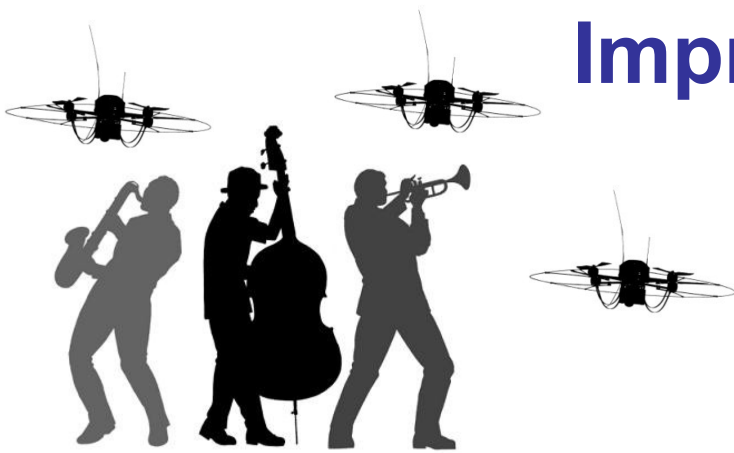
Optimization of Aerial Surveys using an Algorithm Inspired in Musicians Improvisation



João Valente
joao.valente@upm.es

23rd International Conference on Automated Planning and Scheduling
Rome, Italy 10-14 June 2013

1st Workshop on Planning and Robotics (PlanRob) - 10/06/2013

INDUSTRIALES
ETSII | UPM

Index

POLITÉCNICA

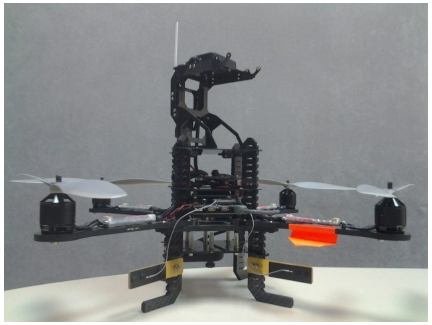
Introduction

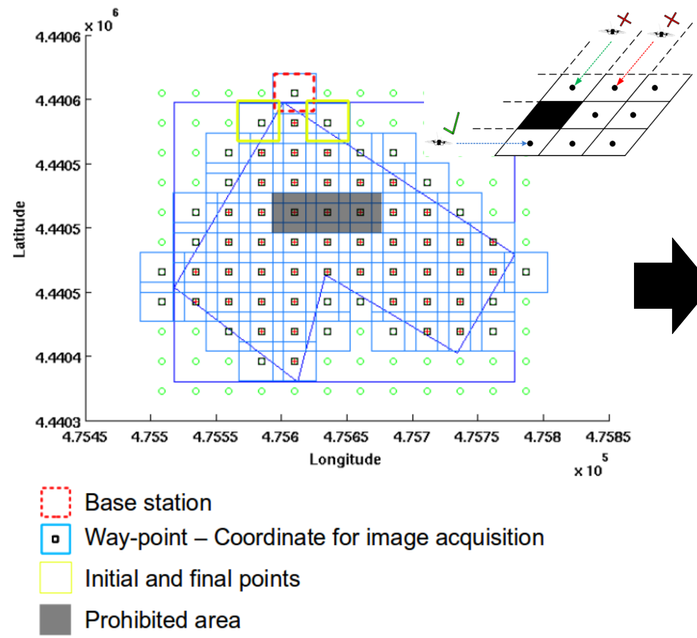
- **Goal:**
  - Compute trajectories for a fleet of mini aerial vehicles shipped with a digital camera subject to a set of restrictions
  - Mosaicking

- **Applications**
  - Monitoring and inspections of Critical infrastructures
  - Precision agriculture

- **Projects:**
  - ROTOS (Multi-Robot System for Large Outdoor Infrastructures Protection. DPI 2010-17998)
  - RHEA (Robot Fleets for Highly Effective Agriculture and Forestry Management. NMP-CP-IP 245986-2)

