

# Efficient Communication-Aware Dynamic Coverage Using Space-Filling Curves

Yuan Yan and Yasamin Mostofi

**Abstract**—In this paper, we consider a dynamic coverage problem where a team of mobile robots needs to cover a set of time-varying Points of Interest (POIs), while maintaining their connectivity to a remote station in a realistic communication environment. Our goal is to design the motion and communication strategies of the robots such that they periodically visit the POIs, communicate the gathered information to a remote station, minimize their total energy costs, including motion, sensing and communication energy, and satisfy other system constraints. In our previous work [1], we have shown how to pose this problem as a Mixed Integer Linear Program (MILP), which is computationally expensive. In this paper, we focus on designing a considerably more computationally-efficient heuristic approach to tackle this problem. More specifically, we propose to utilize the space-filling curves to efficiently assign the POIs and plan the trajectories of the robots. Under certain conditions, we mathematically show that our heuristic approach is at most a constant factor away from the global optimum. Our simulation results then confirm that our approach is considerably faster than solving the MILP, especially when the dimension of the problem is high. They further show that the energy consumption of our approach is only around 26% more than the optimum.

## I. INTRODUCTION

In recent years, considerable progress has been made in the area of mobile sensor networks and networked robotic systems [2]–[9]. One emerging application is to deploy a group of mobile robots to cover a spatially-large environment and collect the sensed information of a number of static sensors that do not have a large enough communication range [1], [8], [10], [11]. In a spatially-large environment, the coverage task cannot typically be achieved by any static configuration of the robotic team due to the limited sensing and communication ranges of the robots. Hence, dynamic motion, sensing and communication strategies need to be planned to accomplish the goal.

In this paper, we consider a dynamic coverage problem in robotic networks where a team of mobile robots needs to cover a set of Points of Interest (POIs) in a time-varying environment and communicate the gathered information to a remote station in an online manner. By a time-varying environment, we refer to an environment where the quantity of interest at each POI is time-varying and thus needs to be persistently monitored by a mobile robot. Our focus is then to minimize the total energy cost of the robots, including motion, sensing and communication energy costs in a realistic

fading communication environment, while guaranteeing that 1) each POI is assigned to one of the robots, and 2) a periodic trajectory is designed for each robot to keep the quantity of interest at every POI bounded. We adopt the linear model used in [1] and [11] to characterize the dynamics of the quantity of interest at each POI.

In [1], we considered this communication-aware and energy-aware dynamic coverage problem and showed how the whole problem can be posed as a Mixed Integer Linear Program (MILP). However, solving an MILP is computationally expensive or not possible, especially when the dimension of the program is high. Hence, in this paper we aim to design a fast heuristic approach to solve this problem. More specifically, we utilize the space-filling curves [12] for POI assignment and trajectory design, which significantly reduces the computational complexity. A space-filling curve is a one-dimensional curve that passes through all the points of a 2D square. By utilizing space-filling curves and their locality property, we show how to reduce the computational complexity of our considered problem significantly. Furthermore, under certain conditions, we mathematically show that our approach is at most a constant factor away from the global optimum. Our simulation results further confirm that our approach is considerably faster than solving the MILP, especially when the dimension of the problem is high. Moreover, they show that the average performance of our approach is only around 26% away from the optimum.

The rest of the paper is organized as follows. Section II describes the motion and communication models, and introduces the basic properties and applications of the space-filling curves. Section III presents our proposed approach and mathematically shows its performance bound. Section IV shows the performance of the proposed approach in a simulation environment. We conclude in Section V.

## II. PROBLEM SETUP

Consider a scenario where a remote station (control center) is tasked with utilizing a team of  $n$  mobile robots to cover a set of  $m$  POIs  $\mathcal{P} = \{p_1, \dots, p_m\}$  in a square obstacle-free workspace<sup>1</sup>  $\mathcal{W} \subset \mathbb{R}^2$ . The remote station maintains a quantity of interest  $\Psi_i(t)$ , for  $i \in \{1, \dots, m\}$ . For instance,  $\Psi_i(t)$  can represent the uncertainty of the state of the  $i$ th POI at the remote station. See [1], [11] for more details on  $\Psi_i(t)$ . In this paper, we say a POI is covered by a robot if the robot senses the POI with its onboard sensor while maintaining connectivity to the remote station to report the

Yuan Yan and Yasamin Mostofi are with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, USA (email: {yuanyan,ymostofi}@ece.ucsb.edu).

