Self-Organizing Map for Data Collection Planning in Persistent Monitoring with Spatial Correlations

Jan Faigl and Petr Váňa
Czech Technical University
Faculty of Electrical Engineering
Technicka 2, 166 27, Prague, Czech Republic
Email: faiglj@fel.cvut.cz

***Abstract*—This paper introduces an extension of the unsupervised learning method to solve data collection planning problems where particular sensor measurements can be spatially correlated. The problem is motivated by monitoring tasks formulated as the Prize-Collecting Traveling Salesman Problem with Neighborhoods (PC-TSPN). A solution of the PC-TSPN consists of a selection of important sensors, determination of the locations to read data from these sensors, and finding the shortest path to visit the locations. The solution cost is defined as a sum of the travel cost and penalty characterizing additional cost associated to sensors from which data are not retrieved. The penalty represents importance of particular sensor measurements to the quality of the model and existing solutions assume the penalties are constant values. However, for spatially close sensor locations, data from one sensor may contain also information about nearby locations and thus, its penalty depends on locations selected for data collection. The proposed generalization of the PC-TSPN solver allows to consider spatial correlations of sensor measurements and the proposed approach provides better solutions than the previous algorithm with fixed penalties.**

I. Introduction

Data collection planning is a problem arising from environment monitoring where it is necessary to efficiently collect data from pre-deployed sensor fields in large environments [1] by a mobile data sink [2]. Having a set of sensor locations, the problem of planning a cost efficient path can be formulated as the combinatorial optimization the Traveling Salesman Problem (TSP) [3], which stands to find a shortest closed path visiting each locations exactly once and return to the origin location. It is known the TSP is NP-hard, unless P=NP [4].

However, a robotic vehicle can be equipped with a wireless communication device to retrieve data from sensors remotely and thus, it may not be necessary to precisely visit the sensor locations and data can be read within the specified communication range ρ [5]. Therefore a disk with radius ρ can be used to form a neighborhood of each sensor, and data from the particular sensor can be retrieved by a robot that visits the sensor neighborhood (disk). This extension leads to a generalization of the TSP called the Traveling Salesman Problem with Neighborhoods (TSPN). Since the TSP is a special case of the TSPN where the neighborhood is a single point, also the TSPN is NP-hard. The optimal approach based on the Mixed Integer Non-linear Programming formulation is too computationally demanding [6], and therefore, soft-computing approaches based on genetic algorithm [5] and self-organizing maps [7] have been proposed.

978-1-5090-1897-0/16/$31.00 ©2016 IEEE

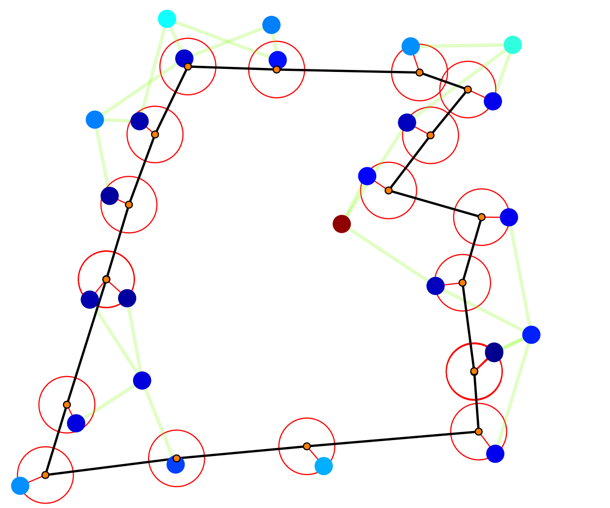


Fig. 1. Example of solution of the PC-TSPN with spatial correlations between measurements at the sensor locations. Color disks are the sensor locations, the color denotes importance of the data (high is red and low is blue). The black straight line segments denote the data collection path. The path points at which data are collected from the sensors are visualized as small orange disks and the red circle denotes a communication radius with the red straight line segment connected to the particular sensor. The light green segments represent spatial correlations between the sensor measurements, i.e., collected data from the particular sensor in the red circle also contain a partial information about the data that can be read from the sensor connected by the green segment.

In addition to the data collection within the communication radius, another important practical aspect of the data collection planning is related to the desired model of the studied phenomena for which individual sensors may provide a different contribution. The importance of the provided data by an individual sensor can be modeled as an additional penalty to the solution cost if data are not retrieved from the sensor. Then, it may be suitable to ignore sensors providing less important data and thus, decrease the overall solution cost. This aspect can be addressed in the Prize-Collecting Traveling Salesman Problem (PC-TSP) introduced in [8]. A solution of the PC-TSP combines a selection of the most suitable locations together with determination of the optimal solution of the TSP for the selected locations such that the sum of the tour length and sum of the penalties of not visited locations is minimal.

