Unifying Multi-Goal Path Planning for Autonomous Data Collection

Jan Faigl[1], *Member, IEEE,* and Geoffrey A. Hollinger[2], *Member, IEEE*

***Abstract*— In this paper, we propose a framework for solving variants of the multi-goal path planning problem with applications to autonomous data collection. Autonomous data collection requires optimizing the trajectory of a mobile vehicle to collect data from a number of stationary sensors in a known configuration. The proposed approach utilizes the self-organizing map (SOM) architecture to provide a unified solution to multi-goal path planning problems. Our approach applies to cases where the vehicle must move within a radius of a sensor to collect data and also where some sensors can be ignored due to a lower priority. We compare our proposed approach to state-of-the-art approximate solutions to variants of the Traveling Salesman Problem (TSP) for random deployments and in an underwater monitoring application domain. Our results demonstrate that the SOM approach outperforms combinatorial heuristic algorithms and also provides a unified approach for solving variants of the multi-goal path planning problem.**

I. Introduction

As pre-deployed sensors are becoming more prominent (e.g., in environmental monitoring and surveillance scenarios), it becomes increasingly important to efficiently collect data from these sensors. For example, sensors are currently in place on the ocean floor to monitor seismic activity, but there is currently no way to collect data from them without physically plugging an underwater vehicle into the sensors [1]. A less costly and more convenient method for collecting this data would be to equip the sensors with wireless communication (e.g., radio in terrestrial domains or acoustic communication in underwater domains) and utilize a mobile vehicle to retrieve the data [2].

The above scenario requires visiting a set of pre-specified goals and can be formulated as a variant to the Traveling Salesman Problem (TSP). However, two key aspects of the problem differentiate it from the classical TSP: (1) the sensors do not need to be visited exactly (the vehicle must only move within a neighborhood of them to download the data); and (2) the data from some sensors may be lower priority (it would be best to ignore those sensors). These two aspects of the problem can be incorporated using the Traveling Salesman with Neighborhoods (TSPN) and the Prize-Collection Traveling Salesman (PC-TSP) formulations.

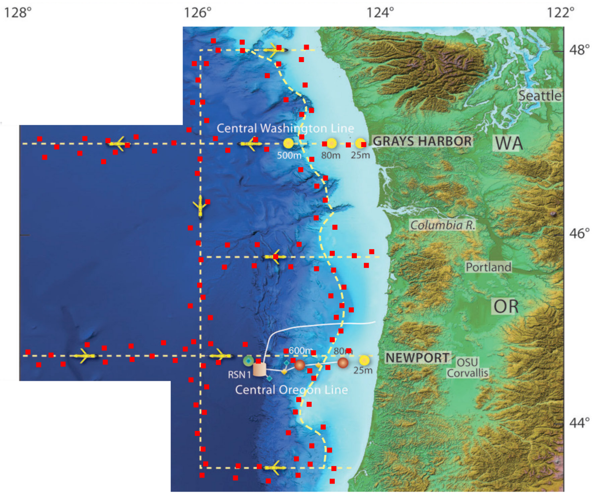


Fig. 1. Ocean Observatories Initiative Endurance Array [3] with simulated sensors placed on the ocean floor (red squares). An autonomous underwater vehicle is tasked with collecting data from these sensors using wireless acoustic communication. Our proposed approach provides a unified framework for planning the data collection tours of an autonomous vehicle in such multi-goal path planning scenarios.

Both the PC-TSP and the TSPN have been studied in the literature, including in our own prior work, and solved using combinatorial optimization techniques [2]. Existing frameworks for the PC-TSP and TSPN problems utilize a "cocktail" approach where heuristics are used to determine which locations to visit and a standard TSP solver is then used to generate a tour. Prior work has left open the problem of developing a unified architecture that can solve this entire class of optimization problems (i.e., TSP, PC-TSP, TSPN, and PC-TSPN). Our proposed approach utilizes self-organizing map to bridge this gap.

The self-organizing map (SOM) is a two-layered artificial neural network that provides a non-linear transformation (map) of a high-dimensional input space into a lower dimensional discrete output space. Its main feature is that it preserves topological properties of the input space in the output space. Although SOM was originally proposed as a data visualization technique, it has also been applied to solve NP-hard routing problems such as the TSP [4].

