Recent Progress in Information Gathering and Surveillance Missions Planning with Unmanned Aerial Vehicles

Jan Faigl

Associate Professor – Department of Computer Science
Head of Computational Robotics Laboratory
Artificial Intelligence Center (AIC)
Center for Robotics and Autonomous Systems (CRAS)
Faculty of Electrical Engineering
Czech Technical University in Prague

July 20, 2018
Established in 18 January, 1707

*Foundation deed signed by Emperor Joseph I*

About 21 000 students enrolled and 1 9000 academic employees – 8 faculties

**Faculty of Electrical Engineering (FEE)**

**Department of Computer Science**

- First CS department in Czechia established in 1964
  - [http://cs.felk.cvut.cz](http://cs.felk.cvut.cz)

**Artificial Intelligence Center (AIC)**

- Research in AI for more than 20 years [http://aic.fel.cvut.cz](http://aic.fel.cvut.cz)

**Computational Robotics Laboratory (ComRob)**

- [https://comrob.fel.cvut.cz](https://comrob.fel.cvut.cz) — established in 2013

  - Focused on robotic information gathering — a problem to create a model of phenomena by autonomous mobile robots performing measurements in dynamic unknown environment.
  - Mostly *aerial* and *ground* (multi-legged) robotic vehicles
Surveillance planning in Mohamed Bin Zayed International Robotic Challenge (MBZIRC) 2017
Provide *curvature-constrained* path to collect the most valuable measurements with shortest possible path/time or under limited travel budget.

Can be formulated as routing problems with Dubins vehicle:
- Dubins Traveling Salesman Problem with Neighborhoods
- Dubins Orienteering Problem with Neighborhoods
Planning Curvature-Constrained Multi-Goal Path Dubins Vehicle for Fixed-Wing and Multi-Rotor Vehicles

- Sharp turns can lead to high error of visiting the requested goals
- Planned paths should support precise trajectory following by the used controller
- **Dubins vehicle** can be used for curvature-constrained paths

- Minimal turning radius $\rho$ and constant forward velocity $v$ with the state $q = (x, y, \theta)$, $q \in SE(2)$, $(x, y) \in \mathbb{R}^2$ and $\theta \in S^1$
- **Optimal path connecting** $q_1, q_2 \in SE(2)$ can be found analytically
- Two types of maneuvers: CSC and CCC

(Dubins, 1957)

The main difficulty is to determine the vehicle headings for a given set/sequence of waypoints

Jan Faigl – Surveillance Planning with UAVs
CACRE 2018, Chengdu, China
Dubins Traveling Salesman Problem (DTSP)

- Having a set of locations to be visited, the problem is to determine a closed shortest Dubins path visiting each location $p_i \in P$ of the given set of $n$ locations $P = \{p_1, \ldots, p_n\}$, $p_i \in \mathbb{R}^2$

1. Permutation $\Sigma = (\sigma_1, \ldots, \sigma_n)$ of visits
   - Sequencing part of the problem – combinatorial optimization

2. Headings $\Theta = \{\theta_{\sigma_1}, \theta_{\sigma_2}, \ldots, \theta_{\sigma_n}\}$ for $p_{\sigma_i} \in P$
   - Continuous optimization

- DTSP is an optimization problem over all possible permutations $\Sigma$ and headings $\Theta$ in the states $(q_{\sigma_1}, q_{\sigma_2}, \ldots, q_{\sigma_n})$ such that $q_{\sigma_i} = (p_{\sigma_i}, \theta_{\sigma_i})$

\[
\text{minimize}_{\Sigma, \Theta} \sum_{i=1}^{n-1} \mathcal{L}(q_{\sigma_i}, q_{\sigma_{i+1}}) + \mathcal{L}(q_{\sigma_n}, q_{\sigma_1}) \quad (1)
\]

subject to $q_i = (p_i, \theta_i) \; i = 1, \ldots, n, \quad (2)$

where $\mathcal{L}(q_{\sigma_i}, q_{\sigma_j})$ is the length of Dubins path between $q_{\sigma_i}$ and $q_{\sigma_j}$. 
Exploiting non-zero sensing range $\delta$ to shorten the requested multi-goal path

