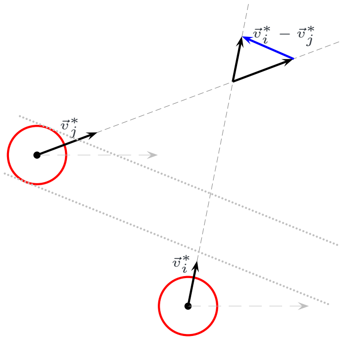
Geometric Construction I



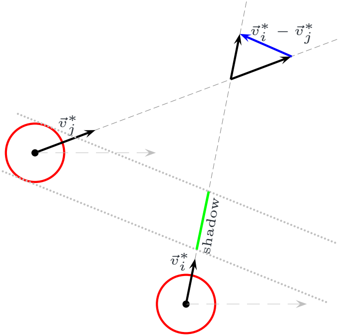
Geometric Construction I



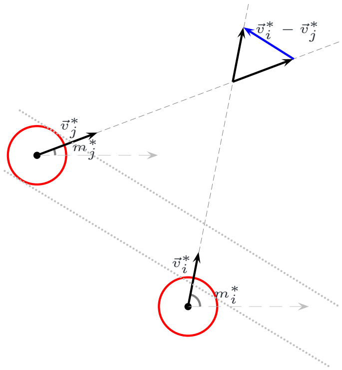
Geometric Construction I



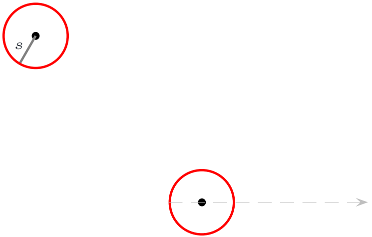
Geometric Construction I



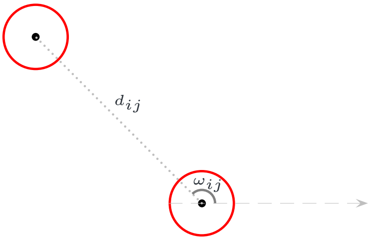
Conflict Situation



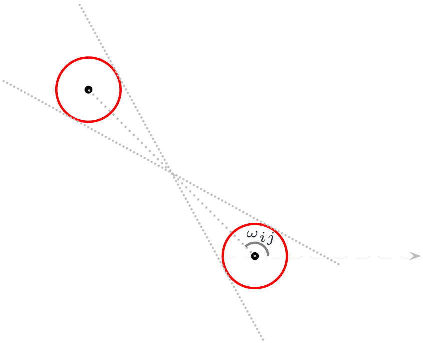
Geometric Construction II



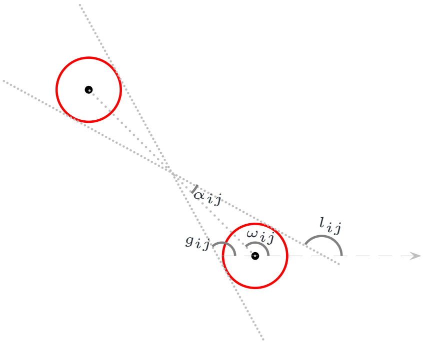
Geometric Construction II



Geometric Construction II



Geometric Construction II



No conflict constraint (aircraft at same altitude level)

$$\frac{(v_i^* \qquad) \sin(m_i^* \qquad) - (v_j^* \qquad) \sin(m_j^* \qquad)}{(v_i^* \qquad) \cos(m_i^* \qquad) - (v_j^* \qquad) \cos(m_j^* \qquad)} \geqslant \tan(l_{ij})$$

or

$$\frac{(v_i^* \qquad) \sin(m_i^* \qquad) - (v_j^* \qquad) \sin(m_j^* \qquad)}{(v_i^* \qquad) \cos(m_i^* \qquad) - (v_j^* \qquad) \cos(m_j^* \qquad)} \leqslant \tan(g_{ij})$$

