2017 International Conference on
Unmanned Aircraft Systems (ICUAS)
June 13-16, 2017, Miami, FL, USA

Dubins Orienteering Problem with Neighborhoods

Robert Pěnička, Jan Faigl, Petr Váňa and Martin Saska

Abstract— **In this paper, we address the Dubins Orienteering Problem with Neighborhoods (DOPN) a novel problem derived from the regular Orienteering Problem (OP). In the OP, one tries to find a maximal reward collecting path through a subset of given target locations, each with associated reward, such that the resulting path length does not exceed the specified travel budget. The Dubins Orienteering Problem (DOP) requires the reward collecting path to satisfy the curvature-constrained model of the Dubins vehicle while reaching precise positions of the target locations. In the newly introduced DOPN, the resulting path also respects the curvature constrained Dubins vehicle as in the DOP; however, the reward can be collected within a close distant neighborhood of the target locations. The studied problem is inspired by data collection scenarios for an Unmanned Aerial Vehicle (UAV), that can be modeled as the Dubins vehicle. Furthermore, the DOPN is a useful problem formulation of data collection scenarios for a UAV with the limited travel budget due to battery discharge and in scenarios where the sensoric data can be collected from a proximity of each target location. The proposed solution of the DOPN is based on the Variable Neighborhood Search method, and the presented computational results in the OP benchmarks support feasibility of the proposed approach.**

I. INTRODUCTION

Unmanned Aerial Vehicles can be used for effective autonomous data collection missions [1] where the goal is to collect sensory information from a predefined set of target locations. A standard approach for multi-goal path planning in data collection scenarios is based on solving the Traveling Salesman Problem (TSP). The planning problem is called the Dubins Traveling Salesman Problem (DTSP) [2] if it is desired to plan a data collection path for the curvature-constrained Dubins vehicle such as the fixed wing UAV or dynamically constrained moving multi-rotor UAV.

For data collection scenarios where the sensory data can be gathered within a vicinity of the target locations, the path planning problem can be formulated as the Traveling Salesman Problem with Neighborhoods (TSPN). By measuring the data within a neighborhood, i.e., from wireless sensors or with a wide-angle camera, the TSPN produces shorter paths compared to the regular TSP because of saving unnecessary visits of the exact target locations [3]. The variant for the Dubins vehicle with neighborhoods is called the Dubins Traveling Salesman Problem with Neighborhoods (DTSPN) [4].

In TSP-based formulations, the total travel cost is being minimized; however, these formulations do not allow to explicitly addressed limited travel budget of the utilized aerial vehicle. Therefore, in the presented paper, we introduce a generalization of the Dubins Orienteering Problem called the Dubins Orienteering Problem with Neighborhoods to find maximal rewarding data collection path for a Dubins vehicle with the constrain on the total travel cost. In the regular Orienteering Problem, each target location has assigned reward and the problem is to find a path from a prescribed starting location to a given ending location such that the path maximizes the sum of the collected rewards while the tour length is within the given travel budget. The OP has been introduced to the computer science and operational research by Tsiligirides [5] in 1984. Although data collection missions can be formulated as one of the mentioned variants of the TSP, the real vehicles such as the fixed-wing UAVs or multi-rotor UAVs have limited flight budget due to the battery discharge. Therefore, visitations of all the target locations with the limited budget cannot be ensured as it is required in the regular TSP formulation. By assigning a priority to each target location, defined as a reward that can be collected from the target, the task can be specified as the Orienteering Problem, see the illustrative scenario in Fig. 1.

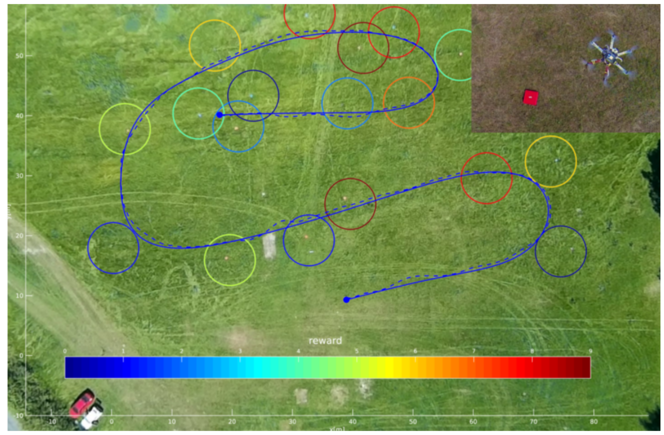


Fig. 1: The top view of the workspace provided by a UAV flying 150 m above the quadrotor following a trajectory computed by our DTSPN method [4]. With assigned reward to each target location, the scenario can be described as the Dubins Orienteering Problem with Neighborhoods (DOPN) in which we request the resulting path has to satisfy the limited travel budget while it also maximizes the sum of the collected rewards.

The introduced DOPN combines both the limited curvature constraint of Dubins vehicle and the ability to measure the data within a predefined circular neighborhood around each target location. The problem is called the Dubins Orienteering Problem [6] as the OP with the Dubins model of the considered UAVs and the found path has typically lower sum of the collected rewards due to the limited minimal turning radius. On the other hand, a solution of the proposed

Authors are with the Czech Technical University, Faculty of Electrical Engineering, Technicka 2, 166 27, Prague, Czech Republic {penicrob|faiglj|vanapet1|saskam1}@fel.cvut.cz

978-1-5090-4494-8/17/$31.00 ©2017 IEEE

DOPN may provide data collection paths with higher rewards because of not visiting the exact positions of the targets. The introduced DOPN is addressed by the herein proposed generalization of the Variable Neighborhood Search (VNS) metaheuristic [7] already deployed for the OP in [8] and utilized for solving the DOP in [6].

