Physical Orienteering Problem for Unmanned Aerial Vehicle Data Collection Planning in Environments with Obstacles

Robert Pěnička, Jan Faigl, and Martin Saska

***Abstract*—This paper concerns a variant of the Orienteering Problem (OP) that arises from multi-goal data collection scenarios where a robot with a limited travel budget is requested to visit given target locations in an environment with obstacles. We call the introduced OP variant the Physical Orienteering Problem (POP). The POP sets out to determine a feasible, collision-free, path that maximizes collected reward from a subset of the target locations and does not exceed the given travel budget. The problem combines motion planning and combinatorial optimization to visit multiple target locations. The proposed solution to the POP is based on the Variable Neighborhood Search (VNS) method combined with the asymptotically optimal sampling-based Probabilistic Roadmap (PRM*) method. The VNS-PRM* uses initial low-dense roadmap that is continuously expanded during the VNS-based POP optimization to shorten paths of the promising solutions, and thus allows maximizing the sum of the collected rewards. The computational results support the feasibility of the proposed approach by a fast determination of high-quality solutions. Moreover, an experimental verification demonstrates the applicability of the proposed VNS-PRM* approach for data collection planning for an unmanned aerial vehicle in an urban-like environment with obstacles.**

***Index Terms*—Motion and Path Planning; Aerial Systems: Applications**

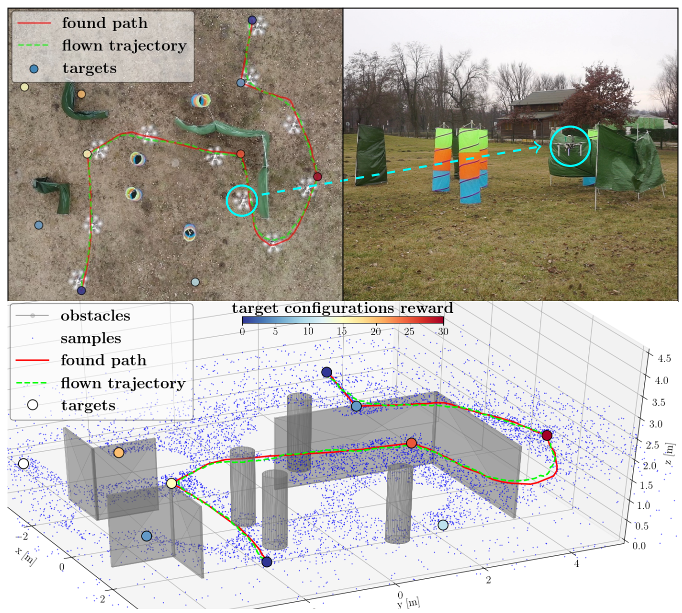


Fig. 1. Experimental verification of the proposed VNS-PRM* solution of the introduced Physical Orienteering Problem (POP) in outdoor data collection scenario with obstacles. We refer to https://youtu.be/xUXYEt4Gnvk for the video from the experimental verification.

I. Introduction

In this paper, we study a generalization of the Orienteering Problem (OP) [1] to address robotic route planning problems in environments with obstacles and with an arbitrary motion model of the used vehicle. The introduced problem is called the Physical Orienteering Problem (POP), and it can be considered as the OP explicitly deployed in the configuration space [2] where both the obstacles and vehicle motion constraints can be addressed. The OP belongs to multi-goal routing problems with profits where each target location has associated reward, and the problem sets out to maximize the sum of collected rewards without exceeding the specified travel budget. The POP stands, for the given initial and terminal locations, to select a subset of the locations and a sequence to visit them together with the determination of cost-efficient and collision-free paths between the individual locations to maximize the sum of collected rewards by saving vehicle travels to fit the budget. Hence, the route and path planning needs to be addressed in a single optimization problem to find a high-quality solution of the POP.

The motivation for the introduced problem is in data collection missions with Unmanned Aerial Vehicles (UAVs) in indoor and urban-like environments where multiple target locations need to be visited for collecting the requested data. Such a mission can be, e.g., to collect desired measurements at the particular locations of interest using UAV equipped with an onboard camera. Another example can be found in wireless sensor networks [3] where a UAV can be used to collect data from the sensors placed in the environment.

The flight time of today's UAVs is usually limited and visiting all target locations can be unfeasible, and therefore, each location can be assigned with a reward to prioritize the most important locations. The existing Euclidean OP [4] or its extension for Dubins vehicle [5] can be used to find the data collection plan. However, in a realistic robotic scenario, the operational environment can contain obstacles and motion

Manuscript received: February, 24, 2019; Revised May, 14, 2019; Accepted June, 4, 2019.

This paper was recommended for publication by Editor Nancy Amato upon evaluation of the Associate Editor and Reviewers' comments. The presented work has been supported by the Czech Science Foundation under research project No. 19-20238S and No. 17-16900Y. The authors acknowledge the support of the OP VVV funded project CZ.02.1.01/0.0/0.0/16_019/0000765 "Research Center for Informatics". The support of the CTU in Prague grant SGS17/187/13 is greatly appreciated.

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

Digital Object Identifier (DOI): see top of this page.

constraints of the utilized vehicles can be more complicated than the Dubins vehicle. Therefore, the POP is introduced to allow deploying budget-limited UAVs in realistic data collection missions in environments with obstacles, see a snapshot of the experimental deployment in Fig. 1.

The proposed solution of the introduced POP is denoted VNS-PRM* because it is based on the Variable Neighborhood Search (VNS) [6] metaheuristic tightly coupled with the asymptotically optimal sampling-based Probabilistic Roadmaps method (PRM*) [7]. The VNS-based combinatorial optimization searches for the rewarding subset of the target locations and order to visit them such that the interconnecting paths are further shortened by the PRM*. The PRM* is employed to create an initial low-dense roadmap of collision-free paths between the target locations. The initial roadmap is then incrementally expanded during the iterative optimization of the OP to improve paths of promising OP solutions. In this way, the VNS-PRM* simultaneously searches both the OP solution and configuration spaces.

The rest of the paper is organized as follows. An overview of related work is summarized in the next section. Section III formally introduces the POP and the proposed VNS-PRM* is described in Section IV. Evaluation results are reported in Section V and concluding remarks are outlined in Section VI.

II. Related Work

The addressed POP combines motion planning and the OP, and thus we briefly overview the relevant sampling-based planning approaches, the most related OP methods, and also existing approaches combining routing with motion planning.

The Rapidly-Exploring Random Trees (RRT) [8] and the Probabilistic Roadmaps (PRM) [9] can be considered as the most fundamental approaches with many modifications and variants [2]. Regarding the addressed POP, the RRT* and PRM* [7] are considered as the most relevant approaches, albeit other methods with path optimality criteria can be utilized [10]. The introduced POP is a multi-goal planning problem which requires a multi-query search, and thus the PRM* is a suitable technique for the proposed solution.