This work is supported in part by the NSF CAREER award # 0846483.

<sup>1</sup>Our approach can be easily extended to any convex workspace.

task online. We assume a time-varying workspace, where  $\Psi_i(t)$  increases at a certain rate if the  $i$ th POI is not being covered by any mobile robot. If the  $i$ th POI is covered by a mobile robot,  $\Psi_i(t)$  then decreases at a rate that depends on the onboard sensing capability of the robot.

In this paper we make the following assumptions. We assume that the sensing ranges of the robots are negligible as compared to the size of the workspace. As a result, the robots need to physically visit each POI to sense the corresponding quantity of interest. Also, since we require each robot to report the sensed quantity to the remote station online, the robot only transmits at the positions of the POIs. Therefore, we have the following dynamics for  $\Psi_i(t)$  [1], [11]:<sup>2</sup>  $\dot{\Psi}_i(t) = \mathbb{I}(\Psi_i(t) \geq 0) \left( \rho_i - \mathbb{I}(\xi_k(t) = p_i) \lambda_k(t) \alpha_{i,k} \right)$ , where  $\mathbb{I}(\cdot)$  denotes the indicator function,  $\xi_k(t)$  is the position of the  $k$ th robot at time  $t$ ,  $\lambda_k(t)$  is a binary variable which is one if the  $k$ th robot is connected to the remote station at time  $t$  and is zero otherwise,  $\rho_i$  is the constant increasing rate of  $\Psi_i(t)$ , and  $\alpha_{i,k}$  is the constant decreasing rate of  $\Psi_i(t)$  if the  $k$ th robot is sensing the  $i$ th POI.

The robot has limited energy budget for its operation. Our goal is then to 1) assign each mobile robot to a nonempty subset of POIs, 2) design a closed periodic trajectory for each robot to repeatedly cover the POIs that are assigned to it, and 3) minimize the total energy consumption of the robots in one period, which includes communication, sensing and motion energy costs, while guaranteeing that the quantity of interest  $\Psi_i(t)$ , for  $i \in \{1, \dots, m\}$ , which is maintained at the remote station, remains bounded (stabilized) all the time.

Fig. 1 shows the considered scenario. As can be seen, two mobile robots are used to cover a set of POIs. Each robot follows a periodic trajectory to cover its assigned POIs. Moreover, the quantity of interest  $\Psi_i(t)$  for each POI remains bounded.

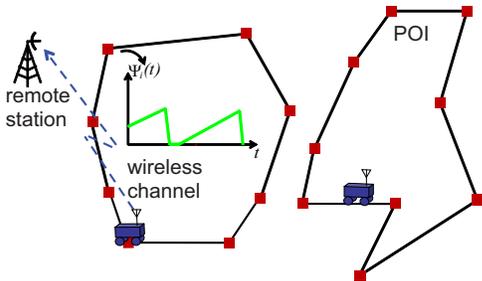


Fig. 1. Dynamic coverage of a time-varying environment using two mobile robots.  $\Psi_i(t)$  is the quantity of interest to keep bounded for the  $i$ th POI.

In [1], we have shown how this problem can be formulated as an MILP and be solved by using available solvers such as IBM ILOG CPLEX [13]. Since the MILP can only be solved relatively efficiently when the dimension of the problem is low, i.e. the number of the robots and the POIs is small, one significant disadvantage of the proposed approach in [1] is its high computational complexity when solving high dimensional problems. In this paper, we then focus

<sup>2</sup>Note that the quantity of interest may have nonlinear dynamics in reality. Then, this simplified linear model can be considered as its first-order approximation [11].

on designing a fast approach based on using space-filling curves, which can efficiently solve this problem even when the dimension of the problem is high, at the cost of a small loss of performance, as we characterize.

