Data Collection Planning with Dubins Airplane Model and Limited Travel Budget

Petr Váňa, Jan Faigl, Jakub Sláma, Robert Pěnička

***Abstract*— In this paper, we address the data collection planning problem for fixed-wing unmanned aircraft vehicle (UAV) with a limited travel budget. We formulate the problem as a variant of the Orienteering Problem (OP) in which the Dubins airplane model is utilized to extend the problem into the three-dimensional space and curvature-constrained vehicles. The proposed Dubins Airplane Orienteering Problem (DA-OP) stands to find the most rewarding data collection trajectory visiting a subset of the given target locations while the trajectory does not exceed the limited travel budget. Contrary to the original OP formulation, the proposed DA-OP combines not only the combinatorial part of determining a subset of the targets to be visited together with determining the sequence to visited them, but it also includes challenges related to continuous optimization in finding the shortest trajectory for Dubins airplane vehicle. The problem is addressed by sampling possible approaching angles to the targets, and a solution is found by the Randomized Variable Neighborhood Search (RVNS) method. A feasibility of the proposed solution is demonstrated by an empirical evaluation on modified benchmarks for the OP instances to the scenarios with varying altitude of the targets.**

I. Introduction

The problem addressed in this paper is motivated by data collection missions in which an Unmanned Aerial Vehicle (UAV) is requested to gather information from a set of target locations while the vehicle travel budget is limited, e.g., due to a limited battery capacity or fuel tank volume. The locations are known in advance, and each location is specified as a 3D point. Moreover, each location is associated with a reward value representing the importance of the measured data at the location, which can be for example a camera snapshot in surveillance missions [1] or measurements received by a remote transmission from sensor fields [2]. Having the limited travel budget, the problem is to select the most valuable locations that can be visited with respect to the available travel budget. This type of problems can be formulated as a variant of the *Orienteering Problem* [3] which is herein extended to consider the kinematic constraints of Dubins airplane model [4] for a fixed-wing aircraft. Therefore, we call the studied problem the *Dubins Airplane Orienteering Problem* (DA-OP).

In the regular OP, a set of all possible target locations with positive rewards is given together with the prescribed initial and final locations of the vehicle. The OP stands to maximize the sum of the collected rewards while the total traveled distance is shorter or equal to the given travel budget T_{max}. Since the budget may not allow visiting all the given targets, the OP is similar to the Knapsack problem in finding the most suitable subset of the targets, such that the sum of the collected rewards is maximized. However, it is necessary to evaluate the shortest path visiting the selected target locations to ensure T_{max} constraint and maximize the collected rewards, e.g., by visiting additional locations as a result of the saved travel cost by the shortest path. Therefore, the OP is also connected to the *Traveling Salesman Problem* (TSP) in which we find the shortest path for a particular subset of the target locations. Notice, the OP becomes a decision version of the TSP when the budget is equal to the minimal required travel distance to visit all the locations. Hence, the OP is at least NP-hard [5].

The authors are with the Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, 166 27 Prague, Czech Republic {vanapet1|faiglj|slamajak|penicrob}@fel.cvut.cz

The presented work has been supported by the Czech Science Foundation (GAČR) under research project No. 16-24206S.

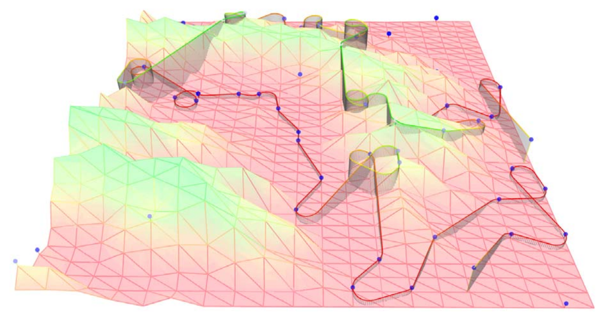


Fig. 1. An instance of the Dubins Airplane Orienteering Problem (DA-OP) together with its solution found by the proposed RVNS-based algorithm. The target locations are represented as the blue disks, and the color of the trajectory and the terrain corresponds to the specific altitude. The lowest parts are in the red while the highest parts are in green.