Problematic



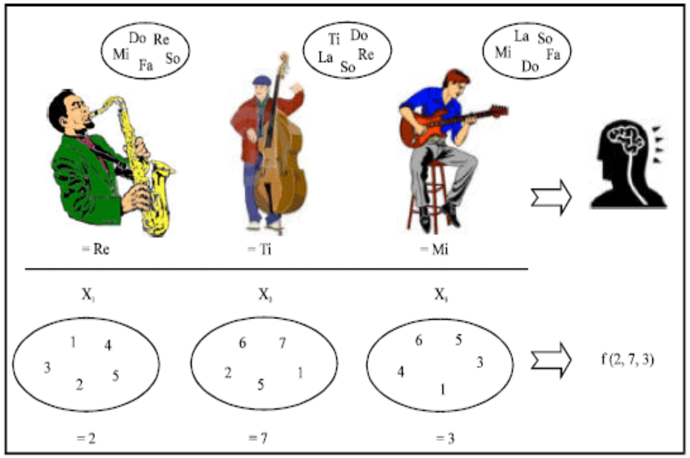


Full coverage trajectories

?

Harmony Search algorithm (I)

- Basic concepts
  - Soft computing, Meta-heuristic approach
  - Inspired by the improvisation process of musicians



[Lee, K. and Z. Geem, 2005]

- Methodology
  - Step 1: Initialization of the optimization problem
  - Step 2: Initialization of the harmony memory (HM)
  - Step 3: Improvisation a New Harmony from the HM set
  - Step 4: Updating HM
  - Step 5: Repeat steps 3 and 4 until the end criterion is satisfied

Lee, K. and Z. Geem, 2005. A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. Comput. Methods Applied Mechanics Eng., 194: 3902-3933.

INDUSTRIALES
ETSII | UPM

Harmony Search algorithm (II)

- **Step 1**: Initialization of the optimization problem

Minimize F(x) subject to $x_i \in X_i$, i = 1,2,...N

Where:

F(x) : Objective function

x : Set of each design variable (xi)

Xi : Set of the possible range of values for each design variable ($a < X_i < b$)

N : Number of design variables

POLITÉCNICA

Harmony Search algorithm (III)

- **Step 2**: Initialization of the harmony memory (HM)
  - Generate random vectors
  - HMS: Harmony Memory Size

$$HM = \begin{bmatrix} X_1^1 & \cdots & X_N^1 & J(X^1) \\ \vdots & \ddots & \vdots & \vdots \\ X_1^{\{HMS\}} & \cdots & X_N^{\{HMS\}} & J(X^{\{HMS\}}) \end{bmatrix}$$

Harmony Search algorithm (IV)

INDUSTRIALES
ETSII | UPM

POLITÉCNICA

- **Step 3**: Improvisation a New Harmony from the HM set
  - New harmony vector, $x' = (x_1', x_2', \ldots, x_n')$
  - Three rules:
    - Random selection
    - Memory consideration
      - HMCR: Harmony Memory Considering Rate

$$x'_i \leftarrow \begin{cases} x_i \in \{x_i^1, x_i^2, \ldots, x_i^{HMS}\}, & w.p\ HMCR \\ x_i \in X_i, & w.p\ 1 - HMCR \end{cases}$$

    - Pitch adjustment
      - PAR: Pitch Adjusting Rate

$$x'_i \leftarrow \begin{cases} x'_i \pm 1, & w.p\ PAR \\ x'_i, & w.p\ 1 - PAR \end{cases}$$

Harmony Search algorithm (V)

- **Step 4**: Updating HM
  - F(X') < F(X) ?

- **Step 5**: Repeat steps 3 and 4 until the end criterion is satisfied
  - Stop criterion, Number of improvisations (NI)

The m-CPP algorithm (I)