In the current paper, we follow the recent advancements of SOM and propose a new adaptation schema to provide a unified architecture for solving TSP variants. We propose a new growing SOM approach to multi-goal path planning problems based on the self-adjusting structure of the neural network from our prior work [5]. To the best of our knowledge, this is the first application of SOM to multi-goal path planning with both neighborhoods and prizes on the goals.

The remainder of this paper is organized as follows. We first discuss related work in autonomous data collection to demonstrate the need for a unified architecture (Section II).

The presented work is supported by the Czech Science Foundation (GAČR) under research project No. 13-18316P. The support under NSF grant IIS-1317815 to Geoffrey Hollinger is also gratefully acknowledged.

[1] J. Faigl is with the dept. of Computer Science, Czech Technical University in Prague, Technická 2, 166 27 Prague, Czech Republic faiglj@fel.cvut.cz

[2] G. Hollinger is with the School of Mechanical, Industrial, and Manufacturing Engineering, Oregon State University, 204 Rogers Hall, Corvallis, OR 97331 geoff.hollinger@oregonstate.edu

The multi-goal path planning problem is then formulated, and the link to autonomous data collection is shown (Section III). We then introduce the proposed unified solution using a self-organizing map (Section IV). Results of the proposed approach and a comparison with state-of-the-art combinatorial heuristic methods are presented in Section V together with a discussion of found insights. Section VI is dedicated to concluding remarks and future work.

II. Related Work

The TSP can be formulated as follows: having a set of locations (cities) in a plane and a distance function between them, the problem stands to find a shortest tour connecting all the given cities, such that each city is visited exactly once and the tour returns to the origin city. This problem arises from many practical applications, and efficient heuristics have been proposed in the operations research community [6].

The autonomous data collection problem is closely related to the TSP. In autonomous data collection, a mobile vehicle must collect data from a pre-deployed set of sensors. If wireless communication is available, the vehicle may not need to move to the same location as the sensor, but instead may only need to move within a radius of the sensor. This situation can be modeled using the TSPN formulation [7], [8]. The case where locations are associated with different "prizes" (and can be ignored for a corresponding penalty) has also been studied by a number of researchers. The prize-collecting TSP was originally introduced by Balas [9] and has been extended to a number of related variants [10]. Goemans and Williamson proposed an approximation algorithm for prize-collecting TSP using an LP primal/dual scheme that achieves a factor of two guarantee [11].

Related problems have been studied in the context of robotic data mules. Bhadauria and Isler derived approximation algorithms for multiple data mules that must traverse a sensor field and download data [12]. In their work, downloading time is considered as part of the tour, and the communication radii are assumed to be uniform, fixed, and deterministic (i.e., data from a sensor is known to be accessible at a given location). Vasilescu et al. developed a system of mobile and stationary nodes for underwater data collection based on the use of both optical and acoustic communication [13]. They described the networking architecture and sensor specifications necessary for underwater data collection and presented field experiments using a mobile network.

In our prior work, we proposed a heuristic solution to the PC-TSPN problem [2]. This approach generates a covering set of goals based on the size of the neighborhoods and then utilizes an approximation algorithm [11] to select a subset of the covering set to visit. The Concorde solver [6] is then used to determine a tour of this subset. We demonstrated that this approach outperforms competing algorithms in autonomous data collection scenarios [2]. The key drawbacks of our prior approach are the need to use an underlying TSP solver and the heuristic nature of the subgoal selection. In the current paper, we propose an SOM architecture that eliminates these drawbacks and provides improved performance.

III. Problem Formulation

The herein studied data collecting task is formulated as a combination of two variants of the traveling salesman problem (TSP) that are called the prize-collecting TSP (PC-TSP) and TSP with neighborhoods (TSPN) and thus, the problem is called the prize-collecting traveling salesman problem with neighborhoods (PC-TSPN). This formulation provides sufficient generalization of the problem to consider both practical aspects of the data collecting: 1) the importance of the data read from a particular sensor station (which is addressed in the PC-TSP); 2) and the ability to consider a communication radius to actually retrieve the data from the station (the TSPN). If it is required that data from all given sensors must be read and communication is reliable only within very close distance of the vehicle to the sensor station, the problem becomes the classical TSP.