Unlike the regular OP, the studied DA-OP extends the target location to 3D and further consider limitations of fixed-wing UAVs. The introduced DA-OP can be considered as an extension of the existing *Dubins Orienteering Problem* (DOP) proposed in [6], which has been further generalized to the *Dubins Orienteering Problem with Neighborhoods* (DOPN) in [7], [8]. In the current paper, we formulate the OP for the Dubins airplane model in 3D and propose to solve the introduced DA-OP by a modified variant of the Variable Neighborhood Search (VNS) algorithm [9]. An example of the DA-OP solution is depicted in Fig. 1.

The rest of the paper is organized as follows. An overview of the related work on the Dubins airplane model and existing OP solvers is presented in the next section. The introduced DA-OP is formally defined in Section III and the proposed solver is described in Section IV. Results on empirical evaluation of the proposed approach are reported in Section V. Finally, Section VI concludes the paper.

II. Related work

The introduced *Dubins Airplane Orienteering Problem* (DA-OP) is mostly related to the two research fields which are directly utilized to address the DA-OP. The first field is related to approaches for determining the shortest possible paths for the Dubins airplane model in 3D, while the second field is a class of methods for solving the OP. The most related approaches of both these fields are further discussed in the rest of this section.

A. Shortest paths for the Dubins Airplane Model

The problem of finding the shortest curvature-constrained path in 2D was addressed by L. E. Dubins [10] in 1957. He showed that the optimal path connecting two locations with prescribed leaving and arrival headings of Dubins vehicle is composed of up to three segments which are either straight segments (S) or circle segments (C). It results in two basic path types of the so-called Dubins maneuvers: CSC and CCC, for which zero length segments are allowed. All the path types can be determined by a closed-form expression which represents a fast method for finding the optimal solution of a simple trajectory planning between two configurations of Dubins vehicle.

The shortest curvature-constrained path in 3D is studied in [11] where the authors proved that every minimizer is either a helicoidal arc or of the form CSC or CCC. This results in the analogy to the 2D curvature-constrained maneuvers since the Dubins maneuver is a special case. However, to the best of our knowledge, there exists no analytic solution of the shortest curvature-constrained path in the 3D.

A suboptimal approach for the 3D path generation is proposed in [12] which enables to satisfy arbitrary initial and final configurations of the vehicle while the constraint on the pitch angle is met. The resulting path is a CSC maneuver, and the Dubins maneuver in the plane is utilized as an initial estimation of the requested maneuver. In [13], numerical and geometric approaches for generating CSC paths in 3D are proposed. The authors claim that the numerical approach finds the optimal solution assuming the points are sufficiently far apart but no formal proof is provided.

The constraint on the pitch angle determines the maximal climb/dive angle and the authors of [4] introduce the *Dubins airplane model* to address both constraints of real UAVs: the bounded curvature and pitch angle. The proposed model treats the vertical position changes independently to the horizontal movement. The Dubins airplane model is further modified in [14] to be more consistent with a fixed-wing UAV kinematics. In [15], the authors propose to utilize 7-th order Bézier curves to compose the path as an alternative approach. However, in this paper we considered the Dubins airplane model [4] as a suitable model for extension of the curvature-constrained data collection planning in 3D.

B. Solving OP-based problems

The DA-OP is a direct extension of the recently introduced DOP [6] where the vehicle altitude is fixed to a constant value. Similarly to the DOP, the basic properties and possible approaches to address the DA-OP can be defined by the underlying combinatorial optimization problems: the *Dubins Traveling Salesman Problem* (DTSP) [16] and the regular Euclidean OP [3]. Therefore a brief overview of the existing approaches is provided in this section.

The first type of methods to the DTSP are decoupled approaches where the sequence of the visits to the targets is determined prior the optimization of the headings, e.g., by solving the Euclidean TSP as in the Alternating algorithm [16]. However, the most related approach to the DA-OP is the sampling based method [17] that allows transforming the DTSP to the Generalized Asymmetric TSP (GATSP) by using a discrete set of possible headings for each target location. Further, the GATSP can be transformed to the Asymmetric TSP (ATSP) that can be solved by existing solvers, e.g., Lin-Kernighan heuristic [18]. Finally, the DTSP can also be addressed by soft-computing techniques such as genetic [19] and memetic [20] algorithms and also by recently proposed unsupervised learning [21], [22].