A combination of the TSPN and PC-TSP in the *Prize-Collecting Traveling Salesman Problem with Neighborhoods* (PC-TSPN) as a suitable formulation for data collection planning has been introduced in [9]. The proposed solution of the PC-TSPN is based on heuristic determination of locations to be visited and a consecutive solution of the ordinary TSP. In [10], we propose a unifying planning approach based on self-organizing map (SOM) [7] to address the whole class of problem formulations for the data collection planning as the PC-TSP, TSPN, and the PC-TSPN, which provides significantly better solutions than the heuristic algorithms [9]. Despite of these results, all the existing approaches for the PC-TSPN rely on fixed penalties (rewards) associated to not visited sensor locations, regardless data from a spatially close sensor are collected or not.

In this paper, the problem of spatially correlated measurements is addressed by a generalization of the existing SOM-based algorithm for the PC-TSPN [10]. Probably the most similar problem with correlated measurements and data collection path is the Correlated Orienteering Problem (COP) introduced in [11], where the problem is to determine a tour that maximizes the utility while its length is below the given constraint. Contrary to [11], the herein proposed solution is targeted to data collection from sensor networks considering a non-zero communication radius and the problem is addressed as a variant of the PC-TSPN to find the cost efficient solution and not to maximize the sum of rewards under a given travel budget as in the Orienteering Problem.

Although the proposed deployment of the existing SOM for the PC-TSPN [10] to the problem with correlated sensor measurements seems to be straightforward, to the best of our knowledge, there is no approach in literature that addresses the proposed problem. Therefore, the main contributions of this paper are considered in the introduction of the model of the spatial correlations to the PC-TSPN and the proposed extension of the SOM method to this generalized problem.

The paper is organized as follows. The PC-TSPN is formally introduced in the next section together with the model of spatial correlation of the penalty values associated to the individual sensors. The proposed approach to the generalized PC-TSPN with spatially correlated penalties is presented in Section III. Evaluation results are reported in Section IV and concluding remarks with future work are in Section V.

II. Problem Statement

The studied data collection planning is formulated as the Prize-Collecting Traveling Salesman Problem with Neighborhoods (PC-TSPN) which combines practical aspects of the robotic data collection considering ability to retrieve data from sensor stations within a communication radius ρ together with the aspect of importance to collect data from a particular sensor. The problem is motivated by persistent environment monitoring missions, where it is requested to periodically retrieve data from the deployed sensor network to create a model of some studied phenomenon.

It is assumed the phenomenon can be measured by up to n sensor stations $\mathbf{S} = \{s_1, \ldots, s_n\}$ located in the operational environment $\mathcal{W}$. For simplicity, we consider the sensors are located in a plane, $\mathcal{W} \subset \mathbb{R}^2$, and each sensor has associated point $s_i \in \mathbb{R}^2$ denoting the coordinates of the sensor. It is assumed the data collection vehicle is operating in $\mathbb{R}^2$ with a constant average velocity and its travel cost $c(p_1, p_2)$ between two points p_1 and p_2 is defined for all pairs of points $p_1, p_2 \in \mathbb{R}^2$ and thus, the travel cost can be directly computed as the Euclidean distance, i.e., $c(p_1, p_2) = |(p_1, p_2)|$.

For the PC-TSP [8], each sensor $s_i \in \mathbf{S}$ has associated penalty $\zeta(s_i) \geq 0$ characterizing the additional cost if the data are not retrieved from s_i. Regarding the TSPN part of the joint formulation of the PC-TSPN, it is assumed the data can be retrieved from the sensor if the vehicle is within the communication range ρ from the sensor, i.e., the data from s_i are collected at the point $p \in \mathbb{R}^2$ if $|(p, s_i)| \leq \rho$.

The data collection planning is a problem to determine a cost efficient path to retrieve data from the sensors. Thus, we can define a subset of k sensors $\mathbf{S}_k$ selected from the whole set of sensors $\mathbf{S}_k \subseteq \mathbf{S}$ from which data are collected. The data collection tour can be then defined as a permutation $\Sigma = (\sigma_1, \ldots, \sigma_k)$ such that $1 \leq \sigma_i \leq n$ and $\sigma_i \neq \sigma_j$ for $i \neq j$. However, data from the individual sensors $s_i \in \mathbf{S}_k$ can be retrieved within the communication radius ρ, and therefore, the data collection path is a sequence of k waypoints $P_k = (p_{\sigma_1}, \ldots, p_{\sigma_k})$ for which $|(p_{\sigma_i}, s_{\sigma_i})| \leq \rho$.