- **Step 1:** Initialization of the optimization problem
  - Employ HS algorithm to find the optimal coverage safe path
  - Minimize $J = J_1 + J_2$
    - *Subject to*
      - x_1 and x_i ,i = 1,...,N

$$J_1 = K_1 \times \sum_{i=1}^{m} \psi_k^{\{i\}} + K_2, \quad k \in \{135°, 90°, 45°, 0°\}$$

$$K_2 > K_1, \quad K_{1,2} \in \mathbb{R}$$

$$\psi_{\pm 135°} > \psi_{\pm 90°} > \psi_{\pm 45°} > \psi_{\pm 0°}$$

$$J_2 = J_2' \times K_3, \qquad K_3 >> K_1, K_2, \qquad K_3 \in \mathbb{R}$$

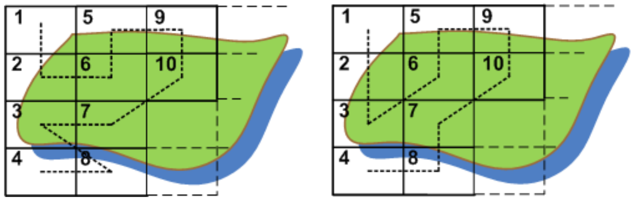
$$J_2' = \mathcal{S}_1 \vee \mathcal{S}_2 \ldots \mathcal{S}_{n-1} \vee \mathcal{S}_n = \vee_{i=1}^{n} \mathcal{S}_i$$

  - Decision variables

$X^{\{j\}} = [x_1, x_2, x_3, \ldots, x_{i-2}, x_{i-1}, x_i]$,

i=1,...,N;

j=1,...,HMS



$X^{\{1\}} = [1, 2, 6, 5, 9, 10, 7, 3, 8, 4]$
$X^{\{2\}} = [1, 2, 3, 6, 5, 9, 10, 7, 8, 4]$

The m-CPP algorithm (II)

- **Step 2**: Initialization of the harmony memory (HM)
  - Generate candidate permutations
  - Random Breath Coverage algorithm
  - Numerical example: $X^{\{1\}} = [1,2,3,6,9,8,7,4,1]$

1	4	7
2	X	8
3	6	9

The m-CPP algorithm (III)

- **Step 3**: Improvisation a New Harmony from the HM set
  - Random selection
  - Memory consideration
    - HMCR: Harmony Memory Considering Rate

$$X'_i \leftarrow \begin{cases} X'_i \in \begin{cases} S_i \in X_i & \exists s \in X_i \\ S_i \in X & \nexists s \in X_i \end{cases}, & w.p\ HMCR \\ X'_i \in S_i, & w.p\ 1 - HMCR \end{cases}$$

$$S = \bigcup_{s \in S} s$$

  - Pitch adjustment
    - PAR: Pitch Adjusting Rate

$$X''_i \leftarrow \begin{cases} X'_i \pm 1, & w.p\ PAR \\ X'_i, & w.p\ 1 - PAR \end{cases}$$

The m-CPP algorithm (IV)

- **Step 4**: Updating HM
  - $J(X') < J(X)$?

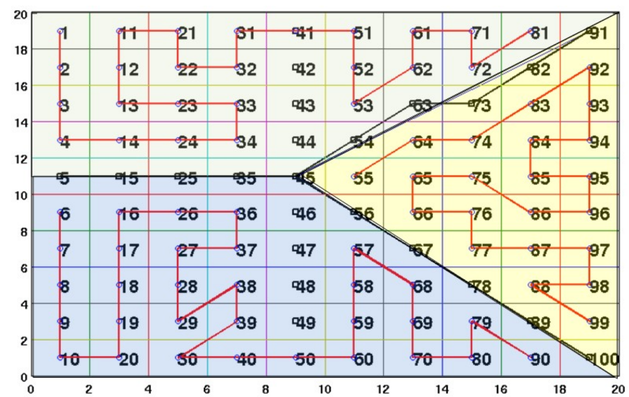
- **Step 5**: Repeat steps 3 and 4 until the end criterion is satisfied
  - Stop criterion
    - Number of improvisations
    - An admissible number of turns (a hypothesis)