The proposed DA-OP is a variant of the OP with the limited travel budget T_{max}, and thus only a subset of the target locations to be visited within T_{max} has to be determined together with the respective sequence of the visits to the selected targets. Since there probably does not exist an algorithm for solving the introduced DA-OP directly, the related existing approaches are Euclidean OP solvers for which a detailed survey can be found in [23]. The most related OP approach is the Variable Neighborhood Search (VNS) metaheuristic proposed by the authors of [9] that has been modified in [24] by Sevkli et al. to tackle the OP. Since the proposed solver of the DA-OP is leveraging on this method, it is described in the next paragraph in a more detailed way.

The basic idea of the VNS-based OP solver [24] is to search the solution space with a predefined set of neighborhood structures which are critical for the algorithm performance. The algorithm starts with an initial solution and explores the solution space by applying two procedures. The first one is the *shake* procedure, which aims to diversify the solution, and thus escapes from local minima. The second procedure is called *local search* and it utilizes the given neighborhood structures to optimize the current solution.

The original VNS algorithm exhaustively examines all possible solution modifications defined by the neighborhood structures. This can be too computationally demanding for large instances, and therefore, a randomized version of the VNS (RVNS) for the OP has been proposed in [24] to speed up a solution of the OP. The RVNS variant has been applied for solving the DOP in [6].

The introduced DA-OP is addressed by a variant of the RVNS utilized in [6], and the proposed method is presented in Section IV right after the formal introduction of the problem presented in the next section.

III. Problem Statement

The introduced *Dubins Airplane Orienteering Problem* (DA-OP) is an extension of the existing DOP [6] into

3D with respect to the constraints of the Dubins airplane model [4], which is a suitable model for a fix-wing aircraft with curvature and pitch angle constraints. The vehicle state q is represented by the configuration (p, θ, ψ), where $p = (x, y, z)$ stands for the vehicle position in 3D, $p \in \mathbb{R}^3$, $\theta \in \mathbb{S}^1$ is the vehicle heading, and $\psi \in \mathbb{S}^1$ is the vertical angle of the vehicle. The dynamics of the vehicle can be described by:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \end{bmatrix} = v \begin{bmatrix} \cos\theta \cdot \cos\psi \\ \sin\theta \cdot \cos\psi \\ \sin\psi \\ u_\theta \cdot \rho^{-1} \end{bmatrix}, \quad (1)$$

where v is a constant forward velocity of the vehicle, ρ stands for the minimum turning radius, and u_θ is the control input controlling the vehicle heading θ. The control output is considered to be limited by $u_\theta \in [-1, 1]$.

In the Dubins airplane model, the pitch angle ψ can be changed immediately to any value from the given interval $\psi \in [\psi_{min}, \psi_{max}]$, which does not fully correspond to real physical constraints but it is a suitable approximation for most of the fixed-wing UAVs.

The DA-OP follows the OP, where Dubins airplane model is utilized and the set of n target locations S is given as $S = \{s_1, s_2, \ldots, s_n\}$ for $s_i \in \mathbb{R}^3$. The DA-OP stands to find a feasible path which maximizes the sum of the collected rewards while the limited travel budget T_{max} is respected. The initial vehicle location s_1 and the final location s_n are given and their rewards are assumed to be zero $r_1 = r_n = 0$. For other locations, individual rewards are given and they are strictly positive, i.e., $r_i > 0$ for all $1 < i < n$.

The DA-OP consists of selecting a subset of k target locations from S and determination of the sequence to their visits which can be described as a sequence $\Sigma_k = (\sigma_1, \ldots, \sigma_k)$ where σ_i stands for the respective index of the target location, i.e., $1 \le i \le k$, $1 \le \sigma_i \le n$ and because of given initial and final locations, $\sigma_1 = 1$ and $\sigma_k = n$. In contrast to the Euclidean OP, the DA-OP contains also a determination of the vehicle heading for each selected location given by $\Theta_k = (\theta_{\sigma_1}, \ldots, \theta_{\sigma_k})$. The vehicle is not allowed to change its altitude while it visits a particular target location, and therefore, the pitch angle of the vehicle at the target location is prescribed to be zero. Hence, the vehicle state q_{σ_i} at the target σ_i is defined by the target location s_{σ_i} and the corresponding heading θ_{σ_i}. The DA-OP can be then defined as the following optimization problem:

$$\begin{aligned} \textit{maximize}_{\,k,\Sigma_k,\Theta_k} & \sum_{i=1}^{k} r_{\sigma_i} \\ \textit{subject to} & \sum_{i=1}^{k-1} \mathcal{L}(q_i, q_{i+1}) \le T_{max}\ , \\ & q_i = (s_{\sigma_i}, \theta_{\sigma_i}, 0), \\ & \sigma_1 = 1, \sigma_k = n\ , \\ & \sigma_i \ne \sigma_j, \forall i, j : i \ne j\ , \end{aligned} \quad (2)$$

where $\mathcal{L}(q_i, q_{i+1})$ is the length of the maneuver of the Dubins airplane model between q_i and q_{i+1} respecting (1).