Thus, the problem is to find a cost efficient path to retrieve data from the given set of n sensor stations. For simplicity, we assume the sensor stations are located in $\mathbb{R}^2$, and their positions are $\mathbf{S} = \{s_1, \ldots, s_n\}$, $s_i \in \mathbb{R}^2$. Each station s_i has associated penalty $\zeta(s_i) \geq 0$ characterizing the additional cost if the data are not retrieved from s_i. The data can be retrieved from the sensor by a vehicle from the ρ distance, the vehicle is operating in $\mathbb{R}^2$, and the traversal cost $c(p_1, p_2)$ is defined for all pairs of points $p_1, p_2 \in \mathbb{R}^2$. Based on these assumptions, the problem is to determine a set of unique goal locations $\mathbf{G} = \{g_1, \ldots, g_k\}$, $k \leq n$, $g_i \in \mathbb{R}^2$, and $g_i \neq g_j$ for all $g_j, g_i \in \mathbf{G}$, at which the sensor data readings are performed, and to find a tour T visiting these locations $\mathbf{G}$ such that the total cost of the tour $\mathcal{C}(T)$ is minimal.

$$\mathcal{C}(T) = \sum_{(g_{l_i}, g_{l_{i+1}}) \in T} c(g_{l_i}, g_{l_{i+1}}) + \sum_{s \in \mathbf{S} \setminus \mathbf{S}_T} \zeta(s), \qquad (1)$$

where $\mathbf{S}_T$ is the subset of the sensor stations $\mathbf{S}_T \subseteq \mathbf{S}$ from which data can be retrieved at the tour goals; i.e., for each s, $s \in \mathbf{S}_T$ there exists $g \in \mathbf{G}$ such that the station s is within the communication radius ρ from g, $|(s, g)| \leq \rho$.

Notice, the requested tour T is a sequence of the determined goal locations $T = (g_{l_1}, \ldots, g_{l_{k-1}}, g_{l_k})$, where $g_{l_j} \in \mathbf{G}$, $l_j \geq 1$, $l_j \leq n$, and $g_{l_1} = g_{l_k}$.

In our motivational problem of autonomous data collection, the environment can be considered as planar and without obstacles. Therefore, the travel cost between two goals $c(g_i, g_j)$ is considered as the Euclidean distance $|(g_i, g_j)|$ for simplicity in the rest of this paper.

IV. Proposed Algorithm for the PC-TSPN

The proposed unifying algorithm for planning data collecting tasks is based on SOM for the TSPN (originally proposed in [5]) that is accompanied by new adaptation rules to consider the penalty associated with the sensor stations.

SOM is a two layered neural network that is trained by an unsupervised learning procedure. The network for the TSP can be considered as a non-linear mapping function of the input space $\mathbb{R}^2$ to one-dimensional output space representing the requested tour. The first layer describes the given sensor stations, and neuron weights represent coordinates in $\mathbb{R}^2$ that adapt to the presented location of the sensor stations. The array of output units describes a sequence of neuron weights that can be connected by straight line segments to form a ring of nodes (neuron weights) that represents the tour.

The unsupervised learning is an iterative procedure in which the sensor stations are presented to the network in a random order, and for each station a winning neuron is selected that is then adapted, together with its neighboring neurons, towards the station. The neighboring nodes are adapted with decreasing power according to the neighbouring function f. However, four important modifications of the standard SOM approach for the TSP [4] need to be considered to address the PC-TSPN:

1) the number of neurons should adapt to the currently selected sensor stations. This is addressed by the winner selection rule proposed in [5], where a new winner neuron is selected as the closest point of the current ring, to the presented goal. Contrary to [5], where the presented input to the network is a sequence of straight line segments, only a single sensor location is considered here. This winner selection might add new neurons to the network, which can be inefficient (regarding the adaptation of winner neighborhoods), and therefore, only winner neurons are preserved and all other neurons are removed at the end of each learning epoch after all the sensor stations are presented to the network. This approach, called ring regeneration, is further developed here to improve convergence of the network in the PC-TSPN. Additional neurons are placed between each preserved winner with neuron weights set to the center of the straight line segment connecting the winners.

2) the next modification considers the communication radius ρ, due to the fact that we must not visit the sensor stations precisely. Rather, it is sufficient to reach a location that will be in ρ distance to the station. This can be represented in the generation of a goal candidate during winner selection. Let the currently presented station to the network be s and its winner neuron be ν_s. Then, the goal candidate to collect data from s is a point g_s on the straight line segment (s,ν_s) for which $|(g_s, s)| < \rho$. Here, we rather consider the sharp inequality to avoid numerical issues.

