Unsupervised Learning-based Data Collection Planning with Dubins Vehicle and Constrained Data Retrieving Time

Jindřiška Deckerová[1[0000-0001-7182-8698]] and Jan Faigl[2[0000-0002-6193-0792]]

Faculty of Electrical Engineering, Czech Technical University in Prague,
Technická 2, 166 27 Prague, Czechia
{deckejin|faiglj}@fel.cvut.cz

Abstract. In remote data collection from sampling stations, a vehicle must be within sufficient distance from a particular station for a predefined minimal time to retrieve required data from the site. The planning task is to find a cost-efficient data collection plan to retrieve data from all the stations. For a fixed-wing aerial vehicle flying with a constant forward velocity, the problem is to determine the shortest feasible path that visits every sensing site and ensure the vehicle is within a reliable communication distance from the station for a sufficient period. We propose to formulate the planning problem as a variant of the Close Enough Dubins Traveling Salesman Problem with Time Constraints (CEDTSP-TC) that is heuristically solved by unsupervised learning of the Growing Self-Organizing Array (GSOA) modified to address the constrained minimal data retrieving time. The proposed method is compared with a baseline based on a sampling-based decoupled approach, and the results support the feasibility of both proposed solvers in random instances.

Keywords: Unsupervised Learning, Growing Self-Organizing Array, Data Collection, Traveling Salesman Problem, Dubins Vehicle

1 Introduction

The studied data collection planning is motivated by the tasks where fixed-wing aerial vehicles retrieve data from sampling stations using wireless communication. The data retrieval is possible only when the vehicle is within a sufficient distance from a sampling station. Besides, the vehicle must be within the distance for a predefined minimal time to retrieve data with a limited data transfer rate. The data collection planning problem with the fixed-wing's motion constraints on the minimal flying velocity and limited vehicle turning radius can be formulated as the *Dubins Traveling Salesman Problem* (DTSP) [22], where the travel cost from one location to another location corresponds to the length of the shortest curvature-constrained (Dubins) path.

Dubins path makes the DTSP a combination of combinatorial and continuous optimization problems to determine the optimal sequence of visits to the target locations and determine the vehicle's heading angles at the locations since the length of the shortest path depends on the angles. Because of the underlying combinatorial TSP, the DTSP is also NP-hard [19]. The DTSP has been addressed by several approaches, such as [22,17,1,13,21,4,26]. Challenging sequence-dependent routing can be addressed by a decoupled approach, where the sequence of visits is determined for the relaxed heading constraints. Then, optimal headings are determined for the sequence using the Dubins path connecting the locations. Besides, the problem can be discretized into purely combinatorial problems by sampling possible heading angles into some finite locations.

In the motivational data collection, the communication range can be exploited in finding the optimal solution of visiting disk-shaped regions around the sampling stations formulated as the *DTSP with Neighborhoods* [20]. Here, in addition to the optimal vehicle heading angle at the data retrieval location, we search for the optimal locations of visits to the regions (neighborhoods). Similar to DTSP, decoupled [22,17,25,15], sampling-based [2], and direct approaches [18,26,11,12,16] have been proposed.

Although the DTSPN might yield shorter tours than using the DTSP with the centers of the disk-shaped neighborhoods, the minimal required time the vehicle spent within the station's communication range is not guaranteed. Therefore, a novel problem formulation is proposed to enable data retrieval by an uninterrupted vehicle presence within the sampling station's neighborhood for the defined minimal time. The introduced problem is studied as the *Close Enough DTSP with Time Constraints* (CEDTSP-TC) to highlight the disk-shaped neighborhoods. The problem turns into determining two waypoint locations (entering and leaving) for each region (disk), such that the waypoints are connected by a minimal length path entirely within the region to retrieve all data.

We propose to address the introduced CEDTSP-TC by unsupervised learning of the Growing Self-Organizing Array (GSOA) [6] already successfully deployed in various data collection planning problems as variants of the TSP [9,10,8,3] providing competitive solutions to existing heuristic methods in short computational times. The GSOA simultaneously determines the sequence of visits with the waypoints and headings [4] and can be relatively easy to modify to other problems, such as addressing spatially correlated measurements [7]. For the CEDTSP-TC, we need to address the minimal length path within the region.

The performance of the developed GSOA-based solution is compared with a baseline decoupled sampling-based approach Centroid-TSP, where the sequence is determined as a solution of the Euclidean TSP (ETSP) using the centers of the disk-shaped regions; then, waypoints are determined from a set of sampled locations on the border of the regions, each with a set of sampled possible heading angles. Both approaches provide competitive results, but GSOA outperforms the baseline method in computational time and solving instances with a low number of samples.