IV. Proposed solution to the Dubins Airplane Orienteering Problem (DA-OP)

The proposed algorithm for the DA-OP leverages on the recently introduced DOP [6]. The main difference of the proposed approach to the DOP [6] is in the altitude changes that are caused by the limited pitch angle necessary to connect target locations by a continuous path. This limitation has to be considered because of possible different altitudes of the waypoints corresponding to the different elevations of the ground locations, which does not occur for the DOP on a plane. Moreover, in DOP, a solution of the Euclidean OP has similar length, especially for small minimal turning radii of the vehicle. Since the Dubins airplane model has limited pitch, the Euclidean distance cannot be simply utilized for a heuristic approach.

The proposed algorithm consists of three main parts to address challenges arising from the DA-OP. The first part is related to the vehicle heading that is uniformly sampled, and all three-dimensional maneuvers are pre-computed for each pair of possible states of the vehicle. The second part is an examination of possible collisions of the computed maneuvers with the terrain, and unfeasible maneuvers are discarded. Each 3D maneuver determined by the Dubins airplane model is sampled, into a finite set of states, and for each such a state the 3D model of the vehicle is checked for a possible collision with the terrain modeled as a mesh, e.g., using RAPID library [25]. Finally, having only the feasible 3D maneuvers connecting the possible waypoints, the solution of the DA-OP is found by the randomized variant of the VNS algorithm similarly as for the DOP in [6]. The first and third parts are further described in the following subsections as the collision test is performed similarly as in any motion planning algorithm such as PRM or RRT.

A. Heading Sampling and 3D Dubins Maneuvers

In the first step of the proposed algorithm, a particular instance of the DA-OP is discretized by sampling the possible vehicle states at each waypoint covering the respective target location. We utilize uniform sampling of m_i samples of the vehicle heading. The heading samples are given by a set $H_i = \{\theta_i^1, \ldots, \theta_i^{m_i}\}$ for the i-th waypoint.

Having the sampled state of the vehicle, a 3D maneuver for the Dubins airplane model connecting each pair of the vehicle state associated with the target locations is determined. Dubins airplane maneuvers are more complex to compute than Dubins maneuvers in the plane because of the possible altitude changes. Based on [4], the maneuvers are divided into three cases according to the altitude difference between the initial and final states of the vehicle: *low altitude*, *high altitude*, and *medium altitude*. The process of determining 3D maneuvers is detailed in [14], and therefore, we provide only a short overview of the procedure here.

First, Dubins maneuver is calculated as a two-dimensional projection of the final maneuver. Then, the next step depends on the altitude change. In the *low altitude* case, Dubins maneuver is long enough for achieving required altitude gain, i.e., the maximal pitch angle is not exceeded. The

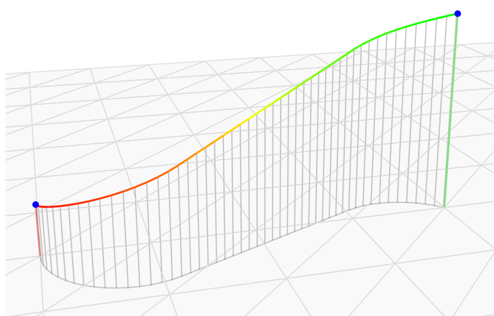


Fig. 2. A Dubins airplane maneuver in 3D with changing the vehicle altitude

calculated 2D Dubins maneuver is modified by changing the turn segments into helices segments, and the direction of the straight segment is changed accordingly. An example of such a *low altitude* maneuver is depicted in Fig. 2 where the altitude changes are highlighted by a color change.

For the *high altitude* case, the altitude gain is too high and cannot be achieved by altitude changes of the 2D trajectory. Therefore, the trajectory is modified by adding a certain number of spiral segments at the start or end of the maneuver to provide sufficient travel distance for the altitude correction according to the motion constraints of the Dubins airplane model. The maneuver is prolonged at the lower end because of minimizing potential terrain collisions.