Having these assumptions, the PC-TSPN can be formulated as the optimization problem to determine the number of sensors k, the subset of the sensors $\mathbf{S}_k$ with the corresponding waypoints P_k, and their sequence Σ such that the solution cost $\mathcal{C}(P_k)$ is minimal

$$\mathcal{C}(P_k) = \sum_{i=2}^{k} \left(|(p_{\sigma_{i-1}}, p_{\sigma_i})|\right) + |(p_{\sigma_k}, p_{\sigma_1})| + \sum_{s \in \mathbf{S} \setminus \mathbf{S}_k} \zeta(s). \quad (1)$$

The first two terms of (1) represent the length of the shortest path connecting the waypoints, i.e., a solution of the TSP, and the last term is the sum of penalties associated to the sensors from which data are not collected from P_k.

The main difficulty of the PC-TSPN is that the solution cost depends on k sensors $\mathbf{S}_k \subseteq \mathbf{S}$ from which data are read, and the shortest path connecting the corresponding P_k, which is represented as the permutation Σ; and all these variables have to be considered simultaneously. Notice, if $\rho = 0$, the problem becomes the PC-TSP, for zero penalties $\zeta(s_i) = 0$ for all $s_i \in \mathbf{S}$ the problem is the TSPN, and finally for zero penalties and $\rho = 0$ the problem is the ordinary TSP.

In [10], we propose a SOM-based algorithm for the PC-TSPN in which we assume the sensor measurements provided by the sensors are independent and thus, the associated penalties $\zeta(s)$ are constant values regardless data from spatially close sensors are retrieved or not. This may not be always satisfied, and therefore, spatially correlated measurements can significantly influence a penalty value of some sensor s_i if data from the neighboring sensors also contain information about the location at s_i. This can be addressed by considering the penalty value is dependent on the selected subset of the sensors $\mathbf{S}_k$, which is further described in the next subsection.

A. Data Collection with Spatially Correlated Measurements

A model of the studied phenomenon is necessary to consider spatial correlations between measurements taken at the particular sensor locations. The phenomenon can be modeled as a time-varying scalar field, $\Psi(p,t)$, $p \in \mathbb{R}^2$ with $\mathcal{W}$ as its support [11]. Thus, the problem is to create a model of the field from data sampled at the locations $\mathbf{S}$, i.e., to collect values $\Psi(s,t)$ at the location s for $s \in \mathbf{S}$. Similarly to [11], we can consider a graph $G(V,E)$ of spatial relations among the sensor nodes V, such that the graph $G(V,E)$ has an edge (v_i, v_j) if and only if $\Psi(v_j,t)$ is dependent on $\Psi(v_i,t)$, where v_i corresponds to the location of the sensor $s_i \in \mathbf{S}$. Then, the value of $\Psi(v_i,t)$ can be expressed as

$$\Psi(v_i,t) = f_i(\Psi(v_{i_1},t), \dots, \Psi(v_{i_l},t)), \tag{2}$$

where $N_i = \{v_{i_1}, \dots, v_{i_l}\}$ are the neighboring sensors of v_i in the graph $G(V,E)$.

For a subset of sensors $\mathbf{S}_k$ and the corresponding subset $V_k \subseteq V$, we can compute a quality of the field model created from the values $\Psi(v,t)$ for $v \in V_k$ as a utility function $J : \{\mathbf{S}_k\} \to \mathbb{R}^+ \cup \{0\}$ that maps data from the sensors $\mathbf{S}_k$ to reals [11]. Then, an individual contribution of some sensor s from which data are not retrieved $s \notin \mathbf{S}_k$ can be expressed as

$$J_{\mathbf{S}_k}(s) = J(\mathbf{S}_k \cup \{s\}) - J(\mathbf{S}_k). \tag{3}$$

Since the PC-TSPN combines the travel cost with the penalty $\zeta(s)$ associated to the sensor s from which data are not collected, we need a function that will scale $J_{\mathbf{S}_k}(s)$ to the penalty $\zeta(s)$, e.g., $\zeta(s) = -\lambda J_{\mathbf{S}_k}(s)$, where $\lambda > 0$.

B. Distance based Correlation of Sensor Measurements

The aforementioned relations (2) and (3) provide a general way how to determine penalties based on the selected sensors $\mathbf{S}_k$. Such an evaluation of the penalties can be directly utilized in the proposed SOM approach to update penalties during the unsupervised learning according to the current solution represented by the network. However, a particular phenomenon model with the spatial correlation is necessary to validate if such an on-line evaluation of the penalties would provide desired improvement of the solution cost. Such a model is domain specific, and therefore a simplified (while still general enough) model based on mutual distances of the sensor locations is proposed here to demonstrate the proposed generalization of SOM for the PC-TSPN with spatial correlations of sensor measurements.