No conflict Constraints

No conflict constraint (aircraft at same altitude level)

$$\frac{(v_i^* + \nu_i)\sin(m_i^* \qquad) - (v_j^* + \nu_j)\sin(m_j^* \qquad)}{(v_i^* + \nu_i)\cos(m_i^* \qquad) - (v_j^* + \nu_j)\cos(m_j^* \qquad)} \geqslant \tan(l_{ij})$$

or

$$\frac{(v_i^* + \nu_i)\sin(m_i^* \qquad) - (v_j^* + \nu_j)\sin(m_j^* \qquad)}{(v_i^* + \nu_i)\cos(m_i^* \qquad) - (v_j^* + \nu_j)\cos(m_j^* \qquad)} \leqslant \tan(g_{ij})$$

No conflict Constraints

No conflict constraint (aircraft at same altitude level)

$$\frac{(v_i^* \qquad) \sin(m_i^* + \mu_i) - (v_j^* \qquad) \sin(m_j^* + \mu_j)}{(v_i^* \qquad) \cos(m_i^* + \mu_i) - (v_j^* \qquad) \cos(m_j^* + \mu_j)} \geqslant \tan(l_{ij})$$

or

$$\frac{(v_i^* \qquad) \sin(m_i^* + \mu_i) - (v_j^* \qquad) \sin(m_j^* + \mu_j)}{(v_i^* \qquad) \cos(m_i^* + \mu_i) - (v_j^* \qquad) \cos(m_j^* + \mu_j)} \leqslant \tan(g_{ij})$$

No conflict Constraints

No conflict constraint (aircraft at same altitude level)

$$\frac{(v_i^* + \nu_i)\sin(m_i^* + \mu_i) - (v_j^* + \nu_j)\sin(m_j^* + \mu_j)}{(v_i^* + \nu_i)\cos(m_i^* + \mu_i) - (v_j^* + \nu_j)\cos(m_j^* + \mu_j)} \geqslant \tan(l_{ij})$$

or

$$\frac{(v_i^* + \nu_i)\sin(m_i^* + \mu_i) - (v_j^* + \nu_j)\sin(m_j^* + \mu_j)}{(v_i^* + \nu_i)\cos(m_i^* + \mu_i) - (v_j^* + \nu_j)\cos(m_j^* + \mu_j)} \leqslant \tan(g_{ij})$$

Unconstrained problem: Penalty cost function I

The infeasibility condition for a pair of aircraft flying at the same altitude level when there is no null denominator is:

$$\tan(g_{ij}) \leqslant \frac{(v_i+\nu_i)\sin(m_i+\mu_i)-(v_j+\nu_j)\sin(m_j+\mu_j)}{(v_i+\nu_i)\cos(m_i+\mu_i)-(v_j+\nu_j)\cos(m_j+\mu_j)} \leqslant \tan(l_{ij})$$

Unconstrained problem: Penalty cost function I

The infeasibility condition for a pair of aircraft flying at the same altitude level when there is no null denominator is:

$$\tan(g_{ij}) \leqslant \frac{(v_i + \nu_i)\sin(m_i + \mu_i) - (v_j + \nu_j)\sin(m_j + \mu_j)}{(v_i + \nu_i)\cos(m_i + \mu_i) - (v_j + \nu_j)\cos(m_j + \mu_j)} \leqslant \tan(l_{ij})$$

whereas when there is a null denominator is:

$$-\cot(g_{ij}) \leqslant \frac{(v_i + \nu_i)\sin(m_i + \mu_i + \pi/2) - (v_j + \nu_j)\sin(m_j + \mu_j + \pi/2)}{(v_i + \nu_i)\cos(m_i + \mu_i + \pi/2) - (v_j + \nu_j)\cos(m_j + \mu_j + \pi/2)} \leqslant -\cot(l_{ij})$$