In this section, we first introduce the concept of space-filling curves which are used to design our heuristic approach. Then, we briefly discuss the communication and motion cost models that are used in this paper. In Section III, we then show how our dynamic coverage problem can be significantly simplified by using the space-filling curves.

### A. Space-Filling Curves

In the late 19th century, Cantor showed a counter-intuitive fact that any finite 2D square (or 3D cube) does not contain more points than a unit interval. After that, a novel question was raised: Is there any continuous one-dimensional curve that passes through every point of a two-dimensional square (or a three-dimensional cube)? During the late 19th century and early 20th century, mathematicians discovered many of these curves. Curves with this property were named space-filling curves [12]. Some of the most celebrated ones are the Hilbert curve, Peano curve and Sierpiński curve. Fig. 2 shows an example of a recursive construction of the Sierpiński curve. It can be seen that, as the recursion continues, the curve covers more and more points in the square. By taking the limit of this recursion process, the curve finally covers all the points. In practice, a few steps of recursion is typically accurate enough to approximate a discrete space. Readers are referred to [12] for more details on various space-filling curves and how to construct them.

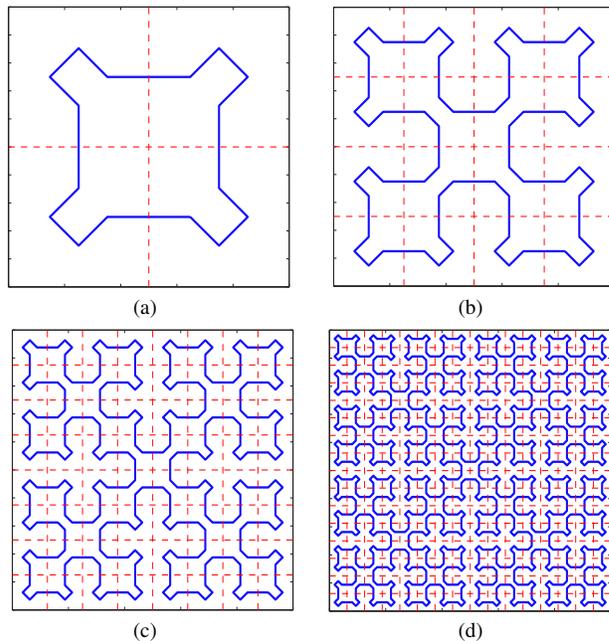


Fig. 2. An example of a recursive construction of the Sierpiński curve. (a) – (d) show the first through the fourth step of the recursion respectively. Sierpiński curve can then be obtained by taking the limit of the recursion.

Because of their recursive and self-similar construction, one of the most important properties of space-filling curves is the *locality* property, which means that any two points that are close in the one-dimensional space are mapped to two

points that are close in the higher dimensional space. More specifically, let  $\text{SF}(\cdot) : \mathcal{C} \rightarrow [0, S]^2$  denote the continuous mapping of a space-filling curve from the unit circle<sup>3</sup> to the square whose length of side is equal to  $S$ , where  $\mathcal{C} = \{x|0 \leq x \leq 1\}$  represents the unit circle. Then, we have [12], [14]

$$\|\text{SF}(x) - \text{SF}(y)\| \leq C_{\text{SF}} \times S \times \mu(x, y), \quad (1)$$

where  $x, y \in \mathcal{C}$ ,  $C_{\text{SF}}$  is a constant depending on the type of the space-filling curve,  $\mu(x, y) = \min\{|x-y|, 1-|x-y|\}^{1/2}$ , and  $\|\cdot\|$  and  $|\cdot|$  denote the Euclidean norm and absolute value of the argument respectively. See Fig. 3 for an illustration of the mapping. For Sierpiński curve of Fig. 2, for instance,  $C_{\text{SF}} = 2$  [12], [14].