INDUSTRIALES
ETSII | UPM

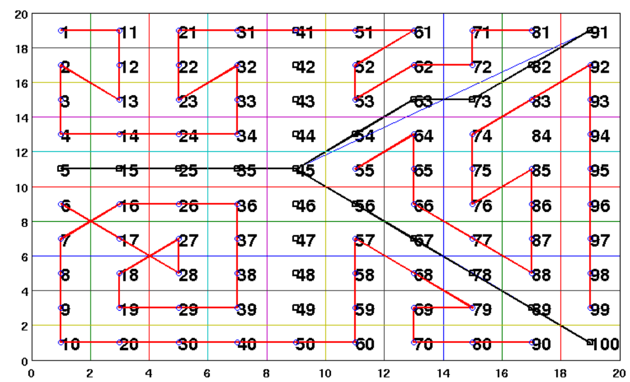
Results achieved (I)

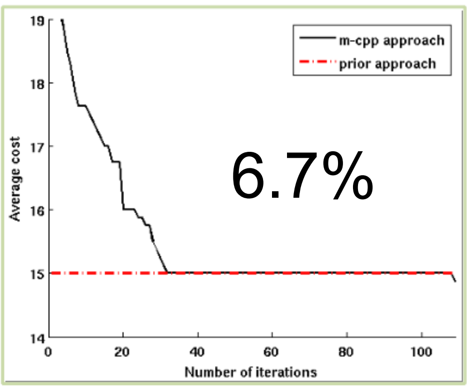
POLITÉCNICA

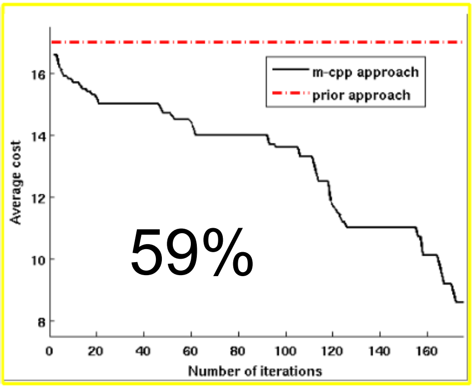
Heuristic approach [7]

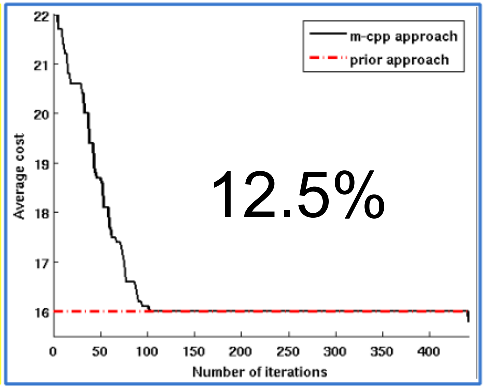


m-CPP approach



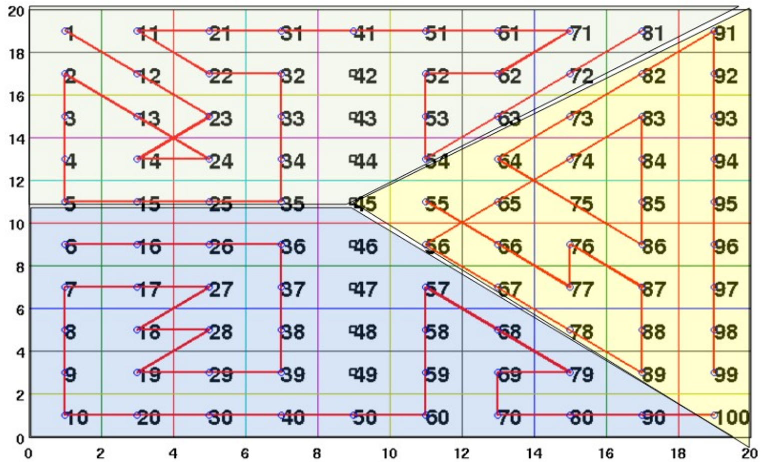






Results achieved (II)

- Removing borders [9]
  - Computing time
    - max 2 minutes per area
  - Area coverage
    - Improved
  - Cost
    - Improved for two
    - Worsened for one



Conclusions

- A novel approach to ACPP employing HS algorithm
  - Improved previous approach
  - Improved airspace safety
  - Improved area coverage

- Computation time an issue
  - Large workspaces
  - Divide to conquer
  - Real time computing

INDUSTRIALES
ETSII | UPM

POLITÉCNICA

Grazie mille!