The infeasibility condition for a pair of aircraft flying at the same altitude level when there is no null denominator is:

$$\tan(g_{ij}) \leqslant \frac{(v_i+\nu_i)\sin(m_i+\mu_i)-(v_j+\nu_j)\sin(m_j+\mu_j)}{(v_i+\nu_i)\cos(m_i+\mu_i)-(v_j+\nu_j)\cos(m_j+\mu_j)} \leqslant \tan(l_{ij})$$

whereas when there is a null denominator is:

$$-\cot(g_{ij}) \leqslant \frac{(v_i+\nu_i)\sin(m_i+\mu_i+\pi/2)-(v_j+\nu_j)\sin(m_j+\mu_j+\pi/2)}{(v_i+\nu_i)\cos(m_i+\mu_i+\pi/2)-(v_j+\nu_j)\cos(m_j+\mu_j+\pi/2)} \leqslant -\cot(l_{ij})$$

And, the objective functions are:

$$\min \sum_{f\in\mathcal{F}} |\nu_f| \quad \min \sum_{f\in\mathcal{F}} |\mu_f| \quad \min \sum_{f\in\mathcal{F}} |\gamma_f|$$

Unconstrained problem: Penalty cost function II

So, the penalty cost function is composed of the objective function and the following one (infeasibility condition):

$$g(\nu,\mu,\gamma) = \begin{cases} \sum_{i<j\in\mathcal{F}} \max\left\{0, \min\{\tan(l_{ij}) - t_{ij}, t_{ij} - \tan(g_{ij})\}\right\} & \text{if } cp_{ij} = 0 \text{ and} \\ & z_i + \gamma_i = z_j + \gamma_j \\ \sum_{i<j\in\mathcal{F}} \max\left\{0, \min\{-\cot(l_{ij}) - t'_{ij}, t'_{ij} + \cot(g_{ij})\}\right\} & \text{if } cp_{ij} = 1 \text{ and} \\ & z_i + \gamma_i = z_j + \gamma_j \end{cases}$$

Unconstrained problem: Penalty cost function II

So, the penalty cost function is composed of the objective function and the following one (infeasibility condition):

$$g(\nu,\mu,\gamma) = \begin{cases} \sum_{i<j\in\mathcal{F}} \max\left\{0, \min\{\tan(l_{ij}) - t_{ij}, t_{ij} - \tan(g_{ij})\}\right\} & \text{if } cp_{ij} = 0 \text{ and} \\ & z_i + \gamma_i = z_j + \gamma_j \\ \sum_{i<j\in\mathcal{F}} \max\left\{0, \min\{-\cot(l_{ij}) - t'_{ij}, t'_{ij} + \cot(g_{ij})\}\right\} & \text{if } cp_{ij} = 1 \text{ and} \\ & z_i + \gamma_i = z_j + \gamma_j \end{cases}$$

where:

$$t_{ij} = \frac{(v_i + \nu_i)\sin(m_i + \mu_i) - (v_j + \nu_j)\sin(m_j + \mu_j)}{(v_i + \nu_i)\cos(m_i + \mu_i) - (v_j + \nu_j)\cos(m_j + \mu_j)}$$

$$t'_{ij} = \frac{(v_i + \nu_i)\sin(m_i + \mu_i + \pi/2) - (v_j + \nu_j)\sin(m_j + \mu_j + \pi/2)}{(v_i + \nu_i)\cos(m_i + \mu_i + \pi/2) - (v_j + \nu_j)\cos(m_j + \mu_j + \pi/2)}$$

Unconstrained problem: Penalty cost function II

So, the penalty cost function is composed of the objective function and the following one (infeasibility condition):

$$g(\nu,\mu,\gamma) = \begin{cases} \sum_{i<j\in\mathcal{F}} \max\left\{0, \min\{\tan(l_{ij}) - t_{ij}, t_{ij} - \tan(g_{ij})\}\right\} & \text{if } cp_{ij} = 0 \text{ and} \\ & z_i + \gamma_i = z_j + \gamma_j \\ \sum_{i<j\in\mathcal{F}} \max\left\{0, \min\{-\cot(l_{ij}) - t'_{ij}, t'_{ij} + \cot(g_{ij})\}\right\} & \text{if } cp_{ij} = 1 \text{ and} \\ & z_i + \gamma_i = z_j + \gamma_j \end{cases}$$

where:

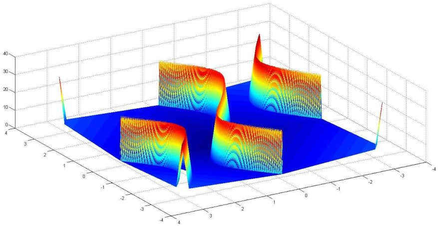
$$t_{ij} = \frac{(v_i+\nu_i)\sin(m_i+\mu_i) - (v_j+\nu_j)\sin(m_j+\mu_j)}{(v_i+\nu_i)\cos(m_i+\mu_i) - (v_j+\nu_j)\cos(m_j+\mu_j)}$$

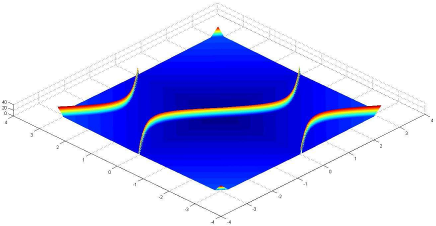
$$t'_{ij} = \frac{(v_i+\nu_i)\sin(m_i+\mu_i+\pi/2) - (v_j+\nu_j)\sin(m_j+\mu_j+\pi/2)}{(v_i+\nu_i)\cos(m_i+\mu_i+\pi/2) - (v_j+\nu_j)\cos(m_j+\mu_j+\pi/2)}$$

Penalty cost function

$$f(\nu,\mu,\gamma) = \min\{M g(\nu,\mu,\gamma) + \text{objective function}\}$$

Penalty cost function





Data structure

The following data structure is used in our VNS approach:

1. Aircraft configuration matrix (for each aircraft: velocity, heading angle, altitude level, abscissa, ordinate, maximum and minimum maneuvers allowed).
2. Solution matrix (for each aircraft: velocity, heading angle and altitude level).
3. Auxiliar matrices:
   - Parameter cp_{ij}.
   - TL, TG, CTL and CTG.
   - A (penalty function).

Algorithm to update matrix A

Algorithm 1: Updating matrix A

Function `Updating`($j, A, \nu, \mu, \gamma, V, T, Z, CP, TL, TG, CTL, CTG$);
$i = 0$;
repeat
    if $CP(i,j) = 0$ *and* $Z_i + \gamma_i = Z_j + \gamma_j$ **then**
        Use T, TL and TG to calculate A_{ij} by using g;
    else
        if $CP(i,j) = 1$ *and* $Z_i + \gamma_i = Z_j + \gamma_j$ **then**
            Use $T + \pi/2$, CTL and CTG to calculate A_{ij} by using g;
        else
            $A_{ij} = 0$;
        end
    end
    $i = i + 1$;
until $i = n$;

Local Search I

The main features of the local search (**intensification** phase) are:

- It is based on first improvement instead of best improvement to obtain a feasible solution as soon as possible. We have tried with best improvement but the solution quality was not different.
- Each aircraft changes its angle of motion ang and $-ang$ radians; its velocity vel and $-vel$ nm/h; its altitude level alt and $-alt$ until no solution improvement.
- When a solution is improved, the parameters of the problem must be updated.

Algorithm 2: First improvement local search for the CDR problem

```
Function FirstImprovement(ν, μ, γ, vel, ang, alt,
A, V, T, Z, CP, TL, TG, CTL, CTG);
k = 1 ;
repeat
    j = 1 ;
    repeat
        Move aircraft j by vel nm/h if k = 1;
        Move aircraft j by ang rads. if k = 2;
        Move aircraft j by alt levels if k = 3;
        if f(x) < f(x') then
            j = 0; x' = x;
        else
            Move aircraft j by −vel nm/h if k = 1;
            Move aircraft j by −ang rads. if k = 2;
            Move aircraft j by −alt levels if k = 3;
            if f(x) < f(x') then
                j = 0; x' = x;
            else
                j = j + 1;
            end
        end
    until j > n;
until k > 3;
Updating(j, A, ν, μ, γ, V, T, Z, CP, TL, TG, CTL, CTG)
```

Shaking I

The shaking procedure (**diversification** phase) consists of

- Parameter k determines both, number of aircraft to consider and corresponding maneuver to modify.
- $n/4$ aircraft are candidates to change any maneuver.
- Randomly, the sign of the corresponding maneuver is chosen.