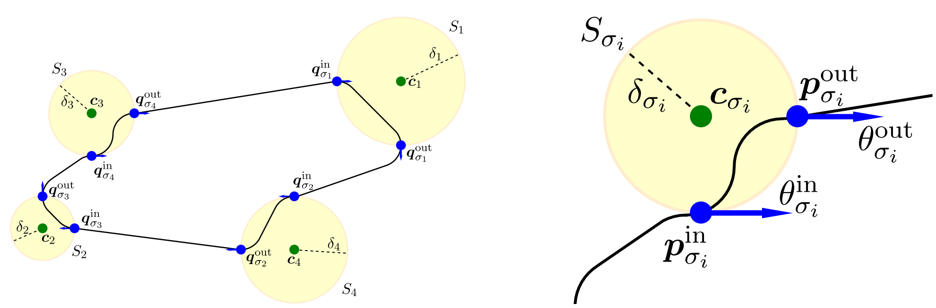
The rest of the paper is organized as follows. The problem is formally defined in Section 2. The proposed Centroid-TSP- and GSOA-based approaches are described in Section 3 and Section 4, respectively. The empirical evaluation is presented in Section 5, and concluding remarks can be found in Section 6.

2 Problem Statement

The studied CEDTSP-TC is to find the cost-efficient curvature-constrained path to visit a given set of disk-shaped regions such that the path passes the regions with at least $\mathcal{L}_{\min}$ length to remotely collect data from the sampling station located at the center of the disk. The curvature-constrained path consists of a sequence of Dubins paths feasible for Dubins vehicle that moves forward with a constant velocity v and has limited minimal turning radius ρ [5]. The vehicle motion can be described by a state equation for a control input u as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = v \begin{bmatrix} \cos\theta \\ \sin\theta \\ \frac{u}{\rho} \end{bmatrix}, \quad |u| \leq 1, \tag{1}$$

where $(x, y) \in \mathbb{R}^2$ is the vehicle's position in the plane and $\theta \in (0, 2\pi]$ denotes the vehicle's heading angle at (x, y). The state of the vehicle is thus $\boldsymbol{q} = (x, y; \theta)$ and $\boldsymbol{q}$ is from the Special Euclidean group $\boldsymbol{q} \in SE(2)$.



(a) Example of the CEDTSP-TC instance with the used notation.

(b) Dubins data collection path path in the disk S_{σ_i}.

Fig. 1. An instance of the CEDTSP-TC with $n = 4$ target regions $\mathcal{S}$ depicted in yellow (1a) together with its solution. The solution is the curvature-constrained path (black) visiting $\mathcal{S}$ in the order $\Sigma = (1, 4, 2, 3)$ at the configurations $Q = \{\boldsymbol{q}^{\text{in}}_{\sigma_1}, \boldsymbol{q}^{\text{out}}_{\sigma_1}, \ldots, \boldsymbol{q}^{\text{in}}_{\sigma_n}, \boldsymbol{q}^{\text{out}}_{\sigma_n}\}$ (blue). The right subfigure (1b) shows a detail of Dubins path within a region with the input configuration $\boldsymbol{q}^{\text{in}}_{\sigma_i} = (\boldsymbol{q}^{\text{in}}_{\sigma_i}; \theta^{\text{in}}_{\sigma_i})$ and output configuration $\boldsymbol{q}^{\text{out}}_{\sigma_i} = (\boldsymbol{q}^{\text{out}}_{\sigma_i}; \theta^{\text{out}}_{\sigma_i})$ that consists of points of visits $\boldsymbol{p}^{\text{in}}_{\sigma_i}$ and $\boldsymbol{p}^{\text{out}}_{\sigma_i}$ and corresponding vehicle heading angles $\theta^{\text{in}}_{\sigma_i}$ and $\theta^{\text{out}}_{\sigma_i}$, respectively.

Dubins path between two configurations $\boldsymbol{q}_i$ and $\boldsymbol{q}_j$ is the shortest path satisfying constraints of Dubins vehicle (1). The path can be found as a closed-form solution [5], and it consists of up to three segments of a straight line (S) and circular arc (C) with the radius ρ. The arc can be a left (L) and right (R) turn that gives up six possible combinations: LSL, LSR, RSL, RSR, LRL, and RLR. The length of Dubins path is denoted $\mathcal{L}(\boldsymbol{q}_i, \boldsymbol{q}_j)$.