The *medium altitude* case is a mix of the two previously described cases. The length of the generated 2D maneuver is not sufficient for the altitude change, but a necessary prolongation is smaller than one spiral turn. Therefore, in this case, the maneuver is generated by adding a third turning segment to achieve the required length for the correct altitude change respecting the Dubins airplane model.

Finally, vehicle maneuvers which intersect with the terrain are removed. Therefore, the proposed algorithm guarantees that the resulting solution is feasible and the vehicle does not collide with the ground.

B. VNS-based algorithm for the DA-OP

Having the sampled states of the vehicle at the target locations with the corresponding maneuvers for the Dubins airplane model, the instance of the DA-OP can be considered as a directed graph where each state stands for the graph node and the edges connecting the nodes are the determined maneuvers. The proposed solution of the DA-OP is based on the VNS discrete optimization technique [24] that searches for a maximal rewarding feasible tour within the graph such that the tour cost does not exceed the limited travel budget T_{max}. The VNS is a meta-heuristic algorithm which searches the solution space using neighborhood structures. Its performance depends on the neighborhood structures utilized for the sequence changes in the `shake` and `localSearch` procedures of Algorithm 1.

The current solution of the proposed algorithm is represented by a sequence P of the all targets except the initial and terminal locations which are prescribed, i.e., P is a sequence of $n-2$ targets $P = (\sigma'_1, \sigma'_2, \ldots, \sigma'_{n-2})$. P is

Algorithm 1: Randomized VNS for the DA-OP

Data: Targets S with the associated rewards r_i, the travel budget T_{max}, sampled vehicle states and the lengths of the corresponding maneuvers
Result: Solution represented by Σ_k

P, Σ_k := foundInitialSolution()
while *termination condition is not met* **do**
  P' := shake(P)
  P'' := localSearch(P')
  Σ''_k := selectLocations(P'', T_{max})
  if $\mathcal{L}(\Sigma''_k) \leq T_{max}$ **and** $R(\Sigma''_k) > R(\Sigma_k)$ **then**
    P := P''
    Σ_k := Σ''_k
  end
end

a permutation of labels of the targets $S \setminus \{s_1, s_n\}$, and thus it holds that for any $\sigma'_i \in P$, $\sigma'_i \neq 1$ and $\sigma'_i \neq n$. This allows the VNS algorithm to modify the whole sequence P without maintaining the prescribed initial and terminal locations. Thus, the VNS neighborhood structures work on the all targets in P.

Due to the limited budget T_{max}, only a subset of k locations can be visited, and therefore, the first $k-2$ elements of P are considered for a solution of the DA-OP that is prolonged by the prescribed initial and terminal locations $\sigma_1 = 1$ and $\sigma_k = n$, i.e., Σ_k is constructed from P as $\Sigma_k = (\sigma_1, \sigma'_1, \sigma'_2, \ldots, \sigma'_{k-2}, \sigma_k)$. However, it is necessary to determine the number of selected targets k to be visited by the trajectory that does not exceed T_{max}. Such a value of k is iteratively determined in the `selectLocations` procedure as the highest number for which the trajectory satisfies T_{max}. The trajectory for a particular k is found using the pre-computed maneuvers as follows.

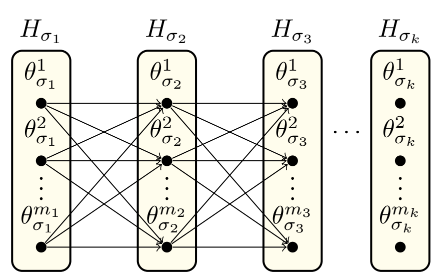


Fig. 3. An example of the graph structure for finding the shortest data collection tour for uniformly sampled headings at the target locations

Since multiple samples of the vehicle state are considered, the sequence Σ_k itself does not fully specify the data collection trajectory. Therefore, each target location has associated particular vehicle state for each sampled heading value, and we can create a graph structure representing possible paths connecting the waypoints of the particular targets in the current sequence Σ_k, see Fig. 3. Then, this graph is used to find the shortest tour by a feed-forward search which time complexity can be bounded by $\mathcal{O}(km^2)$, where m is the number of samples per each target and k is the current number of the selected targets to be visited.