Dubins Traveling Salesman Problem with Neighborhoods (DTSPN) – determine the sequence of visits $\Sigma$, headings $\Theta$, but also the waypoint locations within the respective neighborhoods $P = \{ p_1, \ldots, p_n \}, p_i \in \mathbb{R}^2$
Existing Approaches to the DTSP(N) Heuristics, Resolution Complete, and Sampling-based

- **Sampling-based approaches**
  - Obermeyer, 2009
  - Oberlin et al., 2010
  - Macharet et al., 2016

- **Convex optimization**
  - (Only if the locations are far enough)
  - Goac et al., 2013

- **Lower-bound for the DTSP**
  - Using Dubins Interval Problem (DIP)
  - Manyam et al., 2016

- **Lower-bound for the DTSPN**
  - Using Generalized DIP (GDIP)
  - Váňa and Faigl, 2018

- **Heuristic approaches**
  - Savla et al., 2005
  - Ma and Castanon, 2006
  - Macharet et al., 2011
  - Macharet et al., 2012
  - Ny et al., 2012
  - Yu and Hang, 2012
  - Macharet et al., 2013
  - Zhant et al., 2014
  - Macharet and Campost, 2014
  - Váňa and Faigl, 2015
  - Isaiah and Shima, 2015
  - ...

Jan Faigl – Surveillance Planning with UAVs

CACRE 2018, Chengdu, China
Determine the shortest Dubins maneuver connecting $p_i$ and $p_j$ given the angle intervals $\theta_i \in [\theta_i^{\min}, \theta_i^{\max}]$ and $\theta_j \in [\theta_j^{\min}, \theta_j^{\max}]$.

DIP has closed-form solution.

For the intervals $\Theta_i = \Theta_j = [0, 2\pi)$, the solution is the length of the straight line segment.

It provides lower-bound of the length of the shortest Dubins maneuver connecting $p_i$ and $p_j$. 

Manyam, Rathinam, and Casbeer, 2016
For a sequence of the waypoint locations

\[ P = (p_1, \ldots, p_n) \]

E.g., found as a solution of the Euclidean TSP

We can sample possible heading values at each location \( i \) into a discrete set of \( k \) headings, i.e., \( h_i = \{\theta_i^1, \ldots, \theta_i^k\} \) and create a graph of all possible Dubins maneuvers.

For a set of heading samples, the optimal solution can be found by a forward search of the graph in \( O(nk^3) \).

The key is to determine the most suitable heading samples per each waypoint.

The lower bound can be found using DIP.
Sampling-based Solution of the DTSP with a Given Sequence of Visits \( \Sigma \) – Dubins Touring Problem (DTP)

- For a sequence of the waypoint locations

\[ P = (p_1, \ldots, p_n) \]

*E.g., found as a solution of the Euclidean TSP*

- We can sample possible heading values at each location \( i \) into a discrete set of \( k \) headings, i.e., \( h_i = \{\theta^1_i, \ldots, \theta^k_i\} \) and create a graph of all possible Dubins maneuvers.

- For a set of heading samples, the optimal solution can be found by a forward search of the graph in \( O(nk^3) \)

- The key is to determine the most suitable heading samples per each waypoint

- The lower bound can be found using DIP
Sampling-based Solution of the DTSP (as the DTP) Uniform vs Informed Sampling of the Headings

Uniform sampling

$N = 224$, $T_{cpu} = 128$ ms
$\mathcal{L} = 19.8$, $\mathcal{L}_U = 13.8$,

Lower Bound Solution

$N = 128$, $T_{cpu} = 76$ ms
$\mathcal{L} = 14.4$, $\mathcal{L}_U = 14.2$,

- $N$ – the total number of samples (up to 32 samples per waypoint)
- $\mathcal{L}$ is the length of the tour (blue) and $\mathcal{L}_U$ is the lower bound (red) determined as a solution of the Dubins Interval Problem (DIP)