The CEDTSP-TC formulation follows the previous work [8] and is to visit a given set of n disk-shaped regions $\mathcal{S} = \{S_1, \ldots, S_n\}$, each $S_i \in \mathbb{R}^2$ defined by its center $\boldsymbol{c}_i \in \mathbb{R}^2$ and radius $\delta_i > 0$. However, the data collection path is requested to pass each region's S_i with the minimal length $\mathcal{L}_{\min}$, and such a part of the path is entirely with the region. Then, the solution of the CEDTSP-TC is a sequence $\Sigma = (\sigma_1, \ldots, \sigma_n)$, $\sigma_i \neq \sigma_j$, for $i \neq j$, of visits to $\mathcal{S}$ and a set of configurations Q denoting entering and leaving configurations of the path to visit each region. For a region S_i, the entering configuration is denoted $\boldsymbol{q}_i^{\text{in}}$ and leaving configuration $\boldsymbol{q}_i^{\text{out}}$. The path between the entering and leaving configurations is at least $\mathcal{L}_{\min}$ long, $\mathcal{L}(\boldsymbol{q}_i^{\text{in}}, \boldsymbol{q}_i^{\text{out}}) \geq \mathcal{L}_{\min}$. The set Q can be defined as $Q = \{\boldsymbol{q}_1^{\text{in}}, \boldsymbol{q}_1^{\text{out}}, \ldots, \boldsymbol{q}_n^{\text{in}}, \boldsymbol{q}_n^{\text{out}}\}$. CEDTSP-TC is formally defined as Problem 1, and a problem instance with the used notation is depicted in Fig. 1.

Problem 1 (Close Enough DTSP with Time Constraints (CEDTSP-TC)).

$$\underset{\Sigma, Q}{\text{minimize}} \quad \mathcal{L}(\boldsymbol{q}_{\sigma_n}^{\text{in}}, \boldsymbol{q}_{\sigma_n}^{\text{out}}) + \mathcal{L}(\boldsymbol{q}_{\sigma_n}^{\text{out}}, \boldsymbol{q}_{\sigma_1}^{\text{in}}) + \sum_{i=1}^{n-1} \Big(\mathcal{L}(\boldsymbol{q}_{\sigma_i}^{\text{in}}, \boldsymbol{q}_{\sigma_i}^{\text{out}}) + \mathcal{L}(\boldsymbol{q}_{\sigma_i}^{\text{out}}, \boldsymbol{q}_{\sigma_{i+1}}^{\text{in}}) \Big) \tag{2}$$

$$\text{s.t.} \quad \Sigma = (\sigma_1, \ldots, \sigma_n),\ 1 \leq \sigma_i \leq n,\ \sigma_i \neq \sigma_j \text{ for } i \neq j \tag{3}$$

$$Q = \{\boldsymbol{q}_1^{\text{in}}, \boldsymbol{q}_1^{\text{out}}, \ldots, \boldsymbol{q}_n^{\text{in}}, \boldsymbol{q}_n^{\text{out}}\} \tag{4}$$

$$\mathcal{L}(\boldsymbol{q}_i^{\text{in}}, \boldsymbol{q}_i^{\text{out}}) \geq \mathcal{L}_{\min},\ 1 \leq i \leq n \tag{5}$$

$$\boldsymbol{q}_i^{\text{in}}, \boldsymbol{q}_i^{\text{out}} \in SE(2),\ 1 \leq i \leq n \tag{6}$$

$$\boldsymbol{q}_i^{\text{in}} = (x_i^{\text{in}}, y_i^{\text{in}}, \theta_i^{\text{in}}),\ \boldsymbol{q}_i^{\text{out}} = (x_i^{\text{out}}, y_i^{\text{out}}, \theta_i^{\text{out}}), \tag{7}$$

$$\big\|(x_i^{\text{in}}, y_i^{\text{in}}) - \boldsymbol{c}_i\big\| \leq \delta_i,\ \big\|(x_i^{\text{out}}, y_i^{\text{out}}) - \boldsymbol{c}_i\big\| \leq \delta_i,\ 1 \leq i \leq n \tag{8}$$

3 Decoupled Sampling-based Solution

The decoupled sampling-based approach Centroid-TSP decomposes the problem into the (i) determination of sequence Σ and (ii) configurations Q. The sequence of visits Σ to $\mathcal{S}$ is determined from a solution of the ETSP instance with centers of the regions and Euclidean distance connecting them, for example, using the available GLKH solver [14]. The configurations Q are determined by sampling each S_i at the region border uniformly to ω locations, and for each location, ω heading angles are uniformly sampled. Thus, each S_i is sampled into $\omega \times \omega$ configurations $\mathbf{Q}_i = \{\boldsymbol{q}_1^i, \ldots, \boldsymbol{q}_{\omega\times\omega}^i\}$. The number of samples ω denotes the sampling resolution, and increasing ω might lead to an improved solution.