The main idea of the spatial correlation between measurements from different locations is that data from one or several sensors (locations) provide also information that is included in other sensors, i.e., a value of the scalar field at the particular locations (2). Thus, we can expect the correlation between two measurements at the locations s_i and s_j is increasing with the decreasing mutual distance $|(s_i, s_j)|$. On the other hand, from a certain distance, measurements at s_i and s_j may not be correlated at all. This can be characterized by decreasing penalty $\zeta(s_i)$ if data are collected only from s_j and both locations s_i and s_j are closed enough. For some very close location s_j to s_i, the penalty of not visiting s_j would be zero $\zeta(s_i) = 0$. In addition, we cannot expect that two mutually very close locations s_j and s_k would provide significantly more information about the location s_i in comparison to measurements from a single location s_j or s_k. These ideas about the influence of the correlation between the sensor locations is a source of motivation to propose the penalty evaluation based on geometrical coverage of circle representing the penalty associated to a sensor by other circles that represent the influencing locations as follows.

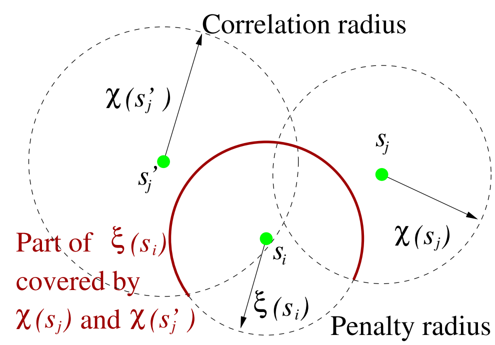


Fig. 2. A visualization of the geometrical relations in the proposed geometrical based computational model of spatial correlations between the sensor locations. Data from the sensors s_j and s_j' are collected, $s_j, s_j' \in \mathbf{S}_k$, and they are also the neighboring sensors of the sensor s_i, i.e., $s_j, s_j' \in N_i$, from which data are not directly retrieved $s_i \notin \mathbf{S}_k$. The red part of the circle $\xi(s_i)$ represents a particular portion of data at s_i that are collected by retrieved data from s_j and s_j'. The length of the red part of the circle is $\mathcal{L}(\xi(s_i), N_i)$ in (4) for the particular selected sensors $\mathbf{S}_k$.

Let S_k be sensors from which data are retrieved by the data collection path, N_i be the neighboring sensors of the sensor s_i in the graph $G(V,E)$ according to (2), $\zeta(s_i)$ be the penalty of s_i without correlations, ξ_i be the penalty radius of the circle centered at s_i, and χ_j be a radius of the circle centered at $s_j \in N_i$. The circles $\xi_i(s_i)$ and $\chi_j(s_j)$ are called the penalty circle (radius) and the correlation circle (radius), respectively, in the rest of this paper. The circumference of the circle $\xi_i(s_i)$ represents the associated penalty $\zeta(s_i)$ to the sensors s_i, i.e., computed from (3) for $\mathbf{S}_k = \emptyset$. If data from the neighboring sensors $s_j \in N_i$ are collected, i.e., $s_j \in \mathbf{S}_k$, the penalty $\zeta(s_i)$ is proportionally decreased according to the part of the circumference of $\xi_i(s_i)$ that is covered by the particular sensor $s_j \in N_i \setminus \mathbf{S}_k$ considering the circle $\chi(s_j)$. The geometrical relations are depicted in Fig. 2 and for a particular $\mathbf{S}_k$ the penalty $\zeta_{\mathbf{S}_k}(s_i)$ is formally computed as:

$$\zeta_{\mathbf{S}_k}(s_i) = \left(1 - \frac{\mathcal{L}(\xi(s_i), N_i)}{\mathrm{circ}(\xi(s_i))}\right)\zeta(s_i), \tag{4}$$

where $\mathrm{circ}(\xi(s_i))$ denotes a circumference of the circle $\xi(s_i)$ with the radius ξ centered at s_i and $\mathcal{L}(\xi(s_i), N_i)$ is the sum of parts of the circle $\xi(s_i)$ covered by the circles $\chi(s_j)$ for $s_j \in \mathbf{S}_k \cap N_i$, i.e., the length of $\bigcup_{s_j \in \mathbf{S}_k \cap N_i}(\xi(s_i) \cap \chi(s_j))$.

Notice, if none of the neighboring sensors of s_i is collected (i.e., $\mathbf{S}_k \cap N_i = \emptyset$) the penalty $\zeta_{\mathbf{S}_k}(s_i)$ associated to s_i is $\zeta_{\mathbf{S}_k}(s_i) = \zeta(s_i)$.