The OP belongs to routing problems with profits, and it has been introduced by Tsiligirides [4] in 1984. Since then, numerous algorithmic solutions have been proposed [1] together with a wide range of formulation variants [11]. The OP can be defined as an Integer Linear Programming (ILP) problem [1] and solved by Branch-and-Bound [12] or Branch-and-Cut [13] algorithms. Existing heuristics, e.g., particle swarm optimization [14] or ant colony optimization [15], provide solutions of similar quality but within a fraction of time required for finding the optimal solution. In particular, the VNS-based [16] solution of the OP performs as one of the best considering the computational time and solution quality, and therefore, the proposed solution builds on the VNS-based optimization operators.

The POP is also related to variants of the OP where the travel cost is not a length of the straight lines connecting the locations as in the regular Euclidean OP. The Dubins Orienteering Problem (DOP) [5] is an extension for Dubins vehicle [17] that requires to optimize the heading angle of the vehicle to find the most rewarding paths. The proposed VNS-PRM* significantly extends the VNS-based method for the DOP [5] by considering the OP in the configuration space with obstacles addressed by tightly coupled PRM* with online roadmap expansion. Besides, the DOP has been used for UAVs in wildfire observation planning [18] and further extended to the DOP with Neighborhoods (DOPN) addressed by the VNS [19] and unsupervised learning [20]. To the best of the authors' knowledge, the only OP variant considering the environments with obstacles is the approach presented in [21]. The method is based on a low level A* search in a grid of Dubins maneuvers to get around obstacles, which limits its application to instances without narrow passages and predefined heading angles. On the other hand, the proposed VNS-PRM* employs sampling-based motion planning that can be used to find collision-free paths also in the 3D with various vehicle motion constraints.

The motion planning combined with routing has been mostly studied in the context of the Traveling Salesman Problem (TSP) where the Physical TSP (PTSP) [22] combines TSP with real-time motion planning in video games. In robotics, a multi-tree Transition-based RRT [23] has been proposed for creating a collision-free roadmap for arbitrary routing problem. Several existing approaches combining the routing problems with motion planning have been introduced for scenarios with Autonomous Underwater Vehicles (AUV), e.g., planning mine countermeasures missions based on the PTSP [24], the Clustered TSP [25] and high-level mission planning [26] combined with motion planning.

The most similar existing problem to the POP is a variant of the Prize Collection Traveling Salesman Problem (PC-TSP) for AUV [27] that uses sampling-based methods for finding collision-free PC-TSP plans. The approach uses initially created PRM navigation roadmap for guiding a sampling-based motion tree considering the vehicle dynamics. A separate PC-TSP solver is used to prioritize the expansion of the motion tree along PC-TSP solutions found on the static navigation roadmap. Contrarily, the proposed VNS-PRM* uses tightly coupled asymptotically optimal PRM*, where vehicle dynamics is considered by different motion primitives, with the VNS-based OP solver within a single optimization algorithm that deals with narrow passages better than the decoupled approach of [27], as shown in Section V.

III. Problem Statement

The proposed Physical Orienteering Problem (POP) combines collision-free path planning with the combinatorial routing of the Orienteering Problem in a single optimization problem. Therefore, we outline the path planning first; then the POP is introduced as an extension of the regular OP with path planning to determine the most rewarding path that does not exceed the given travel budget $\mathrm{T}_{\max}$.

Having the world $\mathcal{W} = \mathbb{R}^2$ or $\mathcal{W} = \mathbb{R}^3$ with the obstacles $\mathcal{O} = \{\mathcal{O}_1, \ldots, \mathcal{O}_m\} \subset \mathcal{W}$, the point-to-point path planning problem is to determine a collision free-path for a robot $\mathcal{A} \subset \mathcal{W}$ between two locations in $\mathcal{W}$ such that the path avoids $\mathcal{O}$. The problem can be formulated using the notion of the configuration space $\mathcal{C}$ [2], which consists of all possible robot configurations $q \in \mathcal{C}$. Let $\mathcal{A}(q) \subset \mathcal{W}$ denotes geometry of the robot at a configuration q. The robot can move in the free space $\mathcal{C}_{free} = \mathcal{C} \setminus \mathcal{C}_{obs}$, where $\mathcal{C}_{obs} = \{q \in \mathcal{C} | \mathcal{A}(q) \cap \mathcal{O} \neq \emptyset\} \subseteq \mathcal{C}$ is a set of configurations where the robot $\mathcal{A}(q)$ collides with $\mathcal{O}$. A solution of the point-to-point path planing between initial $q_I \in \mathcal{C}_{free}$ and goal $q_G \in \mathcal{C}_{free}$ configurations is a path $\tau : [0,1] \to \mathcal{C}_{free}$ with $\tau(0) = q_I$ and $\tau(1) = q_G$, respectively. The cost to travel the path τ can be expressed as a cost function $c(\tau) \to \mathbb{R}_{\geq 0}$. In this work, w.l.o.g. we consider the cost to be the length of the path, i.e., $c(\tau) = \int_0^1 |\tau(t)| dt$. In addition to have a feasible path τ that avoids obstacles $\tau \in \mathcal{C}_{free}$, we are searching for the optimal path τ^* such that $c(\tau^*) = \min\{c(\tau) | \tau \textit{ is feasible}\}$ to find a solution of the introduced POP. Moreover, a single point-to-point collision-free optimal path planning is only a part of the POP, as we need to address finding a path over multiple target locations that can be arbitrarily ordered as in the solution of the OP.

The OP belongs to a class of routing problems with profits where each of all n predefined target locations $S = \{s_1, \ldots, s_n\}$, $s_i \in \mathcal{W}$ have associated reward $r_i \geq 0$ for $i \in \{1, \ldots, n\}$. The OP stands to maximize the sum of the collected rewards R by visiting a subset $S_l \subseteq S$ such that the path to visit S_l does not exceed the given limited travel budget $\mathrm{T_{max}}$. The initial and terminal locations of the path are prescribed and for simplicity they are denoted s_1 and s_n, respectively, both with the zero reward $r_0 = r_n = 0$. The OP is an optimization problem to find the subset S_l of l target locations together with a sequence to visit the target locations in S_l within $\mathrm{T_{max}}$. The sequence can be expressed as a permutation of the target location indexes $\Sigma = (\sigma_1, \ldots, \sigma_l)$ with $1 \leq \sigma_i \leq n$, $\sigma_i \neq \sigma_j$ for $i \neq j$, $\sigma_1 = 1$ and $\sigma_l = n$, because of the prescribed initial and terminal locations. The locations in the sequence have to be connected by a collision-free path not exceeding $\mathrm{T_{max}}$, and thus we need to combine routing and path planning for a solution of the introduced POP.