Solution of the DTSP with Given Sequence of Visits
Uniform vs Informed Sampling

- Refinement iteration 1, the angular resolution $2\pi/4$

**Uniform sampling**

- $\epsilon = 2\pi/4$, $N = 28$, $T_{CPU} = 8$ ms
- $\mathcal{L} = 27.9$, $\mathcal{L}_U = 13.2$

**Informed sampling**

- $\epsilon = 2\pi/4$, $N = 21$, $T_{CPU} = 8$ ms
- $\mathcal{L} = 29.9$, $\mathcal{L}_U = 13.2$
Refinement iteration 1, the angular resolution $2\pi/8$

**Uniform sampling**

$\epsilon = 2\pi/8, \ N = 56, \ T_{CPU} = 16 \text{ ms}$

$L = 20.8, \ L_U = 13.2$

**Informed sampling**

$\epsilon = 2\pi/8, \ N = 28, \ T_{CPU} = 20 \text{ ms}$

$L = 21.0, \ L_U = 13.2$
Solution of the DTSP with Given Sequence of Visits
Uniform vs Informed Sampling

- Refinement iteration 1, the angular resolution $2\pi/16$

**Uniform sampling**

- $\epsilon = 2\pi/16$, $N = 112$, $T_{CPU} = 40$ ms
- $\mathcal{L} = 20.3$, $\mathcal{L}_U = 13.5$

**Informed sampling**

- $\epsilon = 2\pi/16$, $N = 35$, $T_{CPU} = 24$ ms
- $\mathcal{L} = 20.1$, $\mathcal{L}_U = 13.5$
Refinement iteration 1, the angular resolution $2\pi/32$

**Uniform sampling**

- $\epsilon = 2\pi/32$, $N = 224$, $T_{CPU} = 140$ ms
- $\mathcal{L} = 19.8$, $\mathcal{L}_U = 13.8$

**Informed sampling**

- $\epsilon = 2\pi/32$, $N = 44$, $T_{CPU} = 32$ ms
- $\mathcal{L} = 19.9$, $\mathcal{L}_U = 13.8$
Solution of the DTSP with Given Sequence of Visits
Uniform vs Informed Sampling

- Refinement iteration 1, the angular resolution $2\pi/64$

**Uniform sampling**

$$\epsilon = 2\pi/64, \quad N = 448, \quad T_{CPU} = 456 \text{ ms}$$

$$\mathcal{L} = 14.5, \quad \mathcal{L}_U = 14.5$$

**Informed sampling**

$$\epsilon = 2\pi/64, \quad N = 51, \quad T_{CPU} = 48 \text{ ms}$$

$$\mathcal{L} = 19.9, \quad \mathcal{L}_U = 13.9$$
Refinement iteration 1, the angular resolution $2\pi/128$

**Uniform sampling**

\[ \epsilon = \frac{2\pi}{128}, \quad N = 896, \quad T_{CPU} = 1620 \text{ ms} \]
\[ \mathcal{L} = 14.5, \quad \mathcal{L}_U = 14.5 \]

**Informed sampling**

\[ \epsilon = \frac{2\pi}{128}, \quad N = 70, \quad T_{CPU} = 60 \text{ ms} \]
\[ \mathcal{L} = 14.8, \quad \mathcal{L}_U = 14.1 \]
Solution of the DTSP with Given Sequence of Visits
Uniform vs Informed Sampling

- Refinement iteration 1, the angular resolution $2\pi/256$

**Uniform sampling**

$\epsilon = 2\pi/256$, $N = 1792$, $T_{CPU} = 6784$ ms
$\mathcal{L} = 14.4$, $\mathcal{L}_U = 14.3$

**Informed sampling**

$\epsilon = 2\pi/256$, $N = 100$, $T_{CPU} = 88$ ms
$\mathcal{L} = 14.4$, $\mathcal{L}_U = 14.3$
Comparison with the Alternating Algorithm (AA), Local Iterative Optimization (LIO), and Memetic algorithm


A sequence of the waypoint locations is determined as the Euclidean TSP (ETSP)

E.g., as in the Alternating Algorithm (AA)

In Memetic algorithm, similarly to the sampling-based approaches that solve the Generalized TSP, the best sequence of visits is determined during the solution

- In the DTSPN, we need to determine not only the headings, but the waypoint locations themselves.
- Dubins Interval Problem is not sufficient to provide tight lower-bound.