3) to address selection of the sensors based on their penalty, the winner node determined in the aforementioned procedure is chosen as a winner "candidate". The candidate ν_s for the station s becomes the winner only if its distance to the goal candidate g_s is shorter than the sensor penalty, i.e., $|(\nu_s, g_s)| \leq \zeta(s)$; otherwise ν_s and g_s are discarded.

4) finally, several sensor stations can be reached from newly determined goal candidate location g_s. Therefore, after the adaptation of the network to g_s, all stations within ρ distance from g_s are marked as covered for the current epoch, and the network adapts only to the not covered stations.

A. Computational Complexity

The learning procedure is an iterative procedure where the most time consuming operations are the selection of the winner neurons, adaptation of neighboring nodes, and determination of sensors that can be covered. For n sensor stations, the number of neurons can be bounded by $3n$, and the number of selections is $3n^2$. For each winner, all neighboring nodes can eventually be adapted (up to $3n$ nodes), and n sensor stations can be tested to see if they would to be covered by a new goal. This gives $(3n+3n+n)n$ operations for each learning epoch. The number of epochs is related to the convergence of the network.

The adaptation rule is stable, since the learning rate and neighboring function f are always less than 1.0 [14]. The winner neighbors are effectively adapted only for a sufficiently high value of f, which is decreasing with the distance of the neighbor from the winner and its highest possible value is decreased after each learning epoch. Thus, the network is stabilized in a constant number of learning epochs and the overall complexity can be bounded by $O(n^2)$.

B. Discussion

The aforementioned learning procedure creates a SOM that maps the input space into a uni-dimensional structure of a discrete number of units. These units try to best fit the input space. The learning is a stochastic procedure, and it does not provide a guarantee that the optimal solution would be found in a finite number of learning epochs. Thus, the proposed approach represents a heuristic polynomial algorithm for solving NP-hard problems. On the other hand, a feasible solution is available at the end of each learning epoch; hence, an initial guess about the solution quality can be provided quickly. To the best of the authors' knowledge, there are no known approximation bounds for SOM-based TSP solvers. The empirical performance of the proposed solution is evaluated in Section V.

The presented procedure utilizes Euclidean distance in $\mathbb{R}^2$ to find the best matching unit to the presented station. An extension of the approach to $\mathbb{R}^3$ is straightforward by adding one more dimension to the input layer and considering neuron weights as points in $\mathbb{R}^3$. The output layer would be identically the one-dimensional array representing the sequence of goal locations in $\mathbb{R}^3$. In addition to this possible extension, the self-organizing map has already been successfully deployed for problems with obstacles [4], 3D environments [15], and with a general graph used as an input [16]. An extension of the proposed modifications to these types of environment representations is straightforward, and therefore, the proposed adaptation schema for the PC-TSPN can provide a wide range of possible applications. To further foster deployment of the proposed SOM-based PC-TSPN solver, we made the implementation available at [17].

V. Results

The performance of the proposed unifying SOM-based planning approach has been evaluated in a series of data collection problems where it is compared with the existing combinatorial deterministic approach for the PC-TSPN [18]. The deterministic approximation algorithm, denoted as PC-TSPN here, is based on finding a covering set of goals considering the neighborhoods of the sensor stations defined by the communication radius ρ. Then, the corresponding optimal solution of the TSP is found using the Concorde [19] solver on a subset of the covering set determined by the heuristic from [11].

The proposed SOM-based approach is considered in two variants. First, the unsupervised learning procedure is used for simultaneous determination of the goals together with the tour connecting them. In the second variant, only the goals determined by SOM are considered, while the tour is found as the optimal solution of the TSP by [19]. These algorithm variants are denoted as SOM and SOM+TSP, respectively.

A. Simulations with Random Deployments

The algorithm comparison is based on evaluation in several variants of random problem instances for different values of penalties and communication radius ρ. These instances include the standard TSP (for $\rho = 0$ and very high penalties), PC-TSP (for $\rho = 0$), and TSPN (for $\rho > 0$ and very high penalties). The considered communication radius is $0 \leq \rho \leq 50$ km, and penalties of sensor stations (from which data are not collected) are set randomly according to four different schemata listed in Table I.