In the POP, the target locations $s_i \in \mathcal{W}$ of the OP correspond to the target configurations $Q = \{q_1, \ldots, q_n\}, q_i \in \mathcal{C}_{free}$, such that $s_i \in \mathcal{A}(q_i)$ for all $1 \leq i \leq n$. A solution of the POP is a sequence Σ of the configurations $Q_l \subseteq Q$ that maximizes R using collision-free paths with the sum of the cost satisfying $\mathrm{T_{max}}$. We propose to combine the solution of the combinatorial OP with path planning to determine paths τ_i connecting locations S_l in the sequence Σ such that the individual paths are feasibly connected at the target configurations $\tau_i(0) = q_{\sigma_i}$ and $\tau_i(1) = q_{\sigma_{i+1}}$ for $1 \leq i \leq l-1$. Besides, the total path length is limited by the travel budget $\sum_{i=1}^{l-1} c(\tau_i) \leq \mathrm{T_{max}}$. The POP can be understood as the OP in $\mathcal{C}$ and can be summarized in a single optimization problem (1).

$$\begin{aligned} \underset{l, Q_l, \Sigma, \tau_i}{\text{maximize}} \quad & R = \sum_{i=1}^{l} r_{\sigma_i} \\ \text{s.t.} \quad & \sum_{i=1}^{l-1} c(\tau_i) \leq \mathrm{T_{max}}, \\ & \sigma_1 = 1,\, \sigma_l = n,\, \tau_i \in \mathcal{C}_{free}, \\ & \tau_i(0) = q_{\sigma_i},\, \tau_i(1) = q_{\sigma_{i+1}},\, i = 1 \ldots l-1 \end{aligned} \tag{1}$$

The POP objective is to maximize the sum of the collected rewards R by visiting the target configurations Q_l. However, the budget limit $\mathrm{T_{max}}$ requires to evaluate the cost of the path to visit Q_l, and thus it requires to find the appropriate sequence Σ of the configurations together with collision-free paths connecting the configurations in the sequence. Finding the collision-free paths is a challenging problem and determining all possible paths connecting all the locations S is computationally very demanding. Moreover, optimal paths should be determined to ensure $\mathrm{T_{max}}$ while visiting as many highly rewarding locations as possible, which is even more computationally demanding. On the other hand, it is likely that a subset Q_l contains only a small portion of Q, and thus determining all paths is not necessary. Therefore, we propose to address the introduced POP by a combination of the asymptotically optimal motion planner PRM* with the VNS-based solution of the routing part of the POP to continuously improve the PRM* roadmap using the combinatorial solutions to expand the roadmap only in parts of $\mathcal{C}$ that can contribute to the solution of the POP.

IV. Proposed VNS-PRM* method for the POP

The proposed approach to solve the POP combines asymptotically optimal sampling-based PRM* [7] with the combinatorial metaheuristic VNS [6] to solve the OP on the incrementally constructed roadmap. The POP is addressed by a single VNS-based algorithm with an online improvement of the roadmap using PRM* to support finding collision-free trajectories in the configuration space to visit multiple target configurations. The selection of the target locations and sequence to visit them to maximize the sum of the collected rewards is thus optimized together with the paths connecting the selected target locations. The proposed VNS-PRM* combines both feedbacks from (i) the PRM* for finding the OP solution (i.e., the selection and sequence of targets) on the improving roadmap; (ii) the search space of the OP to guide the PRM* sampling of $\mathcal{C}_{free}$.

A. PRM for the Physical Orienteering Problem*

The PRM* is a multi-query asymptotically optimal motion planning algorithm that firstly randomly samples configurations in $\mathcal{C}_{free}$ and creates a graph $G = (V, E)$ (further denoted as the roadmap) by connecting k neighboring samples with a collision-free path. Contrary to the ordinary PRM with a fixed k, in the employed k-nearest PRM*, the value of k increases with the number of vertices m in G as $k(m) = k_{PRM} \log(m)$ where $k_{PRM} > k^*_{PRM} = e(1 + 1/d)$ and d is the dimension of $\mathcal{C}$ [7]. Hence, the VNS-PRM* uses a low-dense initial roadmap consisting of m_{init} random configurations and the target configurations Q. Dijkstra's algorithm is then used to interconnect the target configurations using shortest paths in G between all configurations $q_i, q_j \in Q$ with the respective lengths $c_{i,j} = c(\tau), \tau(0) = q_i, \tau(1) = q_j$.

The maximization of the collected rewards needs minimal path lengths $c_{i,j}$ to visit valuable target configurations within $\mathrm{T_{max}}$. High-quality paths require a large number of samples m_{init}, where most of the samples would not be used for the paths of the final solution of the POP. Therefore, the roadmap is continuously expanded during the VNS-based optimization with the focused sampling on the paths between promising target configurations, i.e., configurations that are in highly rewarded POP solutions found so far. The roadmap initialization and expansion are summarized in Alg. 1.

Algorithm 1: PRM* Initialization and Expansion

In/Out: $G(V, E)$ – existing roadmap, Q – target configurations,
$\mathcal{P} = \{\rho_{i,j}\}_{i,j=1,\ldots,n}$ – sampling density
Input : $M = \{m_{i,j}\}$ number of samples to add between q_i, q_j

1 $V_{new} \leftarrow \emptyset$; $E_{new} \leftarrow \emptyset$
2 **if** *q_1 q_n not connectable in G* **then** `// roadmap initialization`
3 $\quad V \leftarrow \emptyset$; $E \leftarrow \emptyset$
4 $\quad V_{new} \leftarrow Q \cup \{\text{UniformSample } \mathcal{C}_{free}\}_{1,\ldots,m_{init}}$
5 **else** `// roadmap expansion`
6 $\quad$ **foreach** *pair* $i, j \in (1, \ldots, n), i \neq j$ **do**
7 $\quad\quad V_{new} \leftarrow V_{new} \cup \{\text{EllipsoidSample}(q_i, q_j, c_{i,j})\}_{1,\ldots,m_{i,j}}$
8 $\quad\quad \rho_{i,j} \leftarrow \rho_{i,j} + m_{i,j}/\text{EllipsoidVolume}(q_i, q_j, c_{i,j})$
9 **foreach** $v \in V_{new}$ **do**
10 $\quad U \leftarrow \text{kNearest}((V \cup V_{new}, E), v, k(|V| + |V_{new}|)) \setminus v$
11 $\quad$ **foreach** $u \in U$ **do**
12 $\quad\quad$ **if** *CollisionFree*(v, u) **then**
13 $\quad\quad\quad E_{new} \leftarrow E_{new} \cup \{(v, u)\}$
14 $V \leftarrow V \cup V_{new}$; $E \leftarrow E \cup E_{new}$

During the roadmap expansion, new random configurations V_{new} are sampled in the hyperellipsoids (Line 7, Alg. 1) corresponding to all target configurations pairs. The hyperellipsoids are defined by their foci in the respective target configurations q_i and q_j, and by the major axis length equal to the actual shortest path length $c_{i,j}$ between the corresponding target configurations. An individual hyperellipsoid between q_i and q_j is equidistantly sampled for $m_{i,j}$ times [28]. The value of $m_{i,j}$ thus defines a priority in which particular path is optimized, and it is updated at each iteration of the VNS-based solution of the POP. The sampling densities of particular ellipsoids $\rho_{i,j}$ are stored and further used to prioritize sampling of low-density sampled ellipsoids.

B. VNS-based method for the POP

The VNS is based on the two main procedures called *shake* and *local search* to iteratively improve a single incumbent solution. The *shake* procedure performs a random change of the currently best-found solution v to leave a possible local optimum. The *local search* optimizes a randomly changed solution v' using a set of neighborhoods (described as operators) to increase the quality of the incumbent solution.