- Generalized Dubins Interval Problem (GDIP) can be utilized for the DTSPN similarly as the DIP for the DTSP.
Generalized Dubins Interval Problem (GDIP)

- Determine the shortest Dubins maneuver connecting $P_i$ and $P_j$ given the angle intervals $\theta_i \in [\theta_i^{\min}, \theta_i^{\max}]$ and $\theta_j \in [\theta_j^{\min}, \theta_j^{\max}]$

**Full problem (GDIP)**

**One-side version (OS-GDIP)**

- Transformation from the GDIP to the OS-GDIP:
  - $P'_1 = \{p'_1\} = \{(0, 0)\}$
  - $P'_2 = P_2 \oplus \tilde{P}_1 = \bigcup \{p_b - p_a, p_a \in P_1, p_b \in P_2\}$
- A closed-form solution can be found for the OS-GDIP

Closed-form expressions (1-6)

1) S type

2) CS type

3) Cψ type

4) CSC type

5) CSCψ type

6) CCψ C type

Convex optimization (7)

7) CCψ type

Average computational time

<table>
<thead>
<tr>
<th>Problem</th>
<th>Time [µs]</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dubins maneuver</td>
<td>0.58</td>
<td>1.00</td>
</tr>
<tr>
<td>DIP</td>
<td>2.86</td>
<td>4.93</td>
</tr>
<tr>
<td>GDIP</td>
<td>12.63</td>
<td>21.78</td>
</tr>
</tbody>
</table>
Iterative refinement of the neighborhood samples and heading samples

Resolution: 4  
Gap: 69.3%  
Time: 0.079 s
Iterative refinement of the neighborhood samples and heading samples

Resolution: 8  
Gap: 39.4 %  
Time: 0.211 s
Iterative refinement of the neighborhood samples and heading samples

Resolution: 16  
Gap: 19.9 %  
Time: 0.552 s
Iterative refinement of the neighborhood samples and heading samples

Resolution: 32  
Gap: 10.7%  
Time: 1.292 s
Iterative refinement of the neighborhood samples and heading samples

- Resolution: 64
- Gap: 5.3%
- Time: 3.183 s
Iterative refinement of the neighborhood samples and heading samples

Resolution: 128  Gap: 2.6%  Time: 8.994 s
Iterative refinement of the neighborhood samples and heading samples

Resolution: 256  
Gap: 1.3 %  
Time: 33.474 s
For a given sequence of visits to the target regions (locations)

- The algorithm scales linearly with the number of locations
- Complexity of the algorithm is approximately $O(nk^{1.8})$

https://github.com/comrob/gdip

Motivation for Surveillance Planning with Multiple UAVs in MBZIRC 2017 Scenario

- Provide **curvature-constrained** paths for a team of autonomous unmanned aerial vehicles to verify expected objects of interest

\[ v = 5 \text{ m.s}^{-1}, \ 80 \text{ m} \times 60 \text{ m} \text{ testing site for experimental verification of our system for the Mohamed Bin Zayed International Robotics Challenge (MBZIRC)} \]

- Sampling-based methods are relatively slow
- Desired properties of the requested surveillance mission planner are: **fast trajectories and low computational time** (\( \leq 1 \text{ s} \))
Fast heuristic solution based on unsupervised learning for routing problems

Solutions found in less than 0.6 second for the MBZIRC 2017 scenarios

Comparison with Memetic algorithm (Zhang et al., 2014) restricted to the maximal computational time $T_{\text{max}} \in \{1, 5, 10, 60\}$ seconds and $k$ vehicles