The remainder of this paper is organized as follows. A summary of related work is presented in the next section. Section III introduces a formal definition of the proposed DOPN. In Section IV, we present the VNS-based solution of the addressed problem. An evaluation of the method is presented in Section V and a conclusion is provided in VI.

II. Related Work

The newly addressed Dubins Orienteering Problem with Neighborhoods belongs to a wider class of the Orienteering Problems [9] where the objective is to find a length limited path between starting and ending locations with maximal sum of collected rewards from a subset of the specified target locations. The DOPN is also related to the existing Dubins Traveling Salesman Problem [2] and its variant with Neighborhoods [4]. The main difference of the variants of the TSP over the OP is the unlimited travel budget and the requirement to visit all specified target locations. Regarding the literature review, there is no report on solving the DOPN, and therefore, a summary of relevant variants and approaches to the OP and DTSP(N) are presented in this section.

The Orienteering Problem has been studied since 1984 when Tsiligirides [5] introduced Euclidean version of the Orienteering Problem (further denoted the EOP in this paper) together with the deterministic D-algorithm and stochastic S-algorithm. The S-algorithm is based on the Monte-Carlo method with creation of multiple feasible paths and selection of the one with the highest collected reward. The D-algorithm is based on the method for the vehicle routing problem [10]. Tsiligirides also created three Orienteering Problem benchmark instances [11] further denoted as the Set 1, Set 2, and Set 3 with up to 33 target locations.

The fast and effective heuristic for the EOP by Chao et al. [12] considers only target locations reachable within the prescribed budget (inside the respective ellipse around the prescribed start and final locations). The heuristic uses an initial set of generated paths that contains all reachable target locations and tries to improve the most rewarded path by simple operations with the target locations. The used operators are two-point exchange and one-point movement together with the 2-Opt operation. Furthermore Chao proposed two symmetrical benchmark instances, a diamond shaped Set 64 and square shaped Set 66 with up to 66 target locations.

The proposed DOPN method is based on the Variable Neighborhood Search (VNS) [7] a metaheuristic by Hansen and Mladenović for combinatorial optimization. The VNS employs predefined neighborhood structures, in terms of the OP also describable as operations with target locations, that are used to improve an initial solution by the *shaking* and *local search* procedures. The first VNS-based approach to the EOP [13] uses neighborhood structures that motivate the proposed solution of the DOPN. The method randomly changes current best path by either path move or exchange in the *shaking* procedure and then tries to optimize the changed path by multiple one point moves or exchanges in the *local search* procedure to find better path than the current best one.

In our previous work [6], we introduce the Dubins Orienteering Problem (DOP), a variant of the Orienteering Problem for Dubins vehicle, and we proposed the VNS-based method to solve it. The method uses a similar neighborhood structures as the aforementioned VNS method for the EOP [8]. To tackle the problem of finding suitable path for the curvature constrained Dubins vehicle, we proposed an equidistant sampling of heading angle at the target locations. The VNS technique then searches for the most rewarded path together with the appropriate sequence of sampled heading angles to fit the path length within the budget constraint.

Probably the first approach addressing the generalization of the OP to the Orienteering Problem with Neighborhoods (OPN) has been proposed in [14] and further improved in [15]. The approach is based on the unsupervised learning of the Self-Organizing Map (SOM) for the Prize-Collecting Traveling Salesman Problem with Neighborhoods (PC-TSPN) [16], i.e., a variant of the TSP that combines maximization of the rewards (prizes) and minimization of the path length. The approach has been further extended to the variants with multiple vehicles in the OP [17] and also multi-vehicle the PC-TSPN [18]. Moreover, the SOM has also been applied to the DTSP and DTSPN in [19]. Howard, the SOM-based approach has not been deployed for the combined problem formulation as the DOPN.

The introduced DOPN is related to the existing approaches to the DTSP [20] and the DTSPN [4]. The most relevant approaches are the sampling based variants of DTSP where the heading angles at the target locations are sampled and the problem is transformed to the Asymmetric TSP (ATSP) [21] that can be solved optimally for the specified sampling. A similar approach can be used for the DTSPN [22] where both the heading angles and the positions within the neighborhood are sampled, and the problem is transformed into the Generalized TSP (GTSP) and further to the ATSP that can be solved, e.g., by the LKH solver [23].

The proposed approach to solve the introduced DOPN leverages on the previous work, most specifically on the VNS-based solution of the DOP [6] that is generalized by the ideas of the sampling-based solutions of the DTSPN [22].

III. Problem Statement

The proposed Dubins Orienteering Problem with Neighborhoods is inspired by the data collection scenarios with Unmanned Aerial Vehicles. In the former Euclidean Orienteering Problem, a set of given target locations (each with assigned reward) are requested to be visited by the data collection vehicle while the length of the data collection path has to be within the specified travel budget T_{max}. The goal of the EOP is to find a path from the prescribed starting location to the defined ending location such that it maximizes the sum of the collected rewards R and meets the T_{max} constraint.

Although the problem definition of the EOP suits to UAVs with the budget limitation, it does not meet the curvature constraint of Dubins vehicle, and thus a solution of the OP may produce unfeasible paths. Here, we refer to our previous work [6] that formally introduces the Dubins Orienteering Problem (DOP).

For the data collection using UAV we can usually utilize an ability to acquire the data within a small neighborhood radius δ around the target location without reaching the target location precisely. Such an ability can lead to higher sum of the collected rewards R for the same travel budget T_{max}, and thus we can benefit from using the novel problem formulation called the Dubins Orienteering Problem with Neighborhoods. In the following section, we formally introduce the DOPN.