In the VNS for the POP, a solution is represented as a vector $v = (q_{\sigma_1}, \ldots, q_{\sigma_l}, \ldots, q_{\sigma_n})$ of all target configurations Q, where the first l items $(q_{\sigma_1}, \ldots, q_{\sigma_l})$ represent a path within $\mathrm{T_{max}}$ and the remaining part of v gathers the unvisited target configurations. The initial and terminal configurations are prescribed, and thus q_{σ_1} is always q_1 and q_{σ_l} is q_n. The operators of the *shake* and *local search* procedures change the order of target configurations in v to maximize the sum of the collected rewards $R(v) = R(v(l, Q_l, \Sigma)) = \sum_{i=1}^{l} r_{\sigma_i}$ while keeping the path length $\mathcal{L}(v) = \mathcal{L}(v(l, Q_l, \Sigma)) = \sum_{i=1}^{l-1} c_{\sigma_i,\sigma_{i+1}}$ within $\mathrm{T_{max}}$ by moving q_{σ_l} inside v. Thus, the operators change not only the sequence Σ but also the subset of the visited target configurations Q_l. The path of a solution v is found as the shortest path in the roadmap over the sequence of targets $(q_{\sigma_1}, \ldots, q_{\sigma_l})$.

The proposed VNS-PRM* is summarized in Alg. 2. The algorithm starts with $\mathrm{PRM^*initialSampling}()$ that uniformly samples $\mathcal{C}_{free}$ using m_{init} random configurations. Adding m_{init} samples is repeated until the initial q_1 and terminal q_n configurations are connectable by a path with $c_{1,n} \leq \mathrm{T_{max}}$ or until the maximal computational time is reached. The lengths $c_{i,j}$ are determined as the shortest paths between all pairs of the target configurations (Line 2). A greedy procedure is used to create initial incumbent solution v (Line 3) by inserting target configurations between q_1 and q_l (for $q_l = q_n$) according to the minimal path prolongation per target reward. The VNS-PRM* then iteratively improves the incumbent solution during which the roadmap expansions are performed to minimize lengths of promising solutions. The algorithm terminates if one of the stopping condition occurs: the maximal number of iterations, or the number of iterations without improvement, or the maximal computational time.

Algorithm 2: VNS-PRM* for the POP

Input : Q – target configurations, $\mathrm{T_{max}}$ – budget, m_{init} – VNS-PRM* initial number of samples, m_{exp} – number of expanding samples
Output: v – Found data collecting path

1 $G \leftarrow \text{PRM*initialSampling}(m_{init})$
2 updateRoadmapDistances $c_{i,j} \forall i, j \in (1, \ldots, n), i \neq j$
3 $v \leftarrow \text{createInitialPath}(Q, \mathrm{T_{max}})$ `// greedy initial solution`
4 **while** *Stopping condition is not met* **do**
5 $\quad p \leftarrow 1$; $\mathcal{B} \leftarrow 0$ `//` $\beta_{i,j} = 0$ `for all` $i, j \in (1, \ldots, n), i \neq j$
6 $\quad$ **while** $p \leq p_{max}$ **do**
7 $\quad\quad v' \leftarrow \text{shake}(v, p)$
8 $\quad\quad v'' \leftarrow \text{localSearch}(v', p)$
9 $\quad\quad$ **if** $\mathcal{L}(v'') \leq \mathrm{T_{max}}$ **and**
10 $\quad\quad [R(v'') > R(v)$ **or** $[R(v'') = R(v)$ **and** $\mathcal{L}(v'') < \mathcal{L}(v)]]$ **then**
11 $\quad\quad\quad v \leftarrow v''$; $p \leftarrow 1$
12 $\quad\quad$ **else**
13 $\quad\quad\quad p \leftarrow p + 1$
14 $\quad M \leftarrow \text{calculateSampling}(\mathcal{B}, \mathcal{P}, m_{exp})$
15 $\quad G \leftarrow \text{PRM*roadmapExpansion}(G, M)$
16 $\quad c_{i,j} \leftarrow$ updateRoadmapDistances(G,Q) for $\forall i, j \in (1, \ldots, n)$

In each VNS-PRM* iteration, the operators of *shake* and *local search* procedures try to increase the sum of the collected rewards. The reward contribution $\mathcal{B} = \{\beta_{i,j}\} \forall i, j \in (1, \ldots, n), i \neq j$ of each target configuration pair is stored for further focused roadmap expansion. After performing all p_{max} neighborhood operators, the number of additional samples per each target pair, $M = \{m_{i,j}\}$ for all $i, j \in (1, \ldots, n), i \neq j$, is calculated (Alg. 2 Line 14) and the roadmap is expanded (Line 15) together with the update of the shortest paths between all target configurations $c_{i,j}$ (Line 16). The number of additional samples M is based on the reward contributions $\mathcal{B}$ and sampling densities $\mathcal{P} = \{\rho_{i,j}\}$ used in the proposed sampling strategy.

1) Shake: The *shake* procedure creates a new solution v' to get the incumbent solution v from possible local optima. Its two operators ($p_{max} = 2$) tries to randomly select a part of v and alter its position within the vector, but always keep q_{σ_1} and adjust the terminal configuration q_{σ_l} to maximize l but ensure $\mathcal{L}(v) \leq \mathrm{T_{max}}$. The first **Path move** operator ($p = 1$) randomly selects a part of v and moves it to a different position. The **Path exchange** operator ($p = 2$) selects two random non-overlapping parts of v and exchanges their positions. Thus, v'

can have changed both the subset Q_l and sequence Σ to visit the configurations in Q_l.

2) Local search: The *local search* procedure tries to optimize solution v'' (initialized by v') by a sequence of simple one target operations. The employed Randomized VNS (RVNS) variant of the VNS uses randomized *local search* operators where each operator examines $|Q|^2$ simple changes of the solution vector v''. Each change is applied to v'' only if it improves the new solution w, i.e., $R(w) > R(v'')$ or decreases the path length $\mathcal{L}(w) < \mathcal{L}(v'')$ for the same reward. The **One target move** operator ($p = 1$) examines changes where a randomly selected target is moved to a different place within the solution vector. The **One target exchange** operator ($p = 2$) examines changes where two randomly selected targets are exchanged. The procedure is summarized in Alg. 3.

Algorithm 3: Local Search Procedure

Input : Q – target configurations, $\mathrm{T}_{\max}$ – budget, p – actual neighborhood number, v' – actual solution
Output: v'' – created solution, $\mathcal{B}$ – target pairs rewards

$v'' \leftarrow v'$
for $|Q|^2$ **do**
  if $p = 1$ **then** // One target move
    $w_u \leftarrow v''$ with one randomly moved target
  else // ($p = 2$) One target exchange
    $w_u \leftarrow v''$ with one randomly exchanged target
  $\mathcal{B} \leftarrow$updateTargetPairRewards($\mathcal{B}$, w_u)
  $w \leftarrow w_u$ maximize l such that w_{σ_l} is within $\mathrm{T}_{\max}$
  if $R(w) > R(v'')$ **or** $[R(w) = R(v'')$ **and** $\mathcal{L}(w) < \mathcal{L}(v'')]$ **then**
    $v'' \leftarrow w$

The local search operators firstly create a possibly unfeasible solution w_u without adjusted position of q_{σ_l} within w_u, and thus $\mathcal{L}(w_u) \nleq \mathrm{T}_{\max}$. It is because the paths connecting the configurations in w_u can be shortened by a roadmap expansion. Therefore, w_u is used for updating $\mathcal{B}$ in $\mathrm{updateTargetPairRewards}()$ where the reward of each target pair contributing to w_u is stored for the prioritization of the promising solutions in the sampling strategy of roadmap expansion. In this way, promising solutions are stored during the search over the POP combinatorial solution space to guide the expansion of the roadmap.