Shaking II

Algorithm 3: Shaking for the CDR problem

```
Function Shaking(ν, μ, γ, vel, ang, alt, A, V, M, CP,
TL, TG, CTL, CTG);
nn ← k mod n/4 ;
u1 = Rand(0,1); u2 = Rand(0,1) ;
u3 = ⌈3 · Rand(0,1)⌉ ;
vel ← u1 k, ang ← u1 k, alt ← ⌈u1 k⌉ ;
j = 0;
repeat
    if u2 < 0.5 then
        Move aircraft j by vel nm/h if u3 = 1;
        Move aircraft j by ang rads. if u3 = 2;
        Move aircraft j by alt levels if u3 = 3;
    else
        Move aircraft j by −vel nm/h if u3 = 1;
        Move aircraft j by −ang rads. if u3 = 2;
        Move aircraft j by −alt levels if u3 = 3;
    end
    Updating(j, A, ν, μ, γ, V, T, Z, CP, TL, TG, CTL,
    CTG);
    j ← j + 1;
until j = nn;
```

Basic VNS algorithm

Algorithm 4: Steps of the VNS for the CDR problem

```
Function VNS (x, k_max, t_r, t_max);
Calculate CP, TL, TG, CTL, CTG, A ;
FirstImprovement(ν, μ, γ, vel, ang, alt, A, V, T, Z, CP,
TL, TG, CTL, CTG) ;
repeat
    k ← 1;
    repeat
        x' ← Shake(x, k)          /* Shaking */;
        x'' ← FirstImprovement(ν', μ', γ', vel, ang,
        alt, A, V, T, Z, CP, TL, TG, CTL, CTG);
        if f(x'') < f(x) then
            x ← x'; k ← 1    /* Make a move */;
            t_li ← CpuTime() ;
        else
            k ← k + 1     /* Next neighborhood */;
        end
        t ← CpuTime();
    until k = k_max;
    if t − t_li > t_r then
        break;
    end
until t > t_max;
```

Outline

Multi-objective approaches used

- Lexicographic Goal Programming.
- Compromise Programming.
- Compromise Programming combining l_1 and l_∞ distances.

Multi-objective approaches used

- Lexicographic Goal Programming.
- Compromise Programming.
- Compromise Programming combining l_1 and l_∞ distances.

All of them need the *pay-off matrix*.

Pay-off matrix

Ideal value

An **ideal** value for a single objective function is the best possible value when that objective is optimized subject to the corresponding set of constraints.

Pay-off matrix

Ideal value

An **ideal** value for a single objective function is the best possible value when that objective is optimized subject to the corresponding set of constraints.

Non-ideal value

A **nonideal** value is the worst value for a single objective function when optimizing another objective.

- It is a $n \times n$ where n is the number of objective functions.
- In the diagonal the **ideal** values are presented.
- In the rest of positions the **non-ideal** values are presented.

Lexicographic Goal Programming

- Introduced by Chames et al. (1955).
- GP sequentially solves optimization problems for each objective function.
- There is a priority order.

Our priority order is to minimize altitude level, heading angle and velocity changes to meet comfort [1].

1. Optimizing AC.

[1] Following the guidelines in Cetek (2009), Realistic speed change maneuvers for air traffic conflict avoidance and their impact on aircraft economics.

Our priority order is to minimize altitude level, heading angle and velocity changes to meet comfort [1].