A. Dubins Orienteering Problem with Neighborhoods

In all existing variants of the Orienteering Problem [9], a set of target locations $S = \{s_1, \cdots, s_n\}$ to be visited is given and each target location $s_i = (t_i, r_i) = (x_i, y_i, r_i)$ is defined by its position in a plane $t_i = (x_i, y_i)$, $t_i \in \mathbb{R}^2$ and the respective reward r_i collected once the vehicle visits the location. The reward is strictly positive, i.e., $r_i \in \mathbb{R}_{>0}$, for all target locations except the starting and ending ones. The main objective of the OP is to find a subset $S_k \subseteq S$ with k target locations that maximizes the collected reward $R = \sum_{r_i \in S_k} r_i$. This objective is similar to the NP-hard Knapsack problem.

Even though the determination of the maximal rewarding subset S_k is the main objective, the OP path length is constrained by the specified travel budget T_{max}, which usually requires to determine a sequence to visit the target locations in S_k such that the shortest path connecting the locations of S_k meets the T_{max} constraint. The sequence of visit can be described as a permutation $\Sigma = (\sigma_1, \cdots, \sigma_k)$, where $1 \leq \sigma_i \leq n$, $\sigma_i \neq \sigma_j$ for $i \neq j$ and $\sigma_1 = 1$, $\sigma_k = n$. Notice that the OP specifies the starting location s_1 and ending location s_n, therefore they must be kept in the permutation Σ. Furthermore we assume strictly positive rewards $r_i > 0$ for all target locations i $\in (2, n-1)$ except the starting and ending locations $r_1 = r_n = 0$. The finding an appropriate sequence to visit the target locations $S_k \subseteq S$ is similar to the NP-hard TSP where the only objective is to minimize the path length over all target locations S, i.e., $S_k = S$ and the travel budget is not prescribed. The addressed Orienteering Problem is also NP-hard, as it combines the aforementioned TSP and also the NP-hard Knapsack problem of selecting a subset S_k.

For the Dubins Orienteering Problem, the reward collecting path has to respect the kinematic model (1) of Dubins vehicle, where the state of the vehicle $q = (p, \theta)^T = (x, y, \theta)^T$ is described by its position $p = (x, y)$ in plane, i.e., $p \in \mathbb{R}^2$ and the vehicle heading angle θ, $\theta \in \mathbb{S}^1$. The kinematic model assumes a constant forward velocity v of the vehicle that is controlled by the input u, which controls the vehicle straight ahead or steers the vehicle left or right with the minimal turning radius ρ.

$$\dot{q} = \begin{bmatrix} \dot{p} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = v \begin{bmatrix} \cos\theta \\ \sin\theta \\ \frac{u}{\rho} \end{bmatrix}, u \in [-1, 1] \ . \tag{1}$$

For this specific vehicle, Dubins showed that the shortest path between two states can be computed analytically and is either of type CSC or CCC, where 'C' stands for turning right or left and 'S' means going straight [24]. The use of the curvature-constrained Dubins vehicle in the DOP requires to consider the heading angles at the target locations and also the distance for paths between the locations has to respect the vehicle limitations. Note that by changing the heading angles at the target locations we also change the length of the final path which still has to be within the T_{max}. For the DOP we use a vector $\Theta = (\theta_{\sigma_1}, \cdots, \theta_{\sigma_k})$ that holds the selected heading angles θ_{σ_i} at the target locations from S_k with the sequence of visit defined by Σ. The distance between two states of Dubins vehicle q_{σ_i} and q_{σ_j}, at target locations s_{σ_i} and s_{σ_j}, is denoted as $\mathcal{L}_d(q_{\sigma_i}, q_{\sigma_j})$ which is the shortest Dubins maneuver [24] connecting s_{σ_i} and s_{σ_j} and its one of the six possible maneuvers [24].

In the Dubins Orienteering Problem with Neighborhoods, the existing DOP needs to be extended to allow reward collection within a circular neighborhood of each target location. The specific neighborhood is described by the neighborhood radius δ that defines a δ-radius disk centered at the respective target location. For simplicity, all target locations have the same value of δ except the starting location s_1 and ending location s_k with zero neighborhood radius. In contrast to the DOP where k, S_k, Σ, and Θ are determined, the DOPN also requires determination of particular locations of the waypoints $P_k \subseteq \mathbb{R}^2$ at which the rewards are collected by a vehicle that is within δ distance from the respective target locations, i.e., $p_{\sigma_i} \in P_k, t_{\sigma_i} \in S_k$ and $|(p_{\sigma_i}, t_{\sigma_i})| \leq \delta$. Based on these preliminaries, the proposed Dubins Orienteering Problem with Neighborhoods (DOPN) can be formulated as the optimization problem:

$$\begin{aligned} & \underset{k, S_k, P_k, \Sigma, \Theta}{\textit{maximize}} \ R = \sum_{i=1}^{k} r_{\sigma_i} \\ & \textit{subject to} \ \sum_{i=2}^{k} \mathcal{L}_d(q_{\sigma_{i-1}}, q_{\sigma_i}) \leq T_{max} \ , \\ & \qquad q_{\sigma_i} = (p_{\sigma_i}, \theta_{\sigma_i}), p_{\sigma_i} \in P_k, \theta_{\sigma_i} \in \Theta \ , \\ & \qquad \|p_{\sigma_i}, t_{\sigma_i}\| \leq \delta \ , \ i \in (2, k-1) \ , \\ & \qquad \|p_{\sigma_1}, t_{\sigma_1}\| = 0 \ , \ \|p_{\sigma_k}, t_{\sigma_k}\| = 0 \ , \\ & \qquad \sigma_1 = 1 \ , \ \sigma_k = n \ . \end{aligned} \tag{2}$$