After that, the current sequence Σ''_k is evaluated to be a new best solution found so far at the Step 6 of Algorithm 1. The length of the trajectory represented by Σ''_k is $\mathcal{L}(\Sigma''_k)$ and the sum of the collected rewards along the sequence is $R = \sum_{i=1}^{k} r_{\sigma_i}$, which is denoted $R(\Sigma''_k)$ for brevity.

The important part of the VNS-based search algorithm is a generation of the first feasible solution in the `foundInitialSolution` procedure by an iterative insertion of the target locations to the sequence. Then, the initial solution is improved by searching for possible insertion/removal of other targets such that, a target with the highest ratio f of the reward increase to the data collection trajectory prolongation is selected and inserted to the best position in the sequence:

$$f = \frac{R(\Sigma_{k+1}) - R(\Sigma_k)}{\mathcal{L}(\Sigma_{k+1}) - \mathcal{L}(\Sigma_k)}, \quad (3)$$

where Σ_k stands for the solution sequence before the insertion of the particular target.

The VNS-based algorithm utilized four different neighborhood structures which locally changes P to maximize the sum of the collected rewards. In each iteration, the actual solution is randomly changed to explore the state space in the procedure `shake` which chooses from the first two neighborhood structures:

- **Insert** selects a random element in P and moves it to a different randomly chosen position in P,
- **Exchange** selects two random elements in P and exchanges them.

Unlike the previous procedure which aims to explore the space and escape from the local minima, the `localSearch` procedure tries to optimize the current solution and eventually reaches the global optima. In the original VNS, the exhaustive search of all possible changes is used which can be computationally demanding for instances with a high number of targets. In contrast, the utilized randomized variant of the VNS examines only n^2 randomly selected local changes, where n is the number of all targets. Two different neighborhood structures are applied to locally optimize the current solution:

- **Path insert** selects a random sub-sequence from P and moves it to a different randomly chosen position in P,
- **Path exchange** selects two random sub-sequences in P and exchanges them.

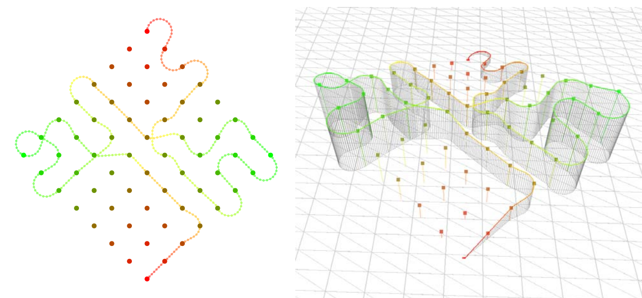
The termination condition can be chosen arbitrarily according to the application scenario. A high number of iterations provides a better solution at the cost of computational requirements. For example the algorithm can be terminated after the specified number of iterations or sooner, e.g., after a given maximum number of iteration without the solution improvement.

V. Results

The proposed VNS-based approach for the DA-OP has been evaluated using standard benchmarks for the Euclidean OP introduced by Chao et al. [26] that have been modified for the 3D problems that require the Dubins airplane model. In particular, we consider the benchmarks Set 64 with 64 target locations where a positive reward value associated with each target location is selected from the interval (0, 42] except the initial and final location which has zero rewards as in the ordinary OP. We define the altitude z_i of each target as a multiplication of the reward value and a specified constant β:

$$z_i = \beta \cdot r_i, \quad (4)$$

which means the most rewarding locations are at the highest altitudes, and thus makes the problem suitable for evaluation of the Dubins airplane model with altitude changes.



(a) 2D projection (b) 3D projection

Fig. 4. An example solution of Set 64 instance with $\beta = 0.1$ and $T_{max} = 80$. The locations at the highest altitude are the most rewarding (shown in green) and the locations at the lowest altitude with the zero reward are shown in red.

Four different values of $\beta \in \{0, 0.05, 0.1, 0.15\}$ have been utilized to show the relationship between the altitude change and the path length. The vehicle turning radius is set to $\rho = 0.7$, the minimum pitch angle to $\psi_{min} = -10°$, and the maximum pitch angle to $\psi_{max} = 20°$. The allowed ascension ratio is greater than the descension to respect properties of real vehicles. The number of heading samples is set to 16 which has been empirically found as a suitable trade-off between the computational requirements and the quality of found solutions. An example of the DA-OP instance is depicted in Fig. 4 in which the asymmetric limits of the pitch angle results in different path patterns near the start and end of the final solution.