3) Sampling strategy: The sampling strategy of the roadmap expansion uses equidistant sampling within the ellipsoid (Alg. 1) based on the solution space search done by the randomized *local search* procedure. The update of the reward contribution $\mathcal{B}$ in $\mathrm{updateTargetPairRewards}()$ is performed for all consecutive pairs of the target configurations q_i, q_j in the solution w_u. The reward of each pair $\beta_{i,j}$ is considered to be increased by $\Delta_\beta(w_u)$ computed from the average reward per a single target in w_u, using the solution reward $R(w_u)$, multiplied by a relative budget overshoot of the solution length $\mathcal{L}(w_u)$ determined as

$$\Delta_\beta(w_u) = \frac{R(w_u)}{l-1}\left(1 - \frac{\mathcal{L}(w_u) - \mathrm{T}_{\max}}{r_o\mathrm{T}_{\max} - \mathrm{T}_{\max}}\right). \quad (2)$$

The ratio r_o is introduced to allow tuning of the overshoot. The pair reward $\beta_{i,j}$ is then updated by

$$\beta_{i,j} \mathrel{+}= \begin{cases} 0 \text{ for } \mathcal{L}(w_u) > r_o\mathrm{T}_{\max} \\ \Delta_\beta(w_u) \text{ for } \mathcal{L}(w_u) \leq r_o\mathrm{T}_{\max}, R(w_u) \leq R(v). \\ 10\Delta_\beta(w_u) \text{ for } \mathcal{L}(w_u) \leq r_o\mathrm{T}_{\max}, R(w_u) > R(v) \end{cases} \quad (3)$$

In (3), we further distinguish solutions w_u satisfying $r_o\mathrm{T}_{\max}$ with the higher reward $R(v)$ than the current best solution for which the increase of $\beta_{i,j}$ is $10\times$ higher to focus sampling of the roadmap. The roadmap expansion thus depends on the rewards $\mathcal{B}$ to focus sampling on the promising sequence of configurations. Besides, the sampling strategy is also designed to depend on its densities $\mathcal{P} = \{\rho_{i,j}\}$ to avoid adding samples to already densely sampled ellipsoids. Thus, the sampling priority $m'_{i,j}$ of each configuration pair (q_i, q_j) is proportional to the reward $\beta_{i,j}$ and inversely proportional to the sampling density $\rho_{i,j}$. This leads to disabling sampling of almost straight line paths with high density.

$$m'_{i,j} = \frac{\beta_{i,j}}{\rho_{i,j}}\ ,\ m_{i,j} = \lceil m_{exp}\frac{m'_{i,j}}{\sum_{i=1}^{i=n}\sum_{j=1}^{j=n} m'_{i,j}}\rceil \quad (4)$$

The number of samples $m_{i,j}$ added to the ellipsoid corresponding to the path between q_i and q_j is determined using (4), where m_{exp} is the number of samples intended to be added to the roadmap during each roadmap expansion.

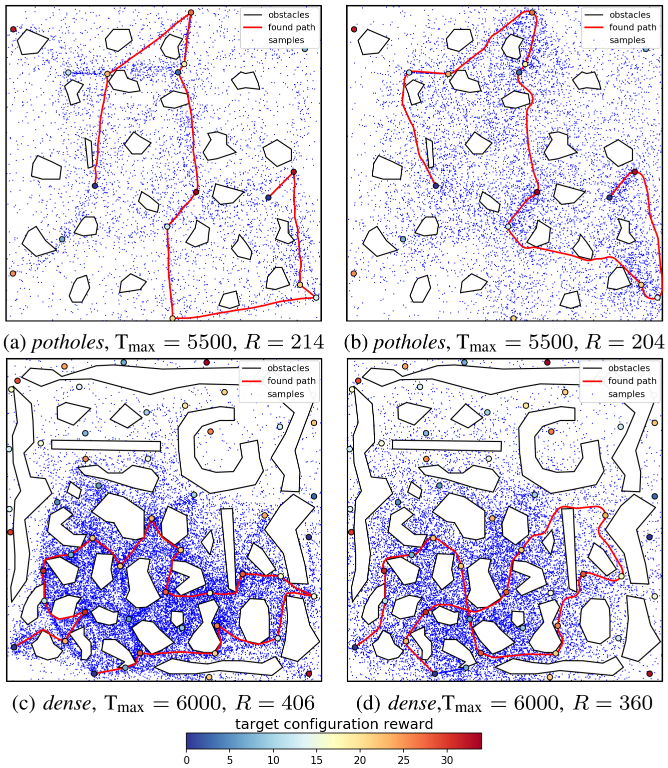
V. RESULTS

The proposed VNS-PRM* for the introduced POP is evaluated in three simulation scenarios and verified in realistic field deployment. First, the feasibility of the approach is verified for instances with a point robot $q = (x, y) \in \mathbb{R}^2$ and compared to the optimal solution found by the Integer Linear Programming (ILP) using visibility graph for the shortest paths between the target locations. Besides, the proposed online roadmap expansion is compared with the usage of a single static high-density roadmap. The VNS-PRM* is then applied to the POP with the curvature-constrained Dubins vehicle, $q = (x, y, \theta) \in SE(2)$ and compared with our implementation of [27]. Finally, the method is used for $q = (x, y, z) \in \mathbb{R}^3$ environment and further verified in a small real outdoor experiment with a hexarotor UAV.

Two different environments denoted *potholes* and *dense* with 17 and 52 target locations, respectively, and with the dimension of 2000×2000 map units are used for the evaluations. The initial roadmap is constructed with $m_{init} = 1000$ uniformly sampled configurations. The number of samples in the roadmap expansion is $m_{exp} = 50$ and the budget overshoot ratio r_o is empirically set to $r_o = 1.2$. The optimization is terminated after the maximal number of 1000 iterations, 50 iterations without improvement, or after one hour of the computational time. The proposed method[1] is implemented in C++, and all the reported results are achieved using a single core of the Intel Xenon processors cluster (2.2GHz-3.3GHz). An example of solutions found by the proposed VNS-PRM* for $q \in \mathbb{R}^2$ are depicted in Fig. 2.