In the DOPN, we need to determine four variables: S_k, P_k, Σ, and Θ. The subset of k target locations S_k and Σ are typical for the EOP, where S_k influences the sum of the collected rewards R, and the permutation Σ defines the length of the path over S_k constrained by the budget T_{max}. In addition to these variables, a solution of the DOP also provides a sequence of heading angles Θ at the target locations that influences the path length because of Dubins vehicle. Finally, in the DOPN, we search for the additional selection of the waypoints $P_k = (p_{\sigma_1}, \cdots, p_{\sigma_k})$ within the neighborhoods of the respective target locations, implied by $\|p_{\sigma_i}, t_{\sigma_i}\|$ that also influences the final path length. Regarding the computational complexity, the DOPN is therefore more challenging than the existing EOP and DOP because it contains an additional part of the continuous optimization to determine the locations of the waypoints in $\mathbb{R}^2$.

IV. Proposed Approach for the DOPN

The proposed solution of the introduced Dubins Orienteering Problem with Neighborhoods is based on the Variable Neighborhoods Search metaheuristic [7]. Even though the existing solution of the DOP already uses the Dubins vehicle model [6], a solution of the proposed DOPN requires an extension of the existing VNS-based Orienteering methods to utilize the reward collection within the neighborhood radius.

VNS is a metaheuristic for combinatorial optimization applicable on various problems [25]. VNS uses an iterative improvement of the currently best achieved solution inside *shake* and *local search* procedures. The algorithm operates on l predefined neighborhood structures $N_l, l = 1, \ldots, l_{max}$ also expressible as operations that are used inside the procedures. The *shake* procedure starts with the current best solution of the combinatorial problem and randomly changes the solution to escape from the possible local minimum. Such randomly created solution is then used in the *local search* to find the best solution in the particular solution neighborhood. The solution produced by the *local search* is then set as the new best solution if it improves the current one, and the solution neighborhood with higher number l is used in the next iteration. In particular, the proposed method for the DOPN is based on the Randomized Variable Neighborhood Search (RVNS) variant of VNS with the randomized *local search* procedure.

To tackle the continuous optimization part of the problem, we propose sampling-base approach to determine the location of the waypoint p_{σ_i} inside the neighborhood of the target location s_{σ_i}. A constant number of samples o are placed equidistantly along the circle with the radius δ that is centered at the respective target location t_{σ_i}. Then, similarly to the solution of the DOP [6], the heading angles of Dubins vehicle at each neighborhood sample are also sampled in a set of discrete values. The possible heading angles from the interval $\langle 0, 2\pi)$ are proportionally sampled into m values. In [6], we propose a sampling based approach for the Dubins Orienteering Problem; however, for solving the DOPN, additional discretization of the neighborhood leads to $(o \cdot m)$ samples per each target location except the starting s_1 and ending s_n locations where the zero radius neighborhood requires only m heading samples. Such a number of samples can be prohibitively large, and solving the DOPN by the combinatorial VNS can be computationally too demanding.

Therefore, the number of samples is reduced by removing unreachable target locations. We propose to preselect a set of reachable target locations S_r such that it contains only target locations reachable on the path between starting and ending locations within the T_{max} distance. The set S_r then contains s_i such that $\mathcal{L}_d(s_1, s_i) + \mathcal{L}_d(s_i, s_n) \leq T_{max}$ for any sampled position inside the neighborhood and for any sampled heading angles. This procedure (denoted in the Alg. 1 as *getReachableLocations*) decreases the number of samples, especially for small travel budgets.

The proposed VNS-based method for the DOPN internally represents the actual problem solution by a vector $v = (q_{\sigma_1}, \ldots, q_{\sigma_k}, q_{\sigma_{k+1}}, \ldots, q_{\sigma_n})$, where $P = (q_{\sigma_1}, \ldots, q_{\sigma_k})$ with k target locations is the actual path for the vehicle defined by S_k and ordered according to Σ. The vector $(q_{\sigma_{k+1}}, \ldots, q_{\sigma_n})$ then consists of all other unvisited target locations from S_r. By using the solution vector v with all reachable target locations, the further explained neighborhood operators for *shake* and *local search* procedures can use the current solution (represented by v) not only for changing the ordering of target locations already present in the path but also for introduction of the previously unvisited target locations.

The proposed VNS-based algorithm for the DOPN is summarized in Algorithm 1 and further detailed description of the neighborhood structures for the *shake* IV-A and *local search* IV-B procedures are described in next sections. For brevity, we denote the DOPN path (defined by S_k, Σ, Θ and P_k) as P, the sum of the rewards collected by the path as $R(P)$ and its length as $\mathcal{L}(P)$.