<table>
<thead>
<tr>
<th>$k$</th>
<th>Memetic 1 s $L_{\text{max}}$ [m]</th>
<th>Memetic 10 s $L_{\text{max}}$ [m]</th>
<th>Unsupervised Learning $L_{\text{max}}$ [m]</th>
<th>$T$ [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>586.01 (24.22)</td>
<td>376.52 (27.17)</td>
<td>363.38 (36.56)</td>
<td>0.55 (0.07)</td>
</tr>
<tr>
<td>2</td>
<td>335.83 (10.67)</td>
<td>212.18 (18.73)</td>
<td>223.76 (40.76)</td>
<td>0.53 (0.01)</td>
</tr>
<tr>
<td>3</td>
<td>240.67 (6.63)</td>
<td>153.37 (12.79)</td>
<td>180.12 (29.49)</td>
<td>0.53 (0.03)</td>
</tr>
</tbody>
</table>


**Dubins Airplane** model describes the vehicle state $q = (p, \theta, \psi)$, $p \in \mathbb{R}^3$ and $\theta, \psi \in S^1$ as

Chitsaz, H., LaValle, S.M. (2017)

- Constant forward velocity $v$, the minimal turning radius $\rho$, and limited pitch angle, i.e., $\psi \in [\psi_{\text{min}}, \psi_{\text{max}}]$

- Parametrization of 3D regions to be visited

Solutions based on LIO (ETSP+LIO), TSP with the travel cost according to Dubins Airplane Model (DAM-TSP+LIO), and sampling-based approach with transformation of the GTSP to the ATSP solved by LKH.


---

**Algorithm 1:** LIO-based Solver for 3D-DTSPN

**Data:** Regions \( \mathcal{R} \)

**Result:** Solution represented by \( Q \) and \( \Sigma \)

1. \( \Sigma \leftarrow \text{getInitialSequence}(\mathcal{R}) \);
2. \( Q \leftarrow \text{getInitialSolution}(\mathcal{R}, \Sigma) \);
3. **while** terminal condition **do**
   4. \( Q \leftarrow \text{optimizeHeadings}(Q, \mathcal{R}, \Sigma) \);
   5. \( Q \leftarrow \text{optimizeAlpha}(Q, \mathcal{R}, \Sigma) \);
   6. \( Q \leftarrow \text{optimizeBeta}(Q, \mathcal{R}, \Sigma) \);
4. **end**
5. **return** \( Q, \Sigma \);
Surveillance Planning with Bézier Curves
DTSPN with Parametrization of 3D Smooth Trajectory

- Multi-rotor aerial vehicles can generally move in arbitrary direction
  - DTSPN variant for surveillance planning with 3D trajectory

- Find a 3D smooth trajectory visiting a given set of 3D regions
- Minimizes the **Travel Time Estimation** (TTE)
- Satisfies limited velocity and acceleration of the vehicle

![Graph showing trajectory](image)

---


Jan Faigl – Surveillance Planning with UAVs
CACRE 2018, Chengdu, China
Benefits of Bézier curves

- Flexible and easy to use
- Start/end direction is given by the first/last two control points

Example of a cubic Bézier curve

$$X(\tau) = B_0(1 - \tau)^3 + 3B_1\tau(1 - \tau)^2 + 3B_2\tau^2(1 - \tau) + B_3\tau^3$$

Visit the most important targets because of limited travel budget.

The problem can be formulated as the Dubins Orienteering Problem (DOP).

It can be solved using sampling-based methods, e.g., with Variable Neighborhood Search (VNS) combinatorial metaheuristic.


Similarly the Dubins Orienteering Problem with Neighborhoods (DOPN) can be formulated and solved.

We need to sample the waypoint locations and headings as in DTSPN.

Summary

- Surveillance planning with curvature-constrained trajectory
  - Dubins Traveling Salesman Problem (with Neighborhoods) – DTSPN
  - Informed sampling-based methods based on
  - Tight lower bound for the DTSPN based on the GDIP
  - 3D data collection planning with Dubins Airplane Model
  - Fast unsupervised learning based methods for DTSPN
  - Surveillance planning with Bézier curves
  - Dubins Orienteering Problem (with Neighborhoods)
The presented work are mostly results of my colleagues from the Computational Robotics Laboratory and Multi-Robot Systems Group.

Work with us within the Research Center for Informatics – http://rci.cvut.cz