1. Optimizing AC.
2. Optimizing TC together with the following additional constraint,

$$\sum_{f \in \mathcal{F}} c_f^{\gamma} \gamma_f \leqslant z_a^* + \lceil \varepsilon (z_a^{**} - z_a^*) \rceil$$

[1] Following the guidelines in Cetek (2009), Realistic speed change maneuvers for air traffic conflict avoidance and their impact on aircraft economics.

Lexicographic Goal Programming for VTAC

Our priority order is to minimize altitude level, heading angle and velocity changes to meet comfort [1].

1. Optimizing AC.
2. Optimizing TC together with the following additional constraint,

$$\sum_{f \in \mathcal{F}} c_f^{\gamma} \gamma_f \leqslant z_a^* + \lceil \varepsilon (z_a^{**} - z_a^*) \rceil$$

3. Optimizing VC together with the previous additional constraint and the following one,

$$\sum_{f \in \mathcal{F}} c_f^{\mu} |\mu_f| \leqslant z_t^* + \varepsilon (z_t^{**} - z_t^*)$$

[1] Following the guidelines in Cetek (2009), Realistic speed change maneuvers for air traffic conflict avoidance and their impact on aircraft economics.

Compromise Programming

- Introduced by Cochrane and Zeleny (1973).
- The decision maker prefers a solution as much closer as possible to the ideal value.
- A distance is minimized in the objective function.

1. Solves the VTAC model with the following objective function (minimizing the l_1 distance):

$$\min \rho_v \frac{\sum_{f \in \mathcal{F}} c_f^{\nu} |\nu_f| - z_v^*}{z_v^{**} - z_v^*} + \rho_t \frac{\sum_{f \in \mathcal{F}} c_f^{\mu} |\mu_f| - z_t^*}{z_t^{**} - z_t^*} + \rho_a \frac{\sum_{f \in \mathcal{F}} c_f^{\gamma} \gamma - z_a^*}{z_a^{**} - z_a^*}$$

Double Compromise Programming

- Introduced by Escudero (1995).
- It tries to bound the maximum deviation with respect to the ideal value.
- It consists on two steps.
  1. Minimizing l_∞ distance.
  2. Minimizing l_1 distance avoiding higher deviations than the obtained in the previous step.

1. Solves the VTAC model with the following objective function (minimizing the l_∞ distance):

$$\min \max \left\{ \frac{\sum_{f \in \mathcal{F}} c_f^\nu |\nu_f| - z_v^*}{z_v^{**} - z_v^*}, \frac{\sum_{f \in \mathcal{F}} c_f^\mu |\mu_f| - z_t^*}{z_t^{**} - z_t^*}, \frac{\sum_{f \in \mathcal{F}} c_f^\gamma |\gamma_f| - z_a^*}{z_a^{**} - z_a^*} \right\}$$

1. Solves the VTAC model with the following objective function (minimizing the l_∞ distance):

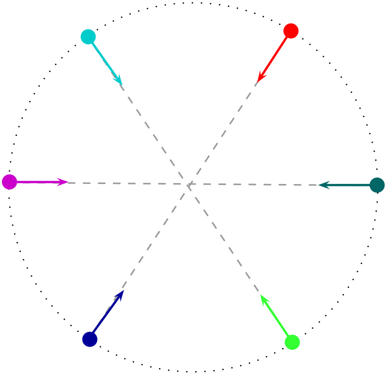
$$\min \max \left\{ \frac{\sum_{f\in\mathcal{F}} c_f^{\nu}|\nu_f| - z_v^*}{z_v^{**} - z_v^*}, \frac{\sum_{f\in\mathcal{F}} c_f^{\mu}|\mu_f| - z_t^*}{z_t^{**} - z_t^*}, \frac{\sum_{f\in\mathcal{F}} c_f^{\gamma}|\gamma_f| - z_a^*}{z_a^{**} - z_a^*} \right\}$$