The first evaluation scenario is focused on the comparison of the proposed method with the optimal solutions for $q \in \mathbb{R}^2$ that has been found using ILP OP formulation [1] in CPLEX 12.6.1 that is denoted ILP-VIS. The VNS-based solution without the PRM* is denoted VNS-VIS and both the ILP-VIS and VNS-VIS use path lengths determined from the visibility graph. The

[1]Method implementation, benchmark instances and obtained solutions are available at https://github.com/ctu-mrs/vns-prm-pop



(a) *potholes*, $\mathrm{T}_{\max} = 5500$, $R = 214$ (b) *potholes*, $\mathrm{T}_{\max} = 5500$, $R = 204$ (c) *dense*, $\mathrm{T}_{\max} = 6000$, $R = 406$ (d) *dense*, $\mathrm{T}_{\max} = 6000$, $R = 360$

Fig. 2. Example of the POP solutions for $q \in \mathbb{R}^2$ in (a) and (c), and $q \in SE(2)$ for $\rho = 60$ in (b) and (d) for both simulation testing environments.

achieved results from fifty trials of all considered instances and budget constraints $\mathrm{T}_{\max}$ are reported in Table I, where R_m and R is the maximal and average collected reward, respectively, σ is the standard deviation, $\mathcal{L}_r$ is the ratio of the average path length with respect to the particular $\mathrm{T}_{\max}$, and T is the average computational time (in seconds). The best solutions are highlighted in bold.

TABLE I
RESULTS ON THE POP INSTANCES WITH VISIBILITY GRAPH FOR $q \in \mathbb{R}^2$

Pr.	$\mathrm{T}_{\max}$	ILP-VIS		VNS-VIS		VNS-PRM*			
		R_m	T	R_m	T	R_m	$R \pm \sigma$	$\mathcal{L}_r$	T
potholes	1500	**48**	0.01	**48**	0.07	**48**	48.0± 0.0	0.87	4.0
	2500	**91**	0.06	**91**	0.11	**91**	89.1± 0.4	0.90	4.4
	3500	**143**	0.11	**143**	0.14	**143**	131.1± 8.1	0.98	9.3
	4500	**176**	0.08	**176**	0.17	168	161.6± 7.9	0.94	13.2
	5500	**214**	0.05	**214**	0.19	**214**	204.0± 8.3	0.97	17.4
	6500	**247**	0.09	**247**	0.21	245	235.8± 8.0	0.97	24.6
	7500	**270**	0.07	**270**	0.18	**270**	266.5± 4.7	0.97	24.4
	8500	**292**	0.06	**292**	0.20	**292**	292.0± 0.0	0.94	21.5
	9500	**299**	0.02	**299**	0.19	**299**	298.6± 1.7	0.93	18.6
dense	2000	**121**	1.22	**121**	1.06	**121**	117.8± 2.7	0.96	31.9
	4000	**284**	0.99	**284**	1.43	**284**	274.5± 6.4	0.98	71.8
	6000	**406**	3.71	**406**	1.70	**406**	397.9± 7.0	0.99	210.3
	8000	**522**	2.08	514	1.88	498	485.0± 9.2	0.98	264.0
	10000	**630**	16.06	618	2.32	613	568.5±17.9	0.99	489.9
	12000	**741**	0.99	718	2.51	705	667.7±20.7	0.99	937.6
	14000	**827**	0.75	803	2.20	791	745.0±22.8	0.99	1221.1
	16000	**892**	1.10	883	2.00	881	826.8±22.0	0.99	1884.0
	18000	**922**	4.11	**922**	1.97	**922**	891.1±16.8	0.99	2187.9

Although the ILP-VIS provides optimal solutions, the approach is usable only with the point robot and configuration space where the visibility graph can be used to determine the shortest collision-free paths. The heuristic VNS-VIS provides competitive results to the optimal solutions, but most importantly, the proposed VNS-PRM* provides the optimal solutions in most of the cases, except the *dense* environment which contains many obstacles. The ILP-VIS and VNS-VIS utilize precomputed visibility graph (not counted in their computational times), and therefore, the high computational requirements of VNS-PRM* are not surprising. The main advantage of the VNS-PRM* is in the applicability for different motion model, e.g., Dubins vehicle, and extendibility for more complex robot shapes. The reported results for the VNS-VIS and VNS-PRM* indicate that the inability to find the optimal solutions using VNS-PRM* is caused by the VNS part of the method as the optimal solution is not found using the shortest paths in the VNS-VIS. Nevertheless, based on the reported results, we consider the proposed approach feasible, and we further report on the impact of the proposed sampling strategy.

The online sampling strategy with the preference of sampling between target configurations of the promising solutions found by the VNS is compared with a solution found on a roadmap created only by the initial sampling, but with a high number of samples m_{init}. The evaluation is performed for the *potholes* environment and the results are reported in Table II, where VNS-Static_Roadmap denotes the variant with only initial sampling that has been considered with $m_{init} \in \{1 \times 10^4, 3 \times 10^4, 6 \times 10^4, 1 \times 10^5, 1.5 \times 10^5\}$ uniform samples in $\mathcal{C}_{free}$ without the online expansion. In addition to the maximal sum of the collected rewards R_m from fifty trials and the corresponding average computational time T in seconds, the average time of the last solution improvement of the VNS-PRM* is reported in the column T_i.

TABLE II
ONLINE SAMPLING STRATEGY VS. INITIAL SAMPLING ONLY

$\mathbf{T}_{\mathbf{max}}$	VNS-PRM*			VNS-Static_Roadmap with m_{init} samples									
				10^4		3×10^4		6×10^4		10^5		1.5×10^5	
	R_m	T	T_i	R_m	T	R_m	T	R_m	T	R_m	T	R_m	T
1500	**48**	4	1	**48**	34	**48**	92	**48**	153	**48**	223	**48**	361
2500	**91**	4	1	89	42	89	83	89	155	**91**	259	**91**	419
3500	**143**	9	4	125	43	132	78	132	177	127	299	**143**	521
4500	**168**	13	5	**168**	39	**168**	115	**168**	219	**168**	369	**168**	569
5500	**214**	17	6	204	41	204	95	**214**	222	**214**	395	**214**	721
6500	**245**	25	8	**245**	47	**245**	123	**245**	204	**245**	478	237	916
7500	**270**	24	9	**270**	45	**270**	120	**270**	292	**270**	515	**270**	818
8500	**292**	21	6	**292**	52	**292**	122	**292**	260	**292**	511	**292**	836
9500	**299**	19	7	**299**	40	**299**	119	**299**	252	**299**	519	**299**	1056

The average computational time of the VNS-PRM* solution is similar to the initial sampling with $m_{init} = 10^4$, but the time of the last solution improvement T_i is significantly lower. Therefore, the relatively high number of 50 iterations without improvement, which however causes the termination in a majority of cases, can be decreased without affecting the solution quality. The computational time of VNS-Static_Roadmap is dominated by the roadmap construction and finding the shortest paths between all target pairs using Dijkstra's algorithm. The computational requirements of VNS itself using already known shortest paths can be seen for VNS-VIS approach in Table I. The VNS-PRM* finds the best solutions in all instances while the computationally demanding high number of initial samples does not provide the best solution for all considered $\mathrm{T}_{\max}$. The average number of samples needed to find solutions using VNS-PRM* in Table II is 6678. Furthermore, the VNS-PRM* is an anytime algorithm which starts with a relatively small number of samples to quickly find a feasible solution that is then continuously improved if more computational time is available, which is shown in Fig. 3.