For a sequence Σ and a set of sampled configurations Q, an oriented search graph with $2n$ layers can be constructed as depicted in Fig. 2. The edges between the nodes of consecutive layers denote Dubins path connecting the associated configurations with the cost corresponding to the length of the path. If the path connecting the entering and leaving configuration of the region is shorter than $\mathcal{L}_{\min}$, the edge does not connect the corresponding nodes, and its cost is set to ∞. The particular solution Q is found by finding the shortest path in the graph using the dynamic program, whose complexity can be bounded by $O(2n\omega^6)$ [24]. Hence, the solution of the CEDTSP-TC is determined as a solution (Σ, Q) of its discretized variant using ω^2 samples.

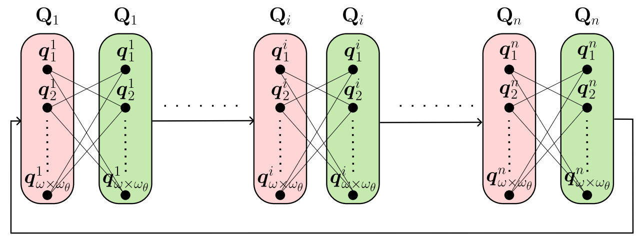


Fig. 2. An oriented search graph with $2n$ layers for n target regions of the CEDTSP-TC. The red layers correspond to the regions' input configurations and the green layers to the output configurations.

4 Growing Self-Organizing Array for the CEDTSP-TC

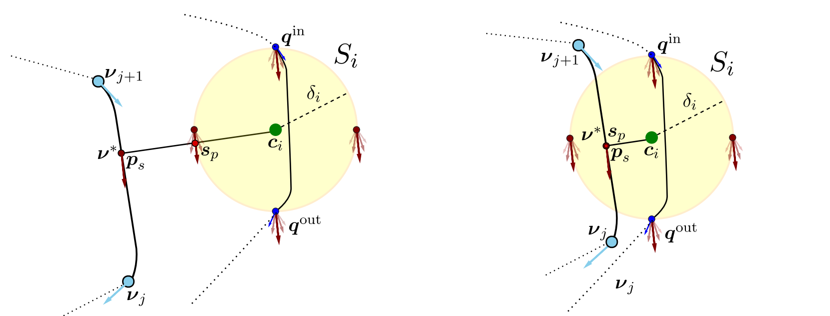
The proposed method to the CEDTSP-TC is based on adopting the unsupervised learning-based heuristic for routing problems GSOA [6], already applied to Dubins routing problems in [8]. Therefore, only the GSOA summary is presented with a focus on the modifications needed to address the CEDTSP-TC. Two variants of the GSOA applications to address the non-Euclidean problem are proposed to demonstrate further prospective modifications. The learning procedure is summarized in Algorithm 1 and works as follows.

Algorithm 1: GSOA for the CEDTSP-TC

Input: $S = \{S_1, \dots, S_n\}$ – a set of regions, $\mathcal{L}_{\min}$ – minimal path length.
Params. : $G = 10$ – the learning gain; $\alpha = 0.0005$ – the gain decreasing rate; $\mu = 0.6$ – the learning rate; and $i_{\max} = \min(120, 1/\alpha)$ – the no. of learning epochs; that are set as suggested in [6].
Output: (Σ, Q) – found a solution.

$\mathcal{N} \leftarrow \{\nu_1\}, i \leftarrow 0$ `// Initialize with geometric center`
while $i \le i_{\max}$ **do**
  foreach S_i *in random permutation of* $\mathcal{S}$ **do**
    Select winner node ν^* of $\mathcal{N}$ for S_i
    Insert ν^* into $\mathcal{N}$
    foreach $\nu \in \mathcal{N}$ *with* d *distance from* ν^**, where* $0 \le d \le 0.2M$ **do**
      Adapt ν toward $\nu^*.\boldsymbol{s}_p$ according to (10) with (11)
  Remove nodes from $\mathcal{N}$ from the previous epoch
  $i \leftarrow i + 1$, $G \leftarrow (1 - i\alpha)G$
  Construct (Σ', Q') by traversing $\mathcal{N}$ from the associated $\boldsymbol{q}_i^{\text{in}}$ and $\boldsymbol{q}_i^{\text{out}}$
  $(\Sigma, Q) \leftarrow (\Sigma', Q')$ **if** (Σ', Q') is shorter than (Σ, Q)
return (Σ, Q)