Each sensor may have an individual value of ζ, ξ and also the influencing radius χ can be distinguished for each pair of sensors, which provide an additional flexibility to model a domain specific spatial correlations. However, any other complicated function describing the spatial correlations between sensor locations can be considered, e.g., according to the general function (2).

The key idea of the proposed SOM-based approach is that whenever a new sensor $s \in \mathbf{S}$ is considered for data collection and s becomes part of $\mathbf{S_k}$, the penalty of each sensor from which data are not collected is recomputed using (4) and thus, it may happen that a penalty can be decreased to the value for which a travel cost to retrieve data from the sensor would be significantly higher and it does not make sense to travel towards such a sensor. Therefore, we can expect a lower solution cost if spatial correlations between the sensor measurements are considered during the solution of the PC-TSPN. The proposed extension of the SOM algorithm for spatial correlations is described in the next section.

III. Self-Organizing Map for the PC-TSPN with Spatial Correlation of Sensors Measurements

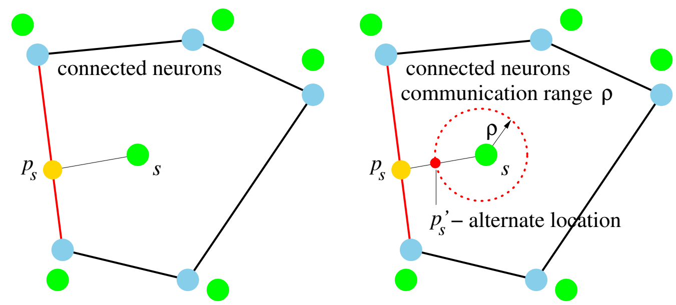
The Self-Organizing Map (SOM) for the PC-TSPN [10] is a two layered growing neural network which maps the input space $\mathbb{R}^2$ into one-dimensional output space representing the data collection path. The input layer serves as the input of the network to describe coordinates of the given sensor locations $\mathbf{S}$. The neuron weights represent coordinates in $\mathbb{R}^2$ which adapt to $\mathbf{S}$ during the unsupervised learning. The output layer consists of m output units $(\nu_1, \ldots, \nu_m)$ organized into an array that prescribes a sequence of the neuron weights that can be connected by straight line segments to form a ring of nodes (neuron weights) that represent a path in $\mathbb{R}^2$. A closed path is required in the TSP and the output layer is a circular array where the last node is connected with the first node.

The training of SOM for the TSP is an unsupervised learning procedure that is performed in series of learning epochs. During each learning epoch, all sensors $\mathbf{S}$ are presented to the network in a random order (to avoid local extreme). For each presented $s_i \in \mathbf{S}$ a winner neuron ν^* is selected using Euclidean distance of the neuron weights to s_i and the winner ν^* together with its neighboring neurons are then adapted towards s_i. The main principle of the unsupervised learning is that the neighboring neurons are adapted with decreasing power according to the neighboring function:

$$f(G, d) = \begin{cases} \mu e^{\frac{-d^2}{G^2}} & \text{for } d < 0.2m \\ 0 & \text{otherwise} \end{cases}, \qquad (5)$$

where μ is the learning rate, m is the number of neurons, and d is the distance of the neighboring neuron from the winner neuron in the number of neurons (in the output layer). Besides, the power of the adaptation controlled by the learning gain G is decreased after each learning epoch i according to the gain decreasing rate α, e.g., $G(i + 1) = (1 - \alpha)G(i)$.

In a standard SOM for the TSP, the number of output units m is fixed (e.g., usually selected as $2n \le m \le 3n$) [12] during the whole learning because the final route has to visit all the locations presented to the network. However, in the PC-TSPN, the final path may not necessarily visit all the sensors regarding the solution cost (1), and therefore, it is a more suitable to adjust the number of neurons according to the currently selected sensors $\mathbf{S}_k$ from which data will be retrieved. That is why SOM for the PC-TSPN is considered as a growing structure and new neurons are added to the network during the winner selection and unnecessary neurons are removed in the ring regeneration procedure after each learning epoch [10].



(a) Closest point p_s of the ring to s (b) Alternate location p'_s

Fig. 3. Determination of the location p_s of the possible winner neuron for the presented sensor stations s together with determination of the alternate location p'_s from which data from s can be retrieved using the communication range ρ. The sensor locations are shown as green disks and neurons are the blue disks connected by straight line segments forming the ring of neurons. The closest point p_s of the ring to the sensor s is visualized as the yellow disk. The alternate location p'_s towards which the network is adapted (if the winner neuron for s is determined) is shown as the red disk. The dotted circle with the radius ρ centered at s represents an area from which data can be reliably retrieved from s.