TABLE I

CONSIDERED VALUES OF PENALTIES IN THE RANDOM DEPLOYMENTS

Penalty Assignment Schema	Penalty Range
very high penalties	$0 \leq \zeta \leq 25000$
high penalties	$0 \leq \zeta \leq 250$
middle penalties	$0 \leq \zeta \leq 25$
low penalties	$0 \leq \zeta \leq 5$

The first data collection scenario consists of 100 randomly placed sensors in a 100 km × 100 km large area. Similarly to [18], the vehicle speed is assumed to be 5 km per hour, and thus the goals are effectively placed in 20 × 20 large square and the cost between the locations is directly computed as their Euclidean distance.

For each scenario, 50 random instances were created, and for each such instance, the penalties were assigned according to the four different schemata depicted in Table I. The considered communication radius ρ consists of 11 different values (ranging from 0 to 50 km), which gives 2200 problems. The PC-TSPN algorithm [18] is deterministic, and thus only a single trial is considered for each problem. However, SOM is a stochastic algorithm, and therefore, 50 trials are solved for each problem instance.

To evaluate this large set of results, we standardize the solution quality as a ratio to a reference value, which allows us to aggregate results across various scenarios, penalty schema, and ρ values. We consider the following solution quality metric: the ratio of the tested algorithm's solution to the optimal solution of the underlying TSP (without considering the penalties and communication radius). Thus, for an optimal tour T_{TSP} of the underlying TSP, the ratio is computed as

$$R = \frac{\mathcal{C}(T)}{\mathcal{C}(T_{TSP})}. \tag{2}$$

The average values of the ratio R with the error bars denoting standard deviations are presented in Fig 2. We note that considering the prizes and neighborhoods, the proposed algorithms should typically outperform the underlying TSP.

In this scenario, the proposed SOM-based approach provides improved results versus the combinatoric PC-TSP heuristic. Regarding SOM and SOM+TSP, the main differences are for $\rho = 0$, which represents a solution of the TSP (or PC-TSP). SOM is a heuristic algorithm, and thus it does not guarantee the optimal solution would be found. The average difference is about 3-5% in comparison to the optimal solution found by [19]. The sensor station selection in the deterministic PC-TSP is not evident, as in most cases all stations are visited for $\rho = 0$, which is indicated by the ratio R close to 1.

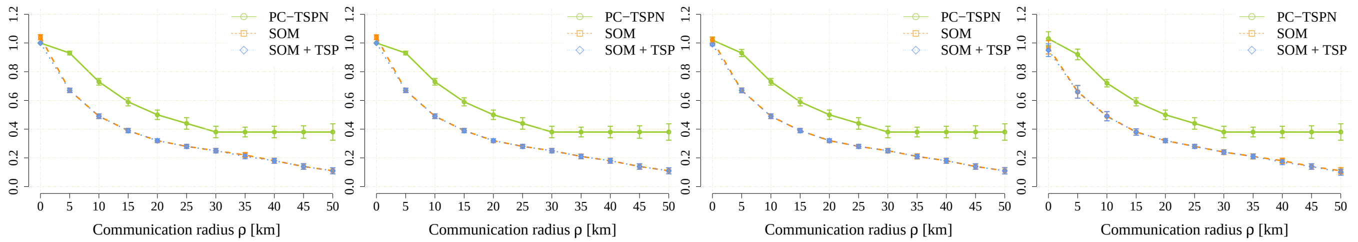
Required Computational Time: The evaluated algorithms are implemented in C++, and all the presented results have been computed using a single core of the iCore7 processor running at 3.4 GHz. The required computational time mainly depends on the number of determined goals, which is higher for high penalties and low ρ. Required computational time is depicted in Fig. 3. The SOM algorithm provides solutions in tens of milliseconds (less than 100 milliseconds), while the optimal solution of the TSP using [19] increases the computational time significantly. With increasing communication radius ρ, the computational burden decreases more significantly than for decreasing value of penalties.

B. Underwater Monitoring Deployment

We have also tested our proposed algorithm using a scenario taken from the Ocean Observatories Initiative (OOI) Endurance Array [3]. The OOI Endurance Array is a deployment of Autonomous Underwater Vehicles (AUVs) that is currently scheduled to go into operation off the coast of Oregon and Washington in 2014. Approximately six AUVs will remain in continuous operation along pre-specified operating lines, and they will monitor important biological and physical measurements related to the coastal ocean. For the purposes of this research, we only consider the case where a single AUV is deployed. Extensions to multiple vehicles is an interesting area for future work.