The proposed DA-OP solver has been implemented in C++ with the Dubins airplane model according to [14]. All the presented results have been computed using a single core of the AMD Phenom 1090T CPU running at 3.2 GHz. The algorithm has been terminated after 10 000 iterations of the main loop.

Average values of the collected rewards for the particular T_{max} and β are depicted in Fig. 5. The results indicate that for small altitude changes (i.e., $\beta = 0.05$), the algorithm is capable of finding *low altitude* maneuvers for all possible maneuvers. With the increased altitude changes for $\beta = 0.1$, the ascending pitch angle is still sufficient, but the limited descenting angle causes small modifications of the original Dubins maneuver found for the 2D projection. Further, for $\beta = 0.15$, the altitude changes are so huge that additional

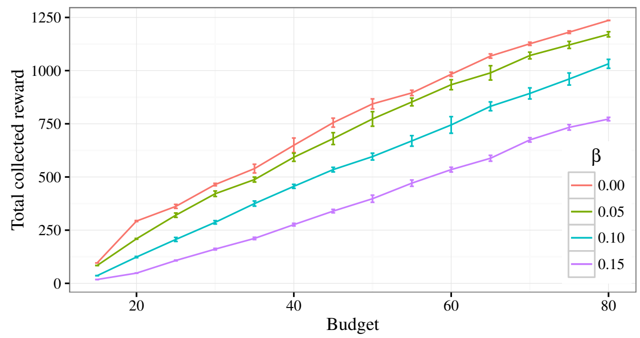


Fig. 5. Average values of the total collected rewards for the given travel budget and particular β. The results are computed from 10 trials.

spiral segments have to be inserted, which increases the length of the maneuvers, and thus the travel budget T_{max} is spent for visiting lower number of targets, and the sum of the collected rewards is lower.

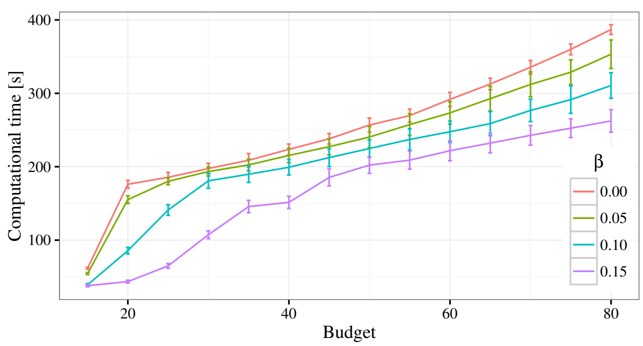


Fig. 6. Average required computational times based on the given travel budget for various β determined from 10 trials

The required computational time is shown in Fig. 6. The proposed VNS-based algorithm runs about hundreds of seconds for the examined instances with 64 targets. It can be observed that the computational time increases with the maximal travel budget, which is natural as a high travel budget allows to visit more locations, and thus more combinations, insertions, and path exchanges can be performed.

VI. Conclusion

In this paper, we introduce a problem formulation for data collection planning with Dubins airplane model suitable for 3D scenarios with fixed-wing UAVs and limited travel budget. The formulation is based on the DOP extended for the Dubins airplane model, and we call the introduced problem as the *Dubins Airplane Orienteering Problem* (DA-OP). We propose to utilize the VNS-based algorithm to solve the DA-OP in which possible vehicle headings are sampled by uniform sampling strategy. The performance of the algorithm has been evaluated in a series of DA-OP instances, and regarding the presented results, the proposed approach seems to be feasible.

Since the proposed approach is probably the first extension of the orienteering problems in 3D data collection planning with UAVs, we aim to investigate the performance of other 3D vehicle models within the presented solution framework for the DA-OP.

References

[1] K. Vicencio, B. Davis, and I. Gentilini, “Multi-goal path planning based on the generalized traveling salesman problem with neighborhoods,” in *IROS*, 2014, pp. 2985–2990.

[2] D. G. Macharet, A. A. Neto, V. F. da Camara Neto, and M. F. Campos, “Data gathering tour optimization for dubins’ vehicles,” in *IEEE Congress on Evolutionary Computation (CEC)*, 2012, pp. 1–8.

[3] P. Vansteenwegen, W. Souffriau, and D. Van Oudheusden, “The orienteering problem: A survey,” *European Journal of Operational Research*, vol. 209, no. 1, pp. 1–10, 2011.