Because the network adaptation towards the currently presented sensor s is conditioned according to the distance of the winner to the target location and the current penalty $\zeta_{\mathbf{S}_k}(s)$ for not visiting s, a location p_s of an eventual winner neuron is determined prior establishing the winner for s. The point p_s is found as the closest point of the current ring of neurons to s. Besides, an intersection of the straight line segment (p_s, s) and the circle with the radius ρ centered at s is utilized to determine a point p'_s at which data from s can be retrieved. Notice, if p_s is already within the communication radius ρ the point p_s is used as the target location p'_s. The determination of p_s and p'_s is visualized in Fig. 3.

The proposed unsupervised learning of SOM for the PC-TSPN with spatial correlations can be summarized as follows:

1) *Initialization*: For n sensor locations $\mathbf{S} = \{s_1, \ldots, s_n\}$ create an initial ring of $m = n$ neurons $\mathcal{N} = \{\nu_1, \ldots, \nu_m\}$ around the first sensor location s_1, e.g., as a small circle. Set the learning gain G to $G \leftarrow 10$ and the current learning epoch $i \leftarrow 1$. Initialize the learning rate $\mu = 0.6$ and the gain decreasing rate $\alpha = 5 \times 10^{-4}$. Set the selected sensors $\mathbf{S}_k \leftarrow \emptyset$.
2) *Randomizing*: Create a random permutation of the sensors $\Pi(\mathbf{S}) \leftarrow \text{permute}(\mathbf{S})$.

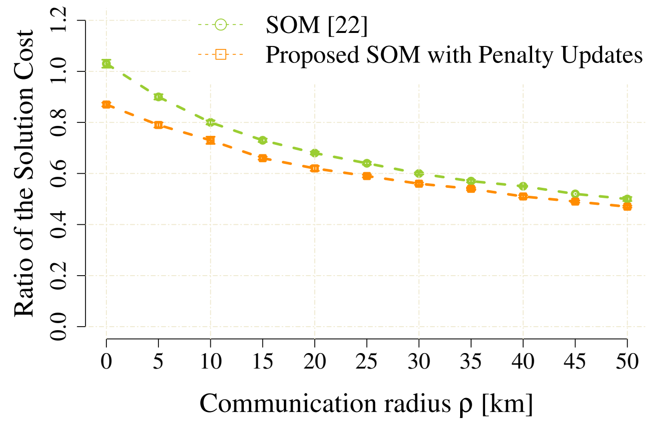
▷ **Learning epoch:**

3) Let the number of neurons in the current ring $\mathcal{N}$ be m and $\mathcal{N}$ represents the solution as the data collection path P_m. Determine the sensors $\mathbf{S}_k$ that are within the communication radius ρ from the ring $\mathcal{N}$, i.e., $\mathbf{S}_k = \{s_i | s_i \in \mathbf{S}, \nu \in \mathcal{N}$ for which $|(\nu, s_i)| \leq \rho\}$. Update penalties ζ_i for all $s_i \in \mathbf{S} \setminus \mathbf{S}_k$ using (4).
4) For each $s \in \Pi(S)$:
   a) *Determine weights of the expected winner p_s and location p'_s to read data from s* using the closest point p_s of the current ring to s (see Fig. 3a) and the intersection point p'_s of the segment (p_s, s) with the ρ-radius circle centered at s, see Fig. 3b.
   b) *Adapt:* `If` $i = 1$ `or` $|(p_s, p'_s)| \leq \zeta(s)$ `Then`
      - Determine the winner neuron ν^* for p_s (select it from the current ring or create a new one);
      - Associate p'_s with ν^* as the target location to read data from s and add s to the set of the selected sensors: $\mathbf{S}_k \leftarrow \mathbf{S}_k \cup \{s\}$;
      - Update penalties $\zeta(s_i)$ of the sensors $s_i \in \mathbf{S} \setminus \mathbf{S}_k$ that can be influenced by s according to (4).
      - Adapt ν^* and its neighbors within the distance d (in the number of neurons) to the determined p'_s according to the neighbouring function (5).
5) *Ring regeneration*: Create a new ring using only the winners for the current epoch i. Add a new neuron between each consecutive winners ν and ν' with the weights set as the midpoint of the segment (ν, ν').
6) *Update the learning gain and the epoch counter*: $G \leftarrow G(1 - i\alpha)$, $i \leftarrow i + 1$.
7) *Termination condition:* `If` the distance of each winner to its associated target location is less than 10^{-3} `Or` $i > 100$, stop the adaptation. `Otherwise` go to Step 2.
8) *Final tour construction:* Traverse the ring for the sequence of winners and construct the data collection path P_k using the target locations associated to the winners.