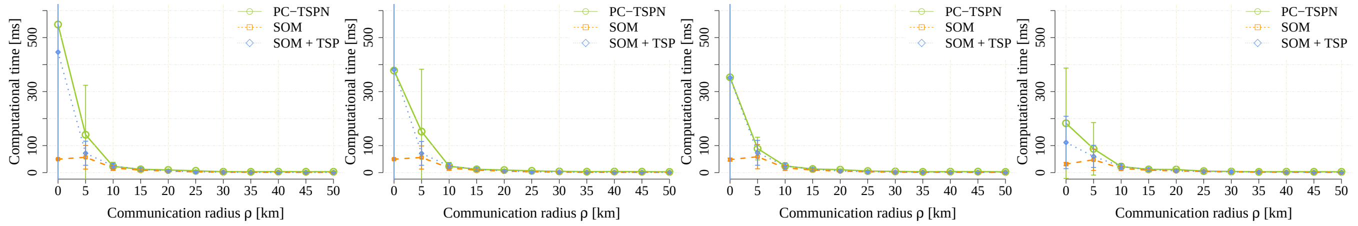
We have devised the following scenario based on the OOI Endurance Array to test our proposed approach. In simulation, we place approximately 100 sensors along the planned AUV paths, and we assume that the sensors and AUV are equipped with acoustic modems for wireless communication. Such acoustic modems are becoming widely available and have been proposed for sensor data collection tasks [2]. Depending on the type of acoustic modem and the power supplied to it, the range of communication can vary, which corresponds to the size of the neighborhoods. Recent work has shown that ranges up to 50 km are possible with current acoustic communications technology [20].[1]

[1] We note that in some scenarios, the communication radius will vary over the region of interest. In these cases, neighborhoods with variable size would need to be considered. This extension to the SOM architecture should be straightforward but is left for future work.



(a) *very high penalties* (b) *high penalties* (c) *middle penalties* (d) *low penalties*

Fig. 2. Average ratios R of the solution cost to the optimal solutions of the related TSP in the 100 km × 100 km area scenario (lower is better)



(a) *very high penalties* (b) *high penalties* (c) *middle penalties* (d) *low penalties*

Fig. 3. Average required computational time (in milliseconds) for the problems in the 100 km × 100 km area scenario (lower is better)

In the OOI scenario, we assign prizes to each sensor based on the calculated variance of a Gaussian Process (GP) [21]. A GP is a non-parametric Bayesian inference technique that can be used to estimate a quantity of interest and also predict the variance of those estimates. In other words, the output of the GP is both an estimate of a quantity (e.g., temperature, salinity, chlorophyl, etc.) over a spatial area and an uncertainty in this estimate at each point in the space. The variance of a GP has been widely used in the adaptive sampling literature as a metric for sensing performance (i.e., reducing the variance corresponds to a better estimate of a quantity of interest over a spatial field) [22].

In the OOI deployments, we input simulated sensor data of ocean temperature from the Regional Ocean Modeling System [23] into a GP using a squared exponential kernel. Hyperparameters were learned using standard conjugate gradient ascent techniques on the marginal likelihood. The resulting variance was then used as the prizes for the deployment in Fig. 1, which yields an instance of the PC-TSPN problem. Fig. 4 demonstrates that the proposed approach provides substantial improvement over the PC-TSPN heuristic in this real-world monitoring scenario.

An example of found solutions is depicted in Fig. 5. Notice that data from all sensors are collected, and the performance improvement of the proposed approach is in the determination of new goal locations. In contrast, the deterministic PC-TSPN algorithm considers only the sensor locations as part of the tour. The red discs denote the most important measurements, which are located at the very far part of the area.

C. Discussion

Regarding the presented results, the proposed SOM-based approach provides superior results and outperforms the heuristic PC-TSPN algorithm in almost all the problem instances examined. Inferior solutions are provided in problems with high penalties and small communication radii. In these setups, the problem is close to standard TSP, where SOM performance is worse than the optimal solution. This drawback is addressed by the proposed combination of SOM+TSP, at the cost of increased computational requirements.