[4] H. Chitsaz and S. M. LaValle, “Time-optimal paths for a dubins airplane,” in *46th IEEE CDC*, 2007, pp. 2379–2384.

[5] B. L. Golden, L. Levy, and R. Vohra, “The orienteering problem,” *Naval research logistics*, vol. 34, no. 3, pp. 307–318, 1987.

[6] R. Pěnička, J. Faigl, P. Váňa, and M. Saska, “Dubins orienteering problem,” *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1210–1217, 2017.

[7] R. Pěnička, J. Faigl, P. Váňa, and M. Saska, “Dubins orienteering problem with neighborhoods,” in *ICUAS*, 2017, pp. 1555–1562.

[8] J. Faigl and R. Pěnička, “On close enough orienteering problem with dubins vehicle,” in *IROS*, 2017, (to appear).

[9] P. Hansen and N. Mladenović, “Variable neighborhood search: Principles and applications,” *European journal of operational research*, vol. 130, no. 3, pp. 449–467, 2001.

[10] L. E. Dubins, “On curves of minimal length with a constraint on average curvature, and with prescribed initial and tetz rminal positions and tangents,” *American Journal of Mathematics*, pp. 497–516, 1957.

[11] H. J. Sussmann, “Shortest 3-dimensional paths with a prescribed curvature bound,” in *34th IEEE CDC*, vol. 4, 1995, pp. 3306–3312.

[12] G. Ambrosino, M. Ariola, U. Ciniglio, F. Corraro, A. Pironti, and M. Virgilio, “Algorithms for 3d uav path generation and tracking,” in *45th IEEE CDC*, 2006, pp. 5275–5280.

[13] S. Hota and D. Ghose, “Optimal path planning for an aerial vehicle in 3d space,” in *49th IEEE CDC*, 2010, pp. 4902–4907.

[14] M. Owen, R. W. Beard, and T. W. McLain, “Implementing dubins airplane paths on fixed-wing uavs,” in *Handbook of Unmanned Aerial Vehicles*. Springer, 2015, pp. 1677–1701.

[15] A. A. Neto, D. G. Macharet, and M. F. Campos, “3d path planning with continuous bounded curvature and pitch angle profiles using 7th order curves,” in *IROS*, 2015, pp. 4923–4928.

[16] K. Savla, E. Frazzoli, and F. Bullo, “On the point-to-point and traveling salesperson problems for dubins’ vehicle,” in *American Control Conference*. IEEE, 2005, pp. 786–791.

[17] K. Obermeyer, P. Oberlin, and S. Darbha, “Sampling-based roadmap methods for a visual reconnaissance uav,” in *AIAA Guidance, Navigation, and Control Conference*, 2010, p. 7568.

[18] K. Helsgaun, “An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic,” *European Journal of Operational Research*, vol. 126, no. 1, pp. 106–130, 2000.

[19] X. Yu and J. Hung, “A genetic algorithm for the dubins traveling salesman problem,” in *IEEE International Symposium on Industrial Electronics*, 2012, pp. 1256–1261.

[20] X. Zhang, J. Chen, B. Xin, and Z. Peng, “A memetic algorithm for path planning of curvature-constrained uavs performing surveillance of multiple ground targets,” *Chinese Journal of Aeronautics*, vol. 27, no. 3, pp. 622–633, 2014.

[21] J. Faigl and P. Váňa, “Self-organizing map for the curvature-constrained traveling salesman problem,” in *ICANN*, 2016, pp. 497–505.

[22] J. Faigl and P. Váňa, “Unsupervised learning for surveillance planning with team of aerial vehicles,” in *IJCNN*, 2017, pp. 4340–4347.

[23] A. Gunawan, H. C. Lau, and P. Vansteenwegen, “Orienteering problem: A survey of recent variants, solution approaches and applications,” *European Journal of Operational Research*, vol. 255, no. 2, pp. 315–332, 2016.

[24] Z. Sevkli and F. Sevilgen, “Variable neighborhood search for the orienteering problem,” *Computer and Information Sciences–ISCIS*, pp. 134–143, 2006.

[25] “Rapid - robust and accurate polygon interference detection,” [cit. 15-05-2017]. [Online]. Available: http://gamma.cs.unc.edu/OBB/

[26] I.-M. Chao, B. L. Golden, and E. A. Wasil, “A fast and effective heuristic for the orienteering problem,” *European Journal of Operational Research*, vol. 88, no. 3, pp. 475–489, 1996.