The GSOA is an iterative procedure where the solution is encoded as an array of nodes $\mathcal{N} = (\nu_1, \dots, \nu_M)$. Each node ν_j is associated with its location $\boldsymbol{\nu}_j \in \mathbb{R}^2$ and heading $\theta_s \in [0, 2\pi)$, the target region $S_{\sigma_i} \in \mathcal{S}$, the waypoint location $\boldsymbol{s}_p \in \mathbb{R}^2$, $\|\boldsymbol{s}_p - \boldsymbol{c}_{\sigma_i}\| \le \delta_{\sigma_i}$, at which the target is visited, and further with the input and output configurations $\boldsymbol{q}^{\text{in}}_{\sigma_i}$ and $\boldsymbol{q}^{\text{out}}_{\sigma_i}$, respectively, where $1 \leq \sigma_i \leq n$ denotes the label of the particular region for the sequence given by the array of nodes. Starting with a single node in the array $\mathcal{N} = \{\nu_1\}$, the node is initialized to the geometric center of the instance with zero heading. For a single iteration (a learning epoch), new nodes are added into $\mathcal{N}$ for each target that is selected randomly from $\mathcal{S}$ to avoid location minima. The node is added as the most suitable for the particular target (the closest point of the path represented by $\mathcal{N}$) and adapted toward the region. The added nodes have associated regions and locations of visits. Therefore, all the previous nodes can be removed from $\mathcal{N}$, and a solution can be retrieved from the array after every epoch. The adaptation can be repeated until the maximum number of iterations $i_{\max}$ is reached.



(a) Winner node $\boldsymbol{\nu}^*$ outside of S_i. (b) Winner node $\boldsymbol{\nu}^*$ inside of S_i.

Fig. 3. Determination of the winner node ν^* for the target region S_i at $\boldsymbol{\nu}^*$ with the points $\boldsymbol{p}_s$ and $\boldsymbol{s}_p$, and configurations $\boldsymbol{q}^{\text{in}}$ and $\boldsymbol{q}^{\text{out}}$. Dubins path between ν_j and ν_{j+1} can be used to sample the heading angles during the adaptation.

A new node added to $\mathcal{N}$ is called the *winner node* ν^* as it is determined for the particular target S_i as the closest point of the path represented by $\mathcal{N}$ to S_i as visualized in Fig. 3. Approximation of Dubins tour can be constructed from the array $\mathcal{N}$ with M nodes using two consecutive nodes of $\mathcal{N}$ to determine Dubins path (ν_j, ν_{j+1}), $\nu_{M+1} = \nu_1$. By iterating over all such Dubins paths, we can determine $\boldsymbol{p}_s$ together with the corresponding heading θ_s of the particular Dubins path as the closest point to S_i centered at $\boldsymbol{c}_i$. The location of the new node ν^* is set to $\boldsymbol{p}_s$, and its heading can be θ_s. The node ν^* is added to $\mathcal{N}$ between the corresponding nodes ν_j and ν_{j+1}, and declared to be the current winner node. Besides, if $\boldsymbol{p}_s$ is inside S_i, we can set the associated waypoint $\boldsymbol{s}_p$ to $\boldsymbol{p}_s$. Otherwise, $\boldsymbol{s}_p$ of the data collection path visits to S_i is determined as the intersection of S_i border and segment $(\boldsymbol{p}_s, \boldsymbol{c}_i)$ as depicted in Fig. 3a.

Since each region needs to be visited with at least $\mathcal{L}_{\min}$ length path entirely in the region, the input and output configurations $\boldsymbol{q}^{\text{in}}_i$ and $\boldsymbol{q}^{\text{out}}_i$, respectively, need to be determined. For the initial employment of GSOA, denoted GSOA-naive, we can utilize a similar approach to the Centroid-TSP and examine sampled configurations $Q = \{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{\omega\times\omega}\}$, where ω denotes the number of sampled headings at each possible input and output locations. For simplicity, we can utilize the uniform samples as in Section 3. Then, $\boldsymbol{q}^{\text{in}}_i$ and $\boldsymbol{q}^{\text{out}}_i$ are determined as the pair that minimizes the path length from the node ν_j to ν_{j+1} as

$$\boldsymbol{q}^{\text{in}}, \boldsymbol{q}^{\text{out}} = \operatorname*{arg\,min}_{\substack{\boldsymbol{q}_1, \boldsymbol{q}_2 \in Q \\ \mathcal{L}(\boldsymbol{q}_1, \boldsymbol{q}_2) \geq \mathcal{L}_{\min}}} \mathcal{L}(\nu_j.\boldsymbol{q}^{\text{out}}, \boldsymbol{q}_1) + \mathcal{L}(\boldsymbol{q}_1, \boldsymbol{q}_2) + \mathcal{L}(\boldsymbol{q}_2, \nu_{j+1}.\boldsymbol{q}^{\text{in}}), \quad (9)$$

where $\mathcal{L}(\cdot, \cdot)$ denotes the length of Dubins path, and the path from $\boldsymbol{q}_1$ to $\boldsymbol{q}_2$ is entirely inside the region, and its length is at least $\mathcal{L}_{\min}$.