Computational complexity of the proposed iterative procedure depends on the number of sensors n and the number of neurons m. Since in each learning epoch up to n new neurons can be added to the network, m is always $m \leq 2n$. Then, up to $2{\cdot}0.2m$ neurons can be adapted and thus, the complexity of the single epoch with constant penalties can be bounded by $O(n^2)$. However, for correlated sensor measurements, penalties of all not collected sensors are updated after selection of each winner. In the worst case, each sensor can influence all other sensors and it may be necessary to determine (2) and (3) which can be bounded by $O(n^2)$ and the complexity of the single epoch can be bounded by $O(n^3)$. However, real computational requirements can be lower due to correlations only between spatially close sensors.

IV. Results

The proposed extension of the SOM approach for the PC-TSPN to spatial correlations between sensor measurements have been empirically evaluated in series of problems motivated by under water data collection missions in the OOI scenario with 128 sensors, which has been also considered in [10]. Although the proposed SOM algorithm and data correlation model allow to consider any phenomenon specific update function of the penalties, the geometric-based model of the correlations (Section II-B) is considered to demonstrate ability to deal with spatial correlations. The geometric model allows to specify individual communication radius ρ, penalty radius ξ, and coverage radius χ for each particular sensor, e.g., to respect local environment conditions; however, a single value of each radius is considered for simplicity.



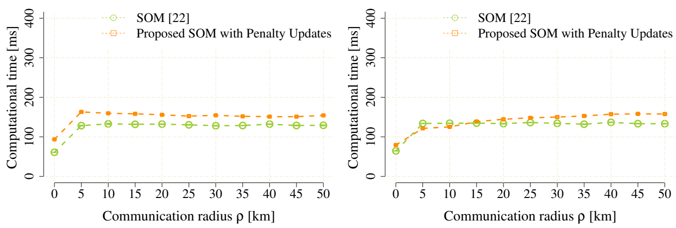
(a) $\chi = 35$ km

Fig. 4. The solution cost found by the SOM-based algorithm for the PC-TSPN [10] and its generalization for spatial correlations with χ in OOI scenario with $n = 128$ sensors. For both approaches the final solution cost is computed with the spatial correlations. Results are average values from 20 trials and standard deviations shown as error bars are negligibly small.

In particular, the communication radius is considered in the range $0 \leq \rho \leq 50$ km, the penalty radius ξ is set to 10 km, and the correlation radius χ is selected from the range $0 \leq \chi \leq 50$ km, which provide a broad spectrum of the problems where correlations between the sensor measurements are less (for zero or small values of χ) and more important (for $\chi \gg 0$).

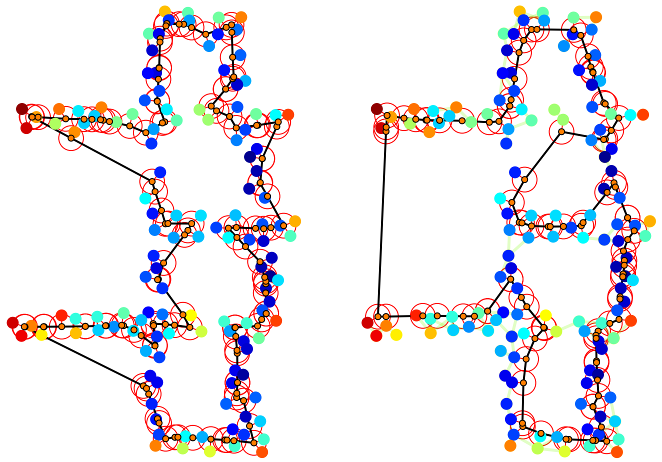
SOM is a randomized algorithm, and therefore, each problem instance for particular values of ρ and χ has been solved 20 times and the performance indicators of the solution cost according to (1) and real computational time are computed as the average values accompanied by the standard deviations. Similarly to [10], the solution cost is considered as the ratio to the reference value computed as the optimal solution of the related TSP visiting all sensors and $\rho = 0$ found by the Concorde [13]. The ratio allows to evaluate benefits of the PC-TSPN formulation in comparison to the ordinary TSP.

The final solution cost is computed according to the spatial correlations using (4) for both approaches: (i) SOM for the PC-TSPN introduced in [10] and; (ii) the herein proposed generalization with the spatial correlations. In the case of not spatially correlated measurements, the both approaches provide identical results. However, the proposed approach provides better results for increasing correlation radius χ. A noticeable improvement can be seen for $\chi = 35$ km in Fig. 4, especially for small communication radii. It can be observed that considering spatial correlations and non-zero communication radius have a great impact to the solution cost.