Algorithm 1: VNS based method for the DOPN

Input : S – Set of the target locations
Input : T_{max} – Maximal allowed budget
Input : o – Number of waypoints for each target
Input : m – Number of heading values for each waypoints
Input : l_{max} – Maximal neighborhood number
Output: P – Found data collecting path

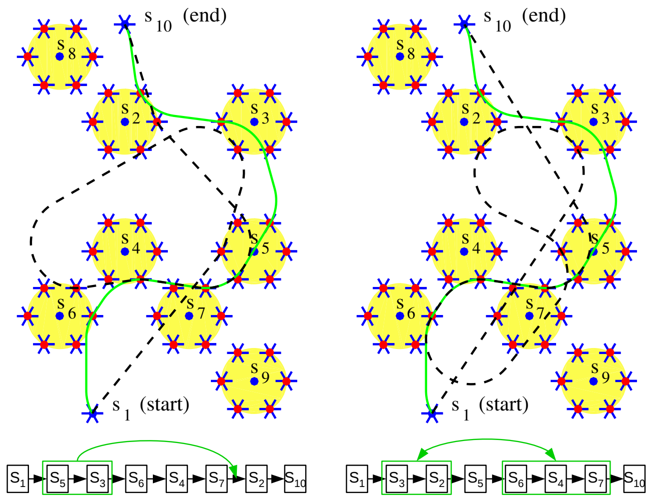
```
S_r ← getReachableLocations(S)
P ← createInitialPath(S_r,T_max)                    // greedy
while Stopping condition is not met do
    l ← 1
    while l ≤ l_max do
        P' ← shake(P, l)
        P'' ← localSearch(P', l)
        if L_d(P'') ≤ T_max and R(P'') > R(P) then
            P ← P''
            l ← 1
        else
            l ← l + 1
```

A. Shake Procedure

The *shake* procedure of the proposed VNS-based solution of the DOPN has two possible neighborhood operators for $l = 1$ and $l = 2$. Both operators consist of moving or exchanging randomly selected parts of the currently best found solution path P. Beside changing the order in which the locations are visited, the method also (after each operation) finds an appropriate waypoint and heading angle a the waypoint for each target location in the path, such that the path length is minimal for the particular order of targets. Notice, that by changing the order of the target locations, the shortest path usually uses different waypoint locations in the target neighborhoods and also different heading angles, and thus the waypoint locations and headings have to be determined after each operation.

Path Move operator for $l = 1$ (shown in Fig. 2a) uses a randomly selected path $(q_{\sigma_i}, \ldots, q_{\sigma_j})$ with $1 < i < j < n$ from the actual solution. Such a selected part of the path is then moved to a new randomly selected position inside the solution vector.

The **Path Exchange** with $l = 2$ is the second neighborhood operator used in the *shake* procedure. In this operator, the randomly selected sub-path $(q_{\sigma_i}, \ldots, q_{\sigma_j})$ is exchanged with a different non-overlapping sub-path $(q_{\sigma_v}, \ldots, q_{\sigma_w})$, see the visualization in Fig. 2b.



(a) **Path Move** (b) **Path Exchange**

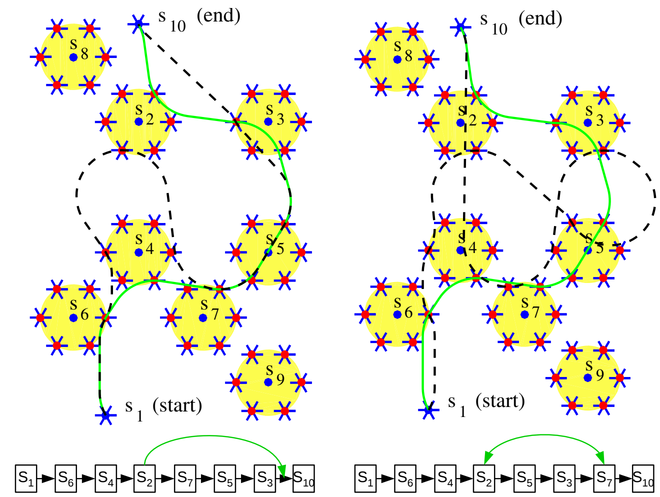
Fig. 2: Examples of the used shaking neighborhood operators **Path Move** and **Path Exchange** with $o = 6$ samples of the target location neighborhood and $m = 6$ samples of the heading angle at each waypoint location. The original paths (dashed black) are changed within the neighborhood to new and shorter paths (green).

B. Local Search Procedure

The *local search* procedure is used to find a local minimum on the path produced by *shaking*. Contrary to the *shake* procedure with only one move/exchange of the solution sub-path, the *local search* tries simple random operations for a number of times that is equal to the square of the number of the target locations. The appropriate waypoints and heading angles at the selected target locations has to be also found after each simple operation to minimize the overall path length.

One Point Move neighborhood corresponds to $l = 1$. This simple neighborhood operator randomly selects one target and move it to a different position within the solution vector.

One Point Exchange, shown in Fig. 3b, exchanges two different randomly selected target locations within the solution path.



(a) **One Point Move** (b) **One Point Exchange**

Fig. 3: Examples of the used local search neighborhood operators **One Point Move** and **One Point Exchange** with the $o = 6$ samples of the waypoint locations per each neighborhood of the target locations and $m = 6$ samples of the heading angles at the waypoint location. The original paths (dashed black) are changed to new paths (green).

V. RESULTS

The proposed solution of the Dubins Orienteering Problem with Neighborhoods has been evaluated on existing benchmark datasets for the Orienteering Problem [11]. Three test instances **Set 1**, **Set 2**, **Set 3** created by Tsiligirides [5] have up to 32 target locations. The datasets **Set 64**, **Set 66** by Chao [12] contain up to 66 target locations with diamond and square shaped placement.

To the best of our knowledge, there is not a method for solving the DOPN, and therefore, the proposed method has been compared with the existing solution of the Dubins Orienteering Problem [6] as the DOPN becomes the DOP for $\delta = 0$. Besides, we also compare the proposed method with the existing SOM-based approach to the Orienteering Problem with Neighborhoods [15] which corresponds to the DOPN for $\rho = 0$.