In the naive approach of GSOA-naive, Q is uniformly sampled. However, we can exploit the exploration capability of GSOA by using the determined heading angle θ_s of the winner node and sample k heading angles locally in the sampling interval $(\theta_s - \varepsilon, \theta_s + \varepsilon)$, where ε denotes the sampling range, and $k \leq \omega$, see Fig. 3. Besides, θ_s is determined as the vehicle heading angle at the midpoint of Dubins path from ν_j to ν_{j+1}. Then, $\boldsymbol{q}_i^{\text{in}}$ and $\boldsymbol{q}_i^{\text{out}}$ are determined according to (9).

Once the winner node ν^* is added into $\mathcal{N}$, it is adapted with its neighboring nodes that are no more than $0.2M$ nodes apart toward S_i, but only if $\boldsymbol{p}_s$ is not already inside of S_i, i.e., $\boldsymbol{p}_s \neq \boldsymbol{s}_p$, see Fig. 3b. Note that the size of the neighboring nodes $0.2M$ denotes the activation bubble employed in one of the earliest unsupervised learning approaches to the TSP [23]. The adaptation adjusts the node's location $\boldsymbol{\nu}$ toward the waypoint $\boldsymbol{s}_p$ of ν^* as of

$$\boldsymbol{\nu} \leftarrow \boldsymbol{\nu} + \mu\, f(G, d)(\boldsymbol{p}_s - \boldsymbol{\nu}), \quad (10)$$

where μ is the learning rate, G is the learning gain, d is the number of nodes between the adapted node ν and winner node ν^*. The neighboring function $f(G, d)$ [12] influences the power of adaptation as of

$$f(G, d) = \begin{cases} e^{-\frac{d^2}{G^2}} & \text{if } d < 0.2M \\ 0 & \text{otherwise} \end{cases}. \quad (11)$$

Since n new nodes are added to $\mathcal{N}$ in each learning epoch, the nodes from the previous epoch are removed. Besides, the learning rate is updated to $\mu \leftarrow (1 - i\alpha)\mu$, where α is the gain decreasing rate, and i is the current number of the performed learning epoch. A solution of the CEDTSP-TC can be obtained after each learning epoch by traversing the array $\mathcal{N}$ and using the associated $\boldsymbol{q}^{\text{in}}$ and $\boldsymbol{q}^{\text{out}}$. The best solution found so far can be maintained during the learning.

We follow the parameterization of GSOA as reported in [6] that is selected based on an empirical evaluation. The learning gain G is set to 10, the gain decreasing rate α is set to 0.0005, and the learning rate μ is set to 0.6. The maximal number of learning epochs is set as the smaller value of $i_{\max}$ ($i_{\max} = 120$) or $1/\alpha$ to ensure G is always above zero.

5 Empirical Evaluation

The proposed solvers (Centroid-TSP and GSOA) to the introduced CEDTSP-TC have been empirically studied in solving 30 randomly generated instances with $\delta \geq \mathcal{L}_{\min}$. Each instance is denoted `random_`n`_`$[a-e]$, where n corresponds to the number of target regions $n \in \{5, 6, 7, 8, 9, 10\}$, and $[a-e]$ denotes particular instances with the same n. The turning radius is set to $\rho = 1.0$, the resolution ω to $\{2, 4, 8, 16, 32\}$, the minimal required length to $\mathcal{L}_{\min} = 0.5$, such as there exists at least one configuration that follows (5). Besides, an ablation study is performed by comparing the performance of the GSOA with that of the GSOA-naive to show the benefits of online sampling.

The solution quality is measured using *relative length* $\mathcal{L}_{\mathrm{rel}}$ of the solution cost found by a particular solver for a particular instance, $\mathcal{L}_{\mathrm{rel}} = (\mathcal{L}/\mathcal{L}^*)$, where $\mathcal{L}^*$ is the best solution cost found among all examined solvers for the particular instance, and $\mathcal{L}$ is the best solution length found for the particular instance among all performed trials of the particular solver. The Centroid-TSP solver employs the GLKH solver [14]. Since the GSOA is a stochastic algorithm, the number of performed trials per instance is set to 10. The number of heading angle samples in the GSOA-naive is ω as in the Centroid-TSP. However, the GSOA uses $k = 4$ samples if $\omega < 8$, otherwise $k = 16$, for the sampling range $\varepsilon = \pi/8$ if $\omega < 8$, otherwise $\varepsilon = \pi/2$.

The solvers are executed using the `Intel® Core™ i9-13900K` and the computational requirements are reported as the average required computational time $\mathrm{T}_{\mathrm{CPU}}$ (in seconds) to obtain a single solution of a particular instance by the solver. A summary of the results is depicted in Fig. 4.