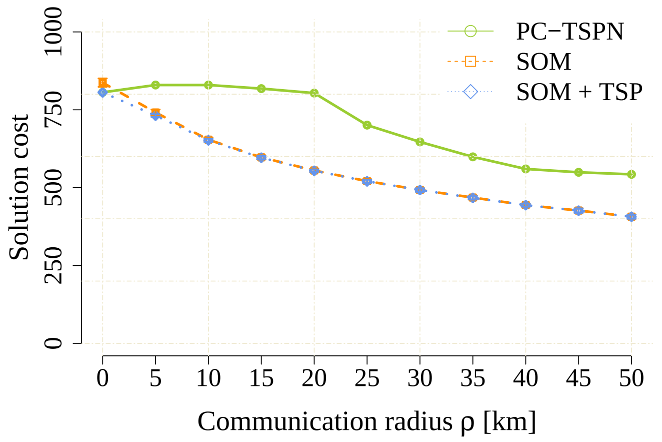
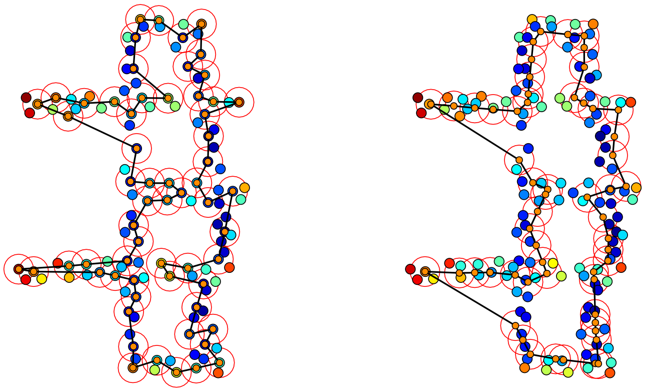


Fig. 4. Comparison of the proposed SOM algorithm to the PC-TSPN heuristic in the OOI data collection scenario (see Fig. 1) with penalties corresponding to the variance of a Gaussian Process

It is also worth mentioning that for low penalties and high neighborhood size, the found solutions are small tours that cover an arbitrary subset of the sensors. When this occurs, it may indicate that the penalties are not set appropriately. Based on our observations, we believe this behavior is also related to the assumptions that the penalties are static and the sensor measurements are independent. The low computational requirements of the proposed approach allow for the possibility that the penalties can be computed online during the learning process, which could further improve the solution.

Differences in the solutions provided by the SOM and



(a) PC-TSPN [18], $C = 647$ (b) Proposed SOM, C=484

Fig. 5. An example of found solutions in the OOI deployment scenario and neighborhood size $\rho = 6$ km. The sensor stations are shown as large discs that are filled by the color according to its penalty, where red denotes high penalty and blue a low penalty value. The found tour is shown as the black straight line segments connecting the determined goal locations that are represented by small orange discs.

SOM+TSP approaches are small, and the only noticeable improvement of the optimal solution of the TSP is for zero neighborhood size and high penalty. Therefore, regarding the results, it seems that selection of appropriate goals is more important than finding optimal tour. The results also indicate that in the addressed problems, the SOM provides a competitive solution to the underlying TSP for $\rho > 0$. Differences between a pure SOM-based TSP solution and the optimal solution are often negligible in comparison to the provided improvement gained by not visiting all sensors.

VI. Conclusions and Future Work

In this paper, we proposed a unifying approach for solving multi-goal path planning problems arising from data collection tasks where it is necessary to consider penalties on the goals and neighborhoods surrounding the goals. Based on the evaluation, the proposed SOM-based algorithm outperforms currently available combinatoric heuristic approaches and provides improved solutions in all the TSPN, PC-TSP, and PC-TSPN problems examined. The proposed approach also achieves substantially lower computational requirements.

Even though combinatoric heuristics for the TSP provide better results in the standard TSP than the discussed SOM method, the main source of solution improvement in the data collection tasks relies on the selection of the appropriate goal locations and consideration of the communication radius during planning. Thus, based on the current results, such an error seems to be negligible. Besides, the final tour over the determined goal locations can always be improved by solving the related TSP optimally (at the cost of higher computational requirements).

During the performance evaluation, we have found out that the main difficulty in solving the data collection problem in the desired way relies on designing appropriate penalties regarding the distance cost between the sensor stations. This is because the penalties are domain specific and depend on the particular phenomena being studied by the sensors. In addition, the current formulation of the problem assumes the sensors provide independent measurements. To overcome this drawback, our future work will consider an on-line determination of sensor penalties during the unsupervised learning stage.