For evaluation of the proposed randomized VNS method, we run the experiments 10 times for each individual problem instance and particular algorithm, i.e., for each travel budget T_{max} in each dataset and each algorithm. The computational results were calculated using C++ implementation running on a single core of Intel i7 3.4GHz CPU. The presented computational times are the average required time to solve a single problem instance. During the solution of the particular problem instance, a combined stopping criterion was the maximal number of 10000 iterations with the maximal number of 5000 iterations without improvement together with the maximal allowed computational time of 4 hours. Both the number of samples o of the waypoint locations at the δ perimeter around each target locations and the number of sampled heading values m were set to 16 samples except the zero neighborhood radius δ=0 with o=1, and also in cases with the zero minimal turning radius ρ=0 with m=1. Abbreviations used further in the presentation of the achieved computational results are listed in Table I.

TABLE I: Abbreviation related with the results

Set 1, **Set 2**, **Set 3**	Test instances created by Tsiligirides [5].
Set 64, **Set 66**	Test instances proposed by Chao [12].
SOM OPN	Self-organizing map-based solution of the OPN [15]

The proposed VNS-based method for the DOPN has been compared using the minimal turning radius ρ=0 with existing SOM-based OPN approach at three representative problems for different neighborhood radii δ. Fig. 4 shows that the proposed VNS-based method outperforms the existing SOM-based solution of the OPN mainly in cases without overlapping the neighborhoods, i.e., $\delta \leq 1.5$.

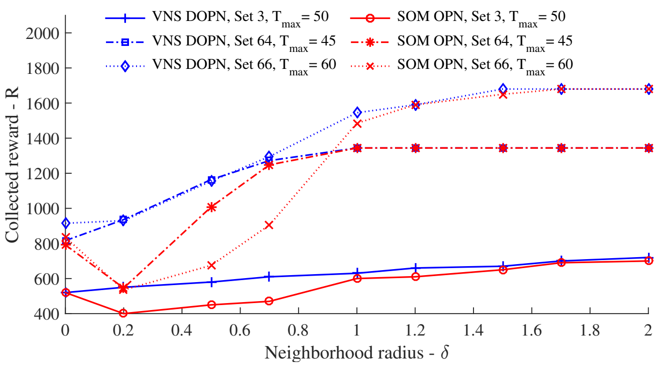


Fig. 4: A comparison of the self-organizing map-based solution of OPN (red) and the VNS-based OPN (blue) in solving OPN, i.e., DOPN with ρ=0, for the selected problems and various neighborhood radius δ

The proposed method uses the randomized VNS metaheuristic, and therefore, the found path does not always collect the same rewards R. Table II presents the maximal achieved sum of the rewards R_{max}, the average collected reward R_{avg}, standard deviation of the collected reward R_{std}, and the computational time for the selected instances with the neighborhood radius δ and turning radius ρ on the problem instances of **Set 66** with T_{max} =60.

TABLE II: Comparison of DOPN for different neighborhood radius δ and turning radius ρ on the Set 66 with $T_{max} = 60$

		δ=0.0	δ=0.2	δ=0.5	δ=1.0	δ=1.5
ρ=0.0	R_{max}	**915**	**950**	**1155**	**1545**	**1680**
	R_{avg}	899	933	1142	1531	1674
	R_{std}	26.1	13.5	12.0	16.4	13.4
	comp. time	46 sec	26 min	22 min	22 min	4 min
ρ=0.3	R_{max}	**895**	**900**	**1110**	**1485**	**1615**
	R_{avg}	887	886	1082	1441	1598
	R_{std}	9.1	11.4	32.9	40.1	18.9
	comp. time	3 min	50 min	80 min	72 min	67 min
ρ=0.5	R_{max}	**895**	**885**	**1090**	**1520**	**1615**
	R_{avg}	883	873	1062	1466	1576
	R_{std}	26.8	11.5	22.0	41.6	29.2
	comp. time	6 min	37 min	65 min	65 min	72 min
ρ=1.0	R_{max}	**870**	**870**	**990**	**1465**	**1585**
	R_{avg}	857	856	946	1427	1531
	R_{std}	14.4	15.2	29.0	37.7	63.1
	comp. time	5 min	40 min	54 min	102 min	96 min
ρ=1.5	R_{max}	**785**	**825**	**960**	**1410**	**1455**
	R_{avg}	726	799	930	1312	1352
	R_{std}	54.1	19.2	30.0	72.9	97.0
	comp. time	71 sec	27 min	57 min	86 min	69 min

The computational results indicate that for almost all turning radii the higher neighborhood radius leads to both higher maximal and higher average sum of the collected rewards. On the contrary, a larger turning radius requires a longer path between the waypoints, and therefore a lower number of target locations can be visited, and the collected sum of rewards is lower.

A comparison of the maximal achieved sum of the collected rewards for **Set 3**, **Set 64** and **Set 66** for the particular travel budget T_{max} are presented in Tables III, IV, and V for $\delta = \{0.0, 0.5, 1.0\}$ and $\rho \in \{0.0, 1.0\}$. Selected solutions found by the proposed DOPN method are shown in Fig. 5 together with the respective values of the collected rewards.