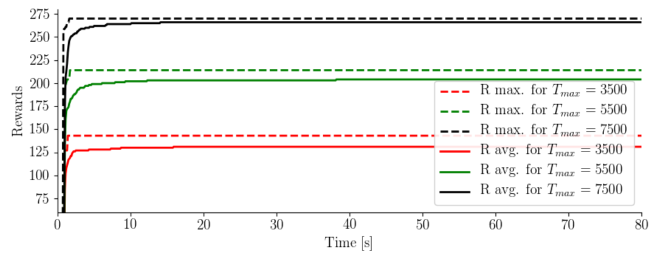


Fig. 3. Evolution of the average and maximal sum of the collected rewards for the *potholes* scenario and selected budgets T_{max}.

The proposed VNS-PRM* has been further examined for curvature-constrained planning with Dubins vehicle, configuration space $SE(2)$, and compared with our implementation of [27] for the POP. In $SE(2)$, the optimal Dubins maneuvers are used as the distance between every two configurations. Because of Dubins vehicle, each target location is considered as 12 configurations with equidistantly spread heading angle θ to allow each target location to be visited using a different vehicle heading. Only a single such sample is, however, allowed to collect the reward associated with the particular target location which transforms the problem into an instance of the Set Orienteering Problem [29]. The implementation of [27] for Dubins vehicle and the POP (denoted as the PRM-MT) uses the navigation PRM roadmap in $\mathbb{R}^2$ with 1000 samples to guide the expansion of the motion tree in $SE(2)$. Since [27] does not address the POP, the following modifications have been made: an ILP OP solver is used instead of the PC-TSP solver, a solution has to reach proximity of the terminal location, the solution length is used instead of the execution time, and the sum of the rewards of unvisited target locations is used instead of the PC-TSP penalty. The results for solved instances with the turning radius of $\rho = 60$ are reported in Table III.

TABLE III
RESULTS ON THE POP INSTANCES WITH DUBINS VEHICLE – $q \in SE(2)$

Pr.	T_{max}	PRM-MT				VNS-PRM*			
		R_m	$R \pm \sigma$	$\mathcal{L}_r$	T_i	R_m	$R \pm \sigma$	$\mathcal{L}_r$	T_i
potholes	1500	**48**	35.6±11.0	0.87	3	**48**	48.0± 0.0	0.89	5
	2500	**89**	70.5±11.0	0.91	21	**89**	89.0± 0.0	0.94	6
	3500	118	94.6±15.7	0.94	145	**127**	122.7± 4.3	0.93	56
	4500	153	107.8±27.4	0.90	263	**168**	161.8± 7.7	0.98	109
	5500	179	109.4±26.9	0.79	363	**204**	190.3±13.3	0.96	124
	6500	221	120.1±35.0	0.74	370	**242**	226.0±10.8	0.97	165
	7500	234	88.5±41.5	0.61	407	**263**	257.1± 5.9	0.95	221
	8500	167	92.4±32.9	0.54	545	**292**	281.3±11.2	0.97	103
	9500	219	91.7±43.1	0.52	552	**299**	295.2± 3.5	0.94	67
dense	2000	80	67.6± 9.5	0.90	219	**108**	107.8± 1.4	0.90	95
	4000	154	74.9±26.4	0.54	2514	**237**	225.2± 4.8	0.96	560
	6000	110	66.9±14.6	0.30	2513	**360**	339.5±10.5	0.98	812
	8000	144	34.3±32.4	0.13	2522	**472**	429.0±11.7	0.98	1065
	10000	130	9.7±21.7	0.03	3042	**564**	510.9±17.0	0.99	1363
	12000	52	2.8±10.2	0.01	3144	**646**	596.7±21.2	0.98	1827
	14000	121	8.1±23.1	0.02	2815	**719**	670.7±20.9	0.99	2356
	16000	84	3.5±13.9	0.01	2510	**806**	742.3±23.9	0.99	2136
	18000	55	8.9±16.2	0.02	2879	**839**	798.8±22.8	0.98	2488

Regarding the reported results, the VNS-PRM* outperforms the PRM-MT in both the maximal achieved rewards R_m and the average rewards R with smaller σ. The average ratio of the used budget $\mathcal{L}_r$ for the PRM-MT indicates that the method is unable to exploit the available travel budget. This is caused by uniform sampling of PRM-MT along the navigation roadmap without considering the ability of the motion tree to reach these random samples. This can be improved by generating samples according to the progress of the motion tree [30]. The average time of the last solution improvement T_i of the VNS-PRM* is also lower than for the PRM-MT in most of the cases. Low values of the collected rewards in *dense* scenarios suggest that the PRM-MT [27] struggles with narrow passages and the guidance along solutions found in static roadmaps becomes less effective for longer T_{max} with the possibility to visit more targets.

The VNS-PRM* is further verified in 3D scenario denoted as *building* that is 20×30×6 large and has seven rooms in each of the two floors, see Fig. 4. One target location is in each room with the reward in the range 5–30, thus 14 targets in the total. The upper floor is accessible only by tight windows and the robot is modeled as a cylindrical object with 0.7 diameter and 0.5 height with the configuration $q \in \mathbb{R}^3$.

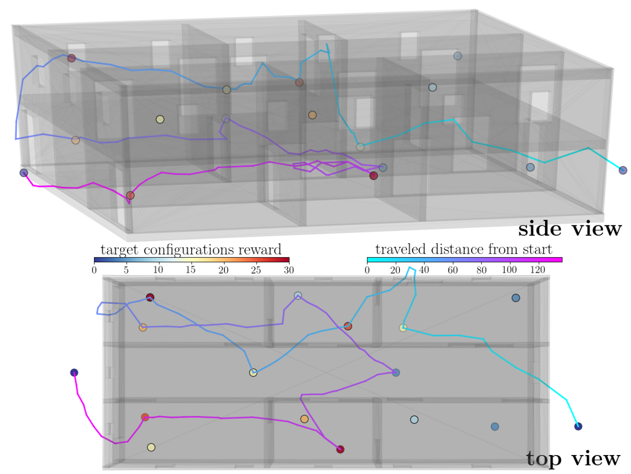


Fig. 4. Example solution of the POP in the *building* environment for $T_{max} = 140$ with the collected reward $R = 230$ and solution length of 136.3.

The computational results for the *building* scenario are depicted in Table IV, where T_{init} denotes the average time to find initial solution with the average reward R_{init} and i is the average number of the VNS-PRM* iterations.

TABLE IV
RESULTS ON THE POP INSTANCES FOR $q \in \mathbb{R}^3$

Pr.	T_{max}	VNS-PRM*						
		R_{init}	R_m	$R \pm \sigma$	$\mathcal{L}_r$	i	T_{init}	T
building	60	4.8	60	29.2±14.5	0.92	71	0.6	6.7
	80	30.5	100	76.8± 8.5	0.95	97	0.5	14.5
	100	45.4	120	101.4±20.3	0.94	108	0.6	23.5
	120	52.6	150	132.1±13.9	0.96	140	0.6	45.5
	140	73.0	175	163.2±11.0	0.96	143	0.6	52.9
	160	81.1	205	179.4±28.5	0.95	137	0.6	47.6
	180	87.2	215	203.7± 7.9	0.96	150	0.6	68.0
	200	92.0	225	215.7± 7.4	0.96	155	0.6	70.5
	220	100.1	230	224.7± 4.3	0.95	147	0.7	67.1

Table IV shows that the VNS-PRM* finds an initial solution within one second with the average solution quality of 35.3% of the best-found solution. The number of iterations i indicates that the algorithm terminates after the maximum of 50 iterations without improvement. Furthermore, the comparison of computational times for $q \in \mathbb{R}^2$, $SE(2)$, and $\mathbb{R}^3$ show the increased computational requirements of planning in $SE(2)$, which is caused by the nearest neighborhood search of the PRM* where k-d trees are not effective in $SE(2)$.