2. Solves the VTAC model with the following objective function (minimizing the l_1 distance) avoiding deviations higher than the one obtained in the previous step:

$$\min \rho_v \frac{\sum_{f\in\mathcal{F}} c_f^{\nu}|\nu_f| - z_v^*}{z_v^{**} - z_v^*} + \rho_t \frac{\sum_{f\in\mathcal{F}} c_f^{\mu}|\mu_f| - z_t^*}{z_t^{**} - z_t^*} + \rho_a \frac{\sum_{f\in\mathcal{F}} c_f^{\gamma}\gamma - z_a^*}{z_a^{**} - z_a^*}$$

Outline

Initial situation, *roundabout* case



Preliminary computational results

Instance		Minotaur						Gap VNS					
Case	**nc**	$\mathbf{z}_v^*$	$\mathbf{z}_t^*$	$\mathbf{z}_a^*$	$\mathbf{z}_v^{**}$	$\mathbf{z}_t^{**}$	$\mathbf{z}_a^{**}$	$\mathbf{g}_v^*$	$\mathbf{g}_t^*$	$\mathbf{g}_a^*$	$\mathbf{g}_v^{**}$	$\mathbf{g}_t^{**}$	$\mathbf{g}_a^{**}$
C2-4	1	0.0000	0.0048	0	9.0000	0.0050	1	0.00	-4.17	0.00	-0.89	38.00	0.00
C3-4	3	0.0000	0.0000	0	2.0010	0.0144	3	0.00	0.00	0.00	4.33	21.53	0.00
C4-4	6	0.0000	0.0096	0	17.3330	0.0191	6	0.00	23.96	0.00	-82.82	5.76	0.00

Table: Ideal and non-ideal values

Instance	Minotaur						VNS					
Case	$\mathbf{t}_v^*$	$\mathbf{t}_t^*$	$\mathbf{t}_a^*$	$\mathbf{t}_v^{**}$	$\mathbf{t}_t^{**}$	$\mathbf{t}_a^{**}$	$\mathbf{t}_v^*$	$\mathbf{t}_t^*$	$\mathbf{t}_a^*$	$\mathbf{t}_v^{**}$	$\mathbf{t}_t^{**}$	$\mathbf{t}_a^{**}$
C2-4	0.56	0.08	0.02	0.03	0.03	0.03	0.00	0.00	0.00	1.12	0.00	0.00
C3-4	0.17	0.13	0.04	0.09	0.08	0.46	0.00	0.01	0.00	0.04	0.00	1.05
C4-4	71.27	1.09	0.10	1.20	0.21	4.78	0.00	2.12	0.00	1.62	0.00	1.30

Table: Computing times (in seconds)

4xIntel Core i5-2430M, 2.40 GHz, 8Gb RAM, Xubuntu 14.04 OS

Gaps obtained as: $\frac{z_{VNS} - z_M}{z_M} \cdot 100\%$

Outline

Conclusions:

- A Variable Neighborhood Search algorithm has been presented.
- Conflict situations are solved by performing the three maneuvers.
- The local search and shaking phases are based on angle and velocity discretization (altitude level changes are discrete).
- A multi-objective framework is presented to provide different methods that could be applied to choose solutions when economic or comfort terms are preferred.

Conclusions and future research

Conclusions:

- A Variable Neighborhood Search algorithm has been presented.
- Conflict situations are solved by performing the three maneuvers.
- The local search and shaking phases are based on angle and velocity discretization (altitude level changes are discrete).
- A multi-objective framework is presented to provide different methods that could be applied to choose solutions when economic or comfort terms are preferred.

Future research:

- Integrate VNS and Multi-objective.
- Validation of the VNS approach in an extensive computational experiment.
- Refine the local search as well as the shaking phase of the algorithm.

Thanks a lot for your attention!

7th International Conference on Research in Air Transportation

A Variable Neighborhood Search approach for the aircraft conflict resolution problem

F. Javier Martín Campo
javier.martin.campo@mat.ucm.es

Joint work with:

A. Alonso-Ayuso, L.F. Escudero, N. Mladenović