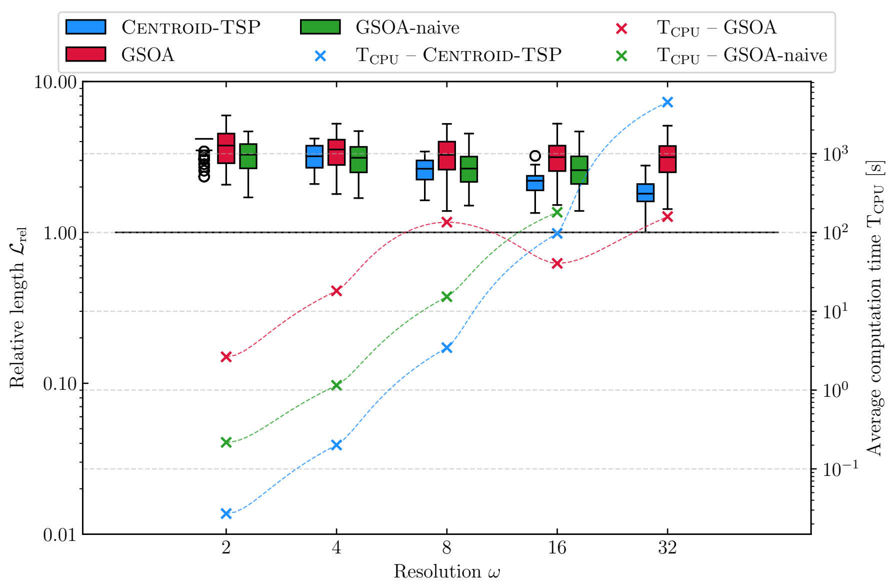


Fig. 4. Effect of resolution ω on the solution length $\mathcal{L}_{\mathrm{rel}}$ and computation time $\mathrm{T}_{\mathrm{CPU}}$ [s] depicted as the five-point summary.

The results indicate that the performance of the GSOA is competitive with the Centroid-TSP. For solving instances with low resolutions ω, the GSOA is more demanding than Centroid-TSP; however, GSOA benefits from online sampling in comparison to GSOA-naive and Centroid-TSP for a dense resolution above 16. Possible solution improvements of the GSOA compared to the GSOA-naive are not significant, but GSOA-naive fails to find solutions for $\omega > 16$, while GSOA is less demanding, which might be further improved.

6 Conclusion

A novel data collection planning problem with Dubins vehicle has been introduced as the Close Enough DTSP with Time Constraints (CEDTSP-TC). It is motivated by the practical need to retrieve data with limited data transfer speed, which requires the data collection vehicle to be within the communication range for a minimum data collection time. Two solution approaches are proposed: the decoupled sampling-based method and unsupervised learning of the GSOA. The reported results support the feasibility of both approaches. Although the developed GSOA heuristic provides competitive results with less computational requirements than the decoupled method, the ablation study indicates further improvements might be possible, which is a subject of our future work.

Acknowledgments. The work was supported by the Czech Science Foundation (GAČR) under research project No. 22-05762S.

References

1. Babel, L.: New heuristic algorithms for the dubins traveling salesman problem. Journal of Heuristics **26**(4), 503–530 (2020). DOI: 10.1007/s10732-020-09440-2
2. Cohen, I., Epstein, C., Isaiah, P., Kuzi, S., Shima, T.: Discretization-based and look-ahead algorithms for the dubins traveling salesperson problem. IEEE Transactions on Automation Science and Engineering **14**(1), 383–390 (2017). DOI: 10.1109/TASE.2016.2602385
3. Deckerová, J., Faigl, J., Krátký, V.: Traveling Salesman Problem with neighborhoods on a sphere in reflectance transformation imaging scenarios. Expert Systems with Applications **198**, 116814 (2022). DOI: 10.1016/j.eswa.2022.116814
4. Drchal, J., Faigl, J., Váňa, P.: WiSM: Windowing surrogate model for evaluation of curvature-constrained tours with dubins vehicle. IEEE Transactions on Cybernetics **52**(2), 1302–1311 (2022). DOI: 10.1109/TCYB.2020.3000465
5. Dubins, L.E.: On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents. American Journal of Mathematics **79**, 497 (1957). DOI: 10.2307/2372560
6. Faigl, J.: GSOA: Growing Self-Organizing Array - Unsupervised learning for the Close-Enough Traveling Salesman Problem and other routing problems. Neurocomputing **312**, 120–134 (2018). DOI: 10.1016/j.neucom.2018.05.079
7. Faigl, J.: Data collection path planning with spatially correlated measurements using growing self-organizing array. Applied Soft Computing **75**, 130–147 (2019). DOI: 10.1016/j.asoc.2018.11.005
8. Faigl, J.: Unsupervised learning-based solution of the Close Enough Dubins Orienteering Problem. Neural Computing and Applications **24**(32), 18193–18211 (2020). DOI: 10.1007/s00521-019-04222-9
9. Faigl, J., Deckerová, J.: On unsupervised learning based multi-goal path planning for visiting 3d regions. In: International Conference on Robotics and Artificial Intelligence (ICRAI). pp. 45–50 (2018). DOI: 10.1145/3297097.3297099
10. Faigl, J., Deckerová, J., Váňa, P.: Fast heuristics for the 3D multi-goal path planning based on the generalized traveling salesman problem with neighborhoods. IEEE Robotics and Automation Letters **4**, 2439–2446 (2019). DOI: 10.1109/LRA.2019.2900507