Finally, the VNS-PRM* has been experimentally verified in a small data collection mission with a hexarotor UAV. The scenario consists of three walls and four cylindrical obstacles representing the indoor- or urban-like environment, see Fig. 1 with the visualization of the results. The environment was about $9 \times 10\,\mathrm{m}$ large with ten target locations, including initial and terminal locations, with the constant altitude, and thus the VNS-PRM* search space is $\mathbb{R}^2$. The UAV is modeled as a cylindrical object with $1.4\,\mathrm{m}$ diameter and $0.5\,\mathrm{m}$ height which corresponds to $1.75\times$ enlargement of the real physical dimension of the UAV to compensate possible localization inaccuracies. The considered travel budget limit was set to $T_{max} = 25\,\mathrm{m}$ and the solution found onboard of the UAV before flight by the VNS-PRM* within $T = 8.4\,\mathrm{s}$ is $24.11\,\mathrm{m}$ long with the collected reward $R = 75$. The model predictive trajectory tracking [31] was used to precisely follow the trajectory and visit all six planned target locations.

VI. Conclusions

A novel generalization of the Orienteering Problem (OP) for robotic data collection scenarios is introduced in this paper. The problem is called Physical Orienteering Problem (POP), and it is suitable for cases where collision-free paths in environment with obstacles are required together with the maximization of collected rewards from the given target locations using the limited travel budget. The proposed solution of newly introduced POP is based on the Variable Neighborhood Search (VNS) metaheuristic for the OP that is combined with the asymptotically optimal motion planner PRM*. The proposed VNS-PRM* starts with a low-dense roadmap that is continuously expanded during the VNS-based route optimization by selecting the most promising solutions for shortening the collision-free paths and thus allowing to maximize the collected rewards. The presented results show that the proposed VNS-PRM* is a feasible and vital method and it can provide optimal solutions when compared on 2D instances with a point robot. Furthermore, the proposed roadmap expansion strategy demonstrates computational benefits in comparison to a very dense initial roadmap. The main benefit of the approach rests in the generalization of the OP for more complex configuration spaces demonstrated in a solution of the POP in $\mathbb{R}^3$ and with Dubins vehicle in $SE(2)$.

References

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

[2] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006.

[3] I. Jawhar, N. Mohamed, J. Al-Jaroodi, and S. Zhang, "A framework for using unmanned aerial vehicles for data collection in linear wireless sensor networks," *Journal of Intelligent & Robotic Systems*, vol. 74, no. 1, pp. 437–453, Apr 2014.

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

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

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

[7] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.

[8] S. M. Lavalle, "Rapidly-exploring random trees: A new tool for path planning," Iowa State University, Tech. Rep., 1998.

[9] L. E. Kavraki, P. Svestka, J. . Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, Aug 1996.

[10] M. Elbanhawi and M. Simic, "Sampling-Based Robot Motion Planning: A Review," *IEEE Access*, vol. 2, pp. 56–77, 2014.

[11] 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.

[12] R. Ramesh, Y.-S. Yoon, and M. H. Karwan, "An optimal algorithm for the orienteering tour problem," *ORSA Journal on Computing*, vol. 4, no. 2, pp. 155–165, 1992.

[13] M. Fischetti, J. J. S. González, and P. Toth, "Solving the orienteering problem through branch-and-cut," *INFORMS Journal on Computing*, vol. 10, no. 2, pp. 133–148, 1998.

[14] A. Şevkli and F. Sevilgen, "StPSO: strengthened particle swarm optimization," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 18, no. 6, pp. 1095–1114, 2010.

[15] M. Schilde, K. F. Doerner, R. F. Hartl, and G. Kiechle, "Metaheuristics for the bi-objective orienteering problem," *Swarm Intelligence*, vol. 3, no. 3, pp. 179–201, 2009.

[16] Z. Sevkli and F. E. Sevilgen, "Variable neighborhood search for the orienteering problem," in *ISCIS*, 2006, pp. 134–143.

[17] 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, July 1957.

[18] A. Bit-Monnot, R. Bailon-Ruiz, and S. Lacroix, "A local search approach to observation planning with multiple uavs," in *ICAPS*, 2018, pp. 437–445.

[19] R. Pěnička, J. Faigl, M. Saska, and P. Váňa, "Data collection planning with non-zero sensing distance for a budget and curvature constrained unmanned aerial vehicle," *Autonomous Robots*, Feb 2019.

[20] J. Faigl and R. Pěnička, "On close enough orienteering problem with dubins vehicle," in *IEEE/RSJ IROS*, 2017, pp. 5646–5652.

[21] D. Thakur, M. Likhachev, J. Keller, V. Kumar, V. Dobrokhodov, K. Jones, J. Wurz, and I. Kaminer, "Planning for opportunistic surveillance with multiple robots," in *IEEE/RSJ IROS*, 2013, pp. 5750–5757.

[22] D. Perez, P. Rohlfshagen, and S. M. Lucas, "The physical travelling salesman problem: Wcci 2012 competition," in *IEEE Congress on Evolutionary Computation*, 2012, pp. 1–8.

[23] D. Devaurs, T. Siméon, and J. Cortés, "A multi-tree extension of the transition-based rrt: Application to ordering-and-pathfinding problems in continuous cost spaces," in *IEEE/RSJ IROS*, 2014, pp. 2991–2996.

[24] J. McMahon and E. Plaku, "Autonomous underwater vehicle mine countermeasures mission planning via the physical traveling salesman problem," in *OCEANS - MTS/IEEE*, 2015, pp. 1–5.

[25] S. Edelkamp, M. Pomarlan, and E. Plaku, "Multiregion inspection by combining clustered traveling salesman tours with sampling-based motion planning," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 428–435, April 2017.

[26] J. McMahon and E. Plaku, "Mission and motion planning for autonomous underwater vehicles operating in spatially and temporally complex environments," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 4, pp. 893–912, Oct 2016.

[27] ——, "Autonomous data collection with limited time for underwater vehicles," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 112–119, Jan 2017.

[28] J. Dezert and C. Musso, "An efficient method for generating points uniformly distributed in hyperellipsoids," in *Workshop on Estimation, Tracking and Fusion:A Tribute to Yaakov Bar-Shalom*, 2001.

[29] R. Pěnička, J. Faigl, and M. Saska, "Variable neighborhood search for the set orienteering problem and its application to other orienteering problem variants," *European Journal of Operational Research*, vol. 276, no. 3, pp. 816 – 825, 2019.

[30] V. Vonásek, J. Faigl, T. Krajník, and L. Přeučil, "RRT-path–a guided rapidly exploring random tree," in *RoMoCo*, 2009, pp. 307–316.

[31] T. Báča, D. Heřt, G. Loianno, M. Saska, and V. Kumar, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *IEEE/RSJ IROS*, 2018, pp. 6753–6760.