TABLE III: Results for the Set 3

T_{max}	δ=0.0		δ=0.5		δ=1.0	
	ρ=0.0	ρ=1.0	ρ=0.0	ρ=1.0	ρ=0.0	ρ=1.0
15	170	160	180	180	210	190
20	200	180	250	230	300	280
25	260	250	320	310	370	360
30	320	310	380	370	450	440
35	390	380	450	440	500	480
40	430	420	500	480	570	540
45	470	450	550	530	600	580
50	520	470	580	570	630	610
55	550	530	620	600	670	640
60	580	560	650	630	710	670
65	610	590	680	650	750	710
70	640	600	720	690	790	740
75	670	640	750	720	800	780
80	700	670	790	750	800	800
85	740	690	800	790	800	800
90	770	740	800	800	800	800
95	790	770	800	800	800	800
100	800	790	800	800	800	800
105	800	800	800	800	800	800
110	800	800	800	800	800	800

TABLE IV: Results for the Set 64

T_{max}	δ=0.0		δ=0.5		δ=1.0	
	ρ=0.0	ρ=1.0	ρ=0.0	ρ=1.0	ρ=0.0	ρ=1.0
15	96	96	204	198	300	300
20	294	252	432	360	576	552
25	390	336	564	486	744	708
30	474	420	714	576	948	912
35	576	510	888	714	1158	1110
40	714	624	1068	876	1290	1236
45	816	696	1164	930	1344	1320
50	900	798	1248	1008	1344	1344
55	984	894	1320	1074	1344	1344
60	1062	948	1344	1140	1344	1344
65	1116	1014	1344	1212	1344	1344
70	1188	1074	1344	1254	1344	1344
75	1236	1116	1344	1290	1344	1344
80	1284	1170	1344	1308	1344	1344

The computational time and the sum of the collected rewards of proposed VNS-based method for the DOPN is significantly influenced by the number of heading angle samples m, which has been shown in our previous work on the DOP [6]. For the herein addressed DOPN, the computational

TABLE V: Results for the Set 66

T_{max}	δ=0.0		δ=0.5		δ=1.0	
	ρ=0.0	ρ=1.0	ρ=0.0	ρ=1.0	ρ=0.0	ρ=1.0
5	10	0	20	0	35	0
10	40	40	70	55	105	90
15	120	95	160	130	220	200
20	205	195	265	245	380	350
25	280	275	390	350	540	540
30	400	370	495	450	685	655
35	465	455	605	530	870	835
40	545	540	725	640	980	940
45	650	645	830	715	1135	1090
50	730	705	920	770	1275	1235
55	815	820	1035	870	1390	1330
60	915	865	1155	940	1545	1375
65	980	955	1255	990	1620	1570
70	1060	1070	1350	1085	1665	1575
75	1140	1115	1445	1185	1680	1650
80	1215	1170	1535	1240	1680	1680
85	1270	1235	1605	1305	1680	1680
90	1340	1295	1635	1390	1680	1680
95	1395	1365	1680	1485	1680	1680
100	1455	1420	1680	1550	1680	1680
105	1520	1470	1680	1610	1680	1680
110	1550	1530	1680	1640	1680	1680
115	1595	1565	1680	1660	1680	1680
120	1625	1605	1680	1680	1680	1680
125	1670	1640	1680	1680	1680	1680
130	1680	1670	1680	1680	1680	1680

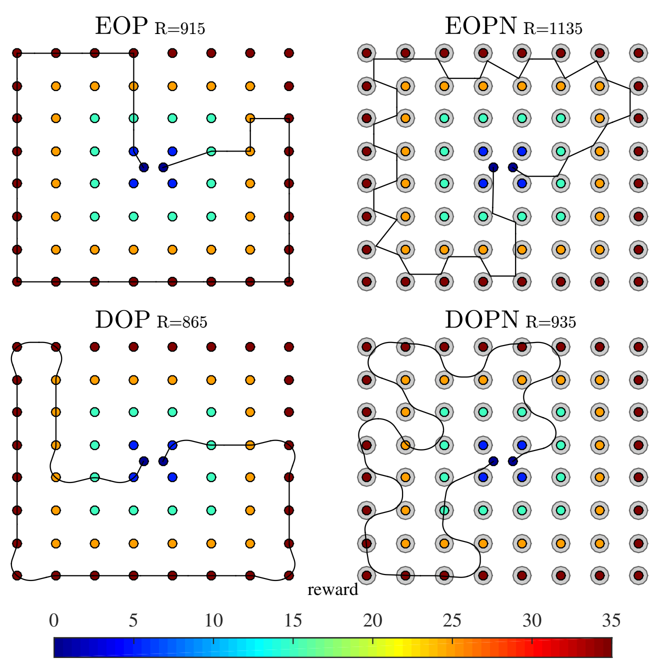


Fig. 5: Solution of the OP, OPN, DOP, and DOPN with respective collected rewards R. The minimal turning radius $\rho = 1.0$ is used for the variants of the OP with Dubins vehicle. The neighborhood radius δ in respective variants is set to $\delta = 0.5$. In all four presented solutions, the same travel budget $T_{max} = 60$ on benchmark Set 66 [12] is utilized.

time, and thus the solution quality is also influenced by the number of samples of waypoints locations o. Fig. 6 shows the influence of the maximal collected rewards R_{max} and the corresponding computational time for increasing number of samples o for the neighborhood radius δ=0.5, turning radius ρ=0.5, and the number of the heading samples m=16.

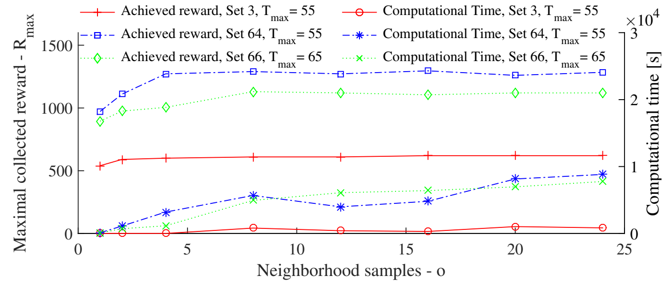


Fig. 6: Influence of the maximal sum of the collected rewards R_{max} and the required computational time on the number of samples o