11. Faigl, J., Váňa, P.: Self-organizing map for the curvature-constrained traveling salesman problem. In: International Conference on Artificial Neural Networks (ICANN). pp. 497–505 (2016). DOI: 10.1007/978-3-319-44781-0_59
12. Faigl, J., Váňa, P.: Unsupervised learning for surveillance planning with team of aerial vehicles. In: International Joint Conference on Neural Networks (IJCNN). pp. 4340–4347 (2017). DOI: 10.1109/IJCNN.2017.7966405
13. Faigl, J., Váňa, P., Drchal, J.: Fast sequence rejection for multi-goal planning with dubins vehicle. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 6773–6780 (2020). DOI: 10.1109/IROS45743.2020.9340644
14. Helsgaun, K.: GLKH (2013), `http://akira.ruc.dk/~keld/research/GLKH/`, [cited on 2024-Jan-8]
15. Isaacs, J.T., Klein, D.J., Hespanha, J.P.: Algorithms for the traveling salesman problem with neighborhoods involving a dubins vehicle. In: American Control Conference. pp. 1704–1709 (2011). DOI: 10.1109/ACC.2011.5991501
16. Macharet, D.G., Campos, M.F.M.: A survey on routing problems and robotic systems. Robotica **36**(12), 1781–1803 (2018). DOI: 10.1017/S0263574718000735
17. Macharet, D.G., Neto, A.A., da Camara Neto, V.F., Campos, M.F.: Nonholonomic path planning optimization for Dubins' vehicles. In: International Conference on Robotics and Automation (ICRA). pp. 4208–4213 (2011). DOI: 10.1109/ICRA.2011.5980239
18. Macharet, D.G., Neto, A.A., da Camara Neto, V.F., Campos, M.F.: Data gathering tour optimization for Dubins' vehicles. In: Congress on Evolutionary Computation (CEC). pp. 1–8. IEEE (2012). DOI: 10.1109/CEC.2012.6256477
19. Ny, J., Feron, E., Frazzoli, E.: On the Dubins Traveling Salesman Problem. IEEE Transactions on Automatic Control **57**(1), 265–270 (2012). DOI: 10.1109/TAC.2011.2166311
20. Obermeyer, K.: Path planning for a uav performing reconnaissance of static ground targets in terrain. In: 2009 AIAA Guidance, Navigation, and Control Conference. pp. 1–11 (2012). DOI: 10.2514/6.2009-5888
21. Obermeyer, K., Oberlin, P., Darbha, S.: Sampling-based roadmap methods for a visual reconnaissance UAV. In: AIAA Guidance, Navigation, and Control Conference. p. 7568 (2010). DOI: 10.2514/6.2010-7568
22. Savla, K., Frazzoli, E., Bullo, F.: On the point-to-point and traveling salesperson problems for Dubins' vehicle. In: American Control Conference. pp. 786–791 vol. 2 (2005). DOI: 10.1109/ACC.2005.1470055
23. Somhom, S., Modares, A., Enkawa, T.: A self-organising model for the travelling salesman problem. Journal of the Operational Research Society **48**(9), 919–928 (1997). DOI: 10.2307/3010491
24. Váňa, P., Faigl, J.: Optimal Solution of the Generalized Dubins Interval Problem Finding the Shortest Curvature-constrained Path Through a Set of Regions. Autonomous Robots **44**(7), 1359–1376 (2020). DOI: 10.1007/s10514-020-09932-x
25. Yu, X., Hung, J.: A genetic algorithm for the Dubins Traveling Salesman Problem. In: International Symposium on Industrial Electronics. pp. 1256–1261. IEEE (2012). DOI: 10.1109/ISIE.2012.6237270
26. Zhang, X., Chen, J., Xin, B., Peng, Z.: A memetic algorithm for path planning of curvature-constrained UAVs performing surveillance of multiple ground targets. Chinese Journal of Aeronautics **27**(3), 622–633 (2014). DOI: 10.1016/j.cja.2014.04.024