References

[1] M. Kohler, P. Lynett, P. Lynett, M. Legg, and D. Weeraratne, “Importance of large-scale bathymetry features on 2011 Tohoku tsunami waveforms through comparison of simulations with the spatially dense ALBACORE OBS array data,” in *Proc. AGU Fall Meeting*, 2012.

[2] G. Hollinger, S. Choudhary, P. Qarabaqi, C. Murphy, U. Mitra, G. Sukhatme, M. Stojanovic, H. Singh, and F. Hover, “Underwater data collection using robotic sensor networks,” *IEEE J. Selected Areas in Communications*, vol. 30, no. 5, pp. 899–911, 2012.

[3] W. Ramos. (2014, Feb.) Ocean observatories initiative. [Online]. Available: http://oceanobservatories.org/

[4] J. Faigl, M. Kulich, V. Vonásek, and L. Přeučil, “An Application of Self-Organizing Map in the non-Euclidean Traveling Salesman Problem,” *Neurocomputing*, vol. 74, no. 5, pp. 671–679, 2011.

[5] J. Faigl and L. Přeučil, “Self-Organizing Map for the Multi-Goal Path Planning with Polygonal Goals,” in *ICANN*, 2011, pp. 85–92.

[6] D. Applegate, R. Bixby, V. Chvátal, and W. Cook, *The Traveling Salesman Problem: A Computational Study*. Princeton, NJ, USA: Princeton University Press, 2007.

[7] A. Dumitrescu and J. Mitchell, “Approximation algorithms for TSP with neighborhoods in the plane,” *J. Algorithms*, vol. 48, no. 1, pp. 135–159, 2003.

[8] M. de Berga, J. Gudmundssonb, M. J. Katzc, C. Levcopoulosd, M. H. Overmarse, and A. F. van der Stappen, “TSP with neighborhoods of varying size,” *J. Algorithms*, vol. 57, no. 1, pp. 22–36, 2005.

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

[10] G. Ausiello, V. Bonifaci, S. Leonardi, and A. Marchetti-Spaccamela, “Prize-Collecting Traveling Salesman and related problems,” in *Handbook of Approximation Algorithms and Metaheuristics*, T. F. Gonzalez, Ed. CRC Press, 2007.

[11] M. Goemans and D. P. Williamson, “A general approximation technique for constrained forest problems,” *SIAM J. Computing*, vol. 24, no. 2, pp. 296–317, 1995.

[12] 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.

[13] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke, “Data collection, storage, and retrieval with an underwater sensor network,” in *Proc. Int. Conf. Embedded Networked Sensor Systems*, 2005, pp. 154–165.

[14] M. Tucci and M. Raugi, “Stability analysis of self-organizing maps and vector quantization algorithms,” in *IJCNN*, 2010, pp. 1–5.

[15] P. Janoušek and J. Faigl, “Speeding up coverage queries in 3d multi-goal path planning,” in *ICRA*, 2013, pp. 5067–5072.

[16] T. Yamakawa, K. Horio, and M. Hoshino, “Self-Organizing Map with Input Data Represented as Graph,” in *Neural Information Processing*. Springer Berlin / Heidelberg, 2006, pp. 907–914.

[17] J. Faigl and G. Hollinger, “SOM for the PC-TSPN,” [cited 10 Jun 2014]. [Online]. Available: http://purl.org/faigl/pctspn

[18] G. Hollinger, U. Mitra, and G. Sukhatme, “Autonomous data collection from underwater sensor networks using acoustic communication,” in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2011, pp. 3564–3570.

[19] D. Applegate, R. Bixby, V. Chvátal, and W. Cook, “CONCORDE TSP Solver,” 2003, [cited 20 Oct 2013]. [Online]. Available: http://www.tsp.gatech.edu/concorde.html

[20] L. Freitag, M. Stojanovic, M. Grund, and S. Singh, “Acoustic communications for regional undersea observatories,” in *Proc. Oceanology International*, London, UK, 2002.

[21] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[22] G. Hollinger, B. Englot, F. Hover, U. Mitra, and G. Sukhatme, “Active planning for underwater inspection and the benefit of adaptivity,” *Int. J. Robotics Research*, vol. 32, no. 1, pp. 3–18, 2013.

[23] A. F. Shchepetkin and J. C. McWilliams, “The Regional Oceanic Modeling System (ROMS): a split-explicit, free-surface, topography-following-coordinate oceanic model,” *Ocean Modelling*, vol. 9, no. 4, pp. 347–404, 2005.