VI. CONCLUSIONS

In this paper, we introduce a novel problem called the Dubins Orienteering Problem with Neighborhoods that is formulated as an extension of existing Dubins Orienteering Problem to address data collection planning for curvature-constraint vehicle and scenarios where it is possible to retrieve data within a circular neighborhood of each target location. The proposed solution is based on the Variable Neighborhood Search metaheuristic for combinatorial optimization. A sampling of the possible locations on the circular border of the neighborhood of each target location is utilized as a suitable discretization schema to determine the location of the waypoints at which rewards are collected from the respective target locations. The computational results show feasibility of the proposed solution. The waypoints within the circular neighborhood of the target locations saves the travel cost which allow to collect rewards from other targets, and thus improve the solution. Furthermore, the results for the Euclidean Orienteering Problem with Neighborhoods (EOPN) indicate that the proposed VNS-based approach outperforms the only existing SOM-based approach for the EOPN in the problem instances with non-overlapping neighborhoods.

For future work, we intend to investigate the possibility of improving both the neighborhood and heading angle samples during the algorithm to limit the influence of the number of samples and their placement to the solution quality and computational requirements. Besides, we also plan to validate the proposed method in realistic experiments of data collection scenario. We will use the UAV platform designed for the MBZIRC competition (see http://mrs.felk.cvut.cz/mbzirc for examples of experimental deployment of the system).

ACKNOWLEDGMENT

The presented work has been supported by the Czech Science Foundation (GAČR) under research project No. 16-24206S and research project No. 17-16900Y. Access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum, provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures" (CESNET LM2015042), is greatly appreciated.

References

[1] S. Siebert and J. Teizer, "Mobile 3d mapping for surveying earthwork projects using an unmanned aerial vehicle (uav) system," *Automation in Construction*, vol. 41, pp. 1 – 14, 2014.

[2] K. Savla, E. Frazzoli, and F. Bullo, "On the point-to-point and traveling salesperson problems for dubins' vehicle," in *ACC*, 2005, pp. 786–791 vol. 2.

[3] J. Faigl, V. Vonásek, and L. Přeučil, "Visiting convex regions in a polygonal map," *Robotics and Autonomous Systems*, 2012, (*accepted, paper in press*), http://dx.doi.org/10.1016/j.robot.2012.08.013.

[4] P. Váňa and J. Faigl, "On the dubins traveling salesman problem with neighborhoods," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 4029–4034.

[5] T. Tsiligirides, "Heuristic methods applied to orienteering," *The Journal of the Operational Research Society*, vol. 35, no. 9, pp. 797–809, 1984.

[6] R. Pěnička, J. Faigl, P. Váňa, and M. Saska, "Dubins orienteering problem," *IEEE Robotics and Automation Letters*, vol. PP, no. 99, pp. 1–1, Feb. 2017.

[7] N. Mladenović and P. Hansen, "Variable neighborhood search," *Computers & Operations Research*, vol. 24, no. 11, pp. 1097 – 1100, 1997.

[8] Z. Sevkli and F. E. Sevilgen, "Variable neighborhood search for the orienteering problem," in *21th International Symposium on Computer and Information Sciences (ISCIS)*, 2006, pp. 134–143.

[9] 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.

[10] A. H. Anthony Wren, "Computer scheduling of vehicles from one or more depots to a number of delivery points," *Operational Research Quarterly (1970-1977)*, vol. 23, no. 3, pp. 333–344, 1972.

[11] "The Orienteering Problem: Test Instances – Department of Mechanical Engineering, University of Leuven," citied on 2016-08-20. [Online]. Available: http://www.mech.kuleuven.be/en/cib/op#section-2

[12] 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.

[13] Z. Sevkli and F. E. Sevilgen, *Variable Neighborhood Search for the Orienteering Problem*. Springer Berlin Heidelberg, 2006, pp. 134–143.

[14] G. Best, J. Faigl, and R. Fitch, "Multi-robot path planning for budgeted active perception with self-organising maps," in *IROS*, 2016, pp. 3164–3171.

[15] J. Faigl, R. Pěnička, and G. Best, "Self-organizing map-based solution for the orienteering problem with neighborhoods," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct 2016, pp. 001 315–001 321.

[16] J. Faigl and G. A. Hollinger, "Unifying multi-goal path planning for autonomous data collection," in *IROS*, 2014, pp. 2937–2942.

[17] J. Faigl, "On self-organizing maps for orienteering problems," in *IJCNN*, 2017, (accepted, to appear).

[18] J. Faigl and G. A. Hollinger, "Autonomous data collection using a self-organizing map," *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, no. 99, pp. 1–13, 2017.

[19] J. Faigl and P. Váňa, "Unsupervised learning for surveillance planning with team of aerial vehicles," in *IJCNN*, 2017, (accepted, to appear).

[20] I. Cohen, C. Epstein, P. Isaiah, S. Kuzi, and T. Shima, "Discretization-based and look-ahead algorithms for the dubins traveling salesperson problem," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 383–390, Jan 2017.

[21] C. E. Noon and J. C. Bean, "An efficient transformation of the generalized traveling salesman problem," *INFOR: Information Systems and Operational Research*, vol. 31, no. 1, pp. 39–44, 1993.

[22] K. J. Obermeyer, P. Oberlin, and S. Darbha, "Sampling-based roadmap methods for a visual reconnaissance uav," in *in AIAA Conference on Guidance, Navigation, and Control*, 2010.

[23] 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.

[24] L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, vol. 79, no. 3, pp. 497–516, 1957.

[25] P. Hansen and N. Mladenović, "Variable neighborhood search: Principles and applications," *European Journal of Operational Research*, vol. 130, no. 3, pp. 449 – 467, 2001.