(a) χ =14 km (b) χ =35 km

Fig. 5. Average computational time to find a solution of the PC-TSPN by the SOM algorithm without and with consideration of spatial correlations between the sensor measurements



(a) SOM [10], C=2761.0 (b) Proposed, C=2477.5

Fig. 6. Best found solutions for the OOI scenario with ρ = 20 km, the penalty radius $\xi = 10$ km and correlation radius $\chi = 35$ km

The SOM algorithm has been implemented in C++ and all the results have been computed using the same computational environment with a single core of 3.4 GHz CPU, and therefore, the influence of dealing with spatial correlations to the computational requirements can be observed in Fig. 5. For $\chi = 14$ km, the computational burden is only slightly increased because of relatively small correlation radius χ, which also holds for higher χ and small radii $\rho \leq 15$ km.

A. Discussion

The presented results support feasibility of the proposed approach to address spatial correlations between sensor measurements at different locations. The generalized algorithm provides solutions with a lower cost than the previous approach without considering the correlations. Moreover, the results indicate the computational burden is only slightly increased. On the other hand, the utilized model of the spatial correlation is based on geometrical relations between the sensor locations and it is relatively computationally inexpensive and a more complex evaluation of (2) can be more demanding.

V. Conclusion

In this paper, a generalization of the PC-TSPN for data collection planning with spatial correlations between sensor measurements has been proposed together with the extension of the SOM-based algorithm to address the generalized problem. The proposed algorithm provides similar results as the previous approach in problem instances where the effect of spatial correlations is low, i.e., for $\chi = 0$, but it provides noticeably better solutions if data from one sensor also contains information about other locations.

The evaluation results are based on an intuitive function to update penalties based on the visited nearby sensors. The proposed algorithm is independent on the particular form of the function to update the penalties, and one of our future research directions is to consider a realistic setup with particular domain specific scalar field.

Here it is worth mentioning that for the PC-TSPN formulation of the data collection planning, it is assumed that a set of sensor locations of some pre-deployed sensor field is given. Then, a solution of the PC-TSPN may not include all the sensors on the data collection path and thus, some of the sensor stations are not necessary. Therefore, the proposed approach can be utilized to determine the particular locations where the sensor stations should be deployed, e.g., considering a very dense sensor field. Investigation of such a utilization of the proposed approach is a subject of our future work.

Acknowledgments – The presented work was supported by the Czech Science Foundation (GAČR) under research project No. 15-09600Y. The support of grant No. SGS16/235/OHK3/3T/13 to Petr Váňa is gratefully acknowledged.

References

[1] D. Bhadauria, O. Tekdas, and V. Isler, "Robotic data mules for collecting data over sparse sensor fields," *Journal of Field Robotics*, vol. 28, no. 3, pp. 388–404, 2011.

[2] O. Tekdas, V. Isler, J. H. Lim, and A. Terzis, "Using mobile robots to harvest data from sensor fields," *IEEE Wireless Communications*, vol. 16, no. 1, p. 22, 2009.

[3] M. Dunbabin and L. Marques, "Robots for Environmental Monitoring: Significant Advancements and Applications," *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 24–39, 2012.

[4] D. L. Applegate, R. E. Bixby, V. Chvatal, and W. J. Cook, *The Traveling Salesman Problem: A Computational Study (Princeton Series in Applied Mathematics)*. Princeton, NJ, USA: Princeton University Press, 2007.

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

[6] I. Gentilini, F. Margot, and K. Shimada, "The travelling salesman problem with neighbourhoods: MINLP solution," *Optimization Methods and Software*, vol. 28, no. 2, pp. 364–378, 2013.

[7] J. Faigl and L. Přeučil, "Self-Organizing Map for the Multi-Goal Path Planning with Polygonal Goals," in *International Conference on Artificial Neural Networks (ICANN)*. Heidelberg: Springer, 2011, pp. 85–92.

[8] E. Balas, "The Prize Collecting Traveling Salesman Problem," *Networks*, vol. 19, pp. 621–636, 1989.

[9] G. Hollinger, U. Mitra, and G. Sukhatme, "Autonomous data collection from underwater sensor networks using acoustic communication," in *IROS*, 2011, pp. 3564–3570.

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

[11] J. Yu, M. Schwager, and D. Rus, "Correlated orienteering problem and its application to informative path planning for persistent monitoring tasks," in *IROS*, 2014, pp. 342–349.

[12] S. Somhom, A. Modares, and T. Enkawa, "A self-organising model for the travelling salesman problem," *Journal of the Operational Research Society*, vol. 48, no. 9, pp. 919–928, 1997.

[13] D. Applegate, R. Bixby, V. Chvátal, and W. Cook, "Concorde tsp solver," 2003, [cited 24 April 2016]. [Online]. Available: http://www.tsp.gatech.edu/concorde.html