Due to this *locality* property, space-filling curves are widely used in computational science [12]. The applications that are most related to this paper are solving the Traveling Salesman Problem (TSP) [14]–[16] and the Vehicle Routing Problem (VRP) [17], [18]. For instance, the TSP problem can be simply solved as follows [14]. First, map all the POIs in the square into the unit circle using  $\text{SF}^{-1}(\cdot)$ . Then, order the mapped POIs on the unit circle in a clockwise or counter-clockwise direction. Finally, build the tour by connecting the POIs in the square based on this order. Fig. 3 shows an example of solving the TSP problem by using this approach. When the POIs are i.i.d. and the number of POIs becomes large, the heuristic tour is roughly 35% away from the optimum [16]. However, the computational complexity of this approach is extremely low. In Section III, we will use this idea to design periodic trajectories for communication-aware and energy-aware optimization of the robots.

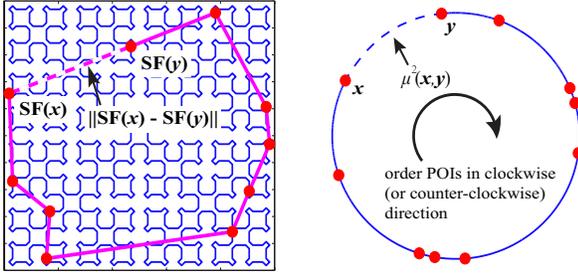


Fig. 3. An example of using the Sierpiński curve to solve the TSP problem with 10 POIs. The red dots in the left and right figures show the positions of the POIs in the square and their corresponding mapping in the unit circle respectively. The line segments that connect the POIs in the square form the tour obtained based on ordering of the POIs in the unit circle. The figure also shows an illustration of the mapping  $\text{SF}(\cdot)$ .

### B. Communication Model

In this paper we assume that the robot is connected to the remote station if the received signal-to-noise ratio (SNR) at the remote station is above a required threshold  $\text{SNR}_{\text{th}}$ . More specifically,  $\lambda_k(t) = \begin{cases} 1 & \text{if } P_{\text{TX},k}(t)\gamma(t) \geq \text{SNR}_{\text{th}}, \\ 0 & \text{otherwise,} \end{cases}$  for  $k \in \{1, \dots, n\}$ , where  $P_{\text{TX},k}(t)$  denotes the transmit power of the  $k$ th robot at time  $t$  and  $\gamma(t)$  is the channel-to-noise ratio (CNR) from the robot to the remote station at time

<sup>3</sup>Note that the domain of  $\text{SF}(\cdot)$  for closed space-filling curves, such as the Sierpiński curve of Fig. 2, is the unit circle. However, for open space-filling curves, such as Hilbert curve, the domain is the unit interval.

$t$ . Then, the minimum required communication power to maintain the connectivity becomes  $\text{SNR}_{\text{th}}/\gamma(t)$ . Note that  $\lambda_k(t)$ ,  $P_{\text{TX},k}(t)$  and  $\gamma(t)$  are also functions of the position of the  $k$ th robot. For example,  $\gamma(t) = \gamma(\xi_k(t))$ , where  $\gamma(q)$  denotes the CNR at position  $q \in \mathcal{W}$ .

For the purpose of planning, the robots need to know the channel at the POIs. In practice, however, we may only have a few CNR measurements available in the workspace as opposed to the whole channel map. Our previously-proposed probabilistic channel assessment framework [19], [20] can then be used to estimate the channel quality at unvisited locations, based on a small number of a priori channel measurements in the same workspace. More specifically, the channel quality at  $q$  (in dB) can be characterized by a Gaussian random variable,  $\Upsilon_{\text{dB}}(q)$ , with the mean of  $\bar{\Upsilon}_{\text{dB}}(q)$  and the variance of  $\sigma_{\text{dB}}^2(q)$ . See [19], [20] for more details on the expressions of  $\bar{\Upsilon}_{\text{dB}}(q)$  and  $\sigma_{\text{dB}}^2(q)$ , and the performance of this framework with real data and in different environments.

Since we are assessing the channel quality probabilistically, the minimum required communication power to maintain the connectivity also becomes a random variable. In this paper, we then assume that the  $k$ th robot is connected to the remote station if  $P_{\text{TX},k}(\xi_k(t))\Upsilon(\xi_k(t)) \geq \text{SNR}_{\text{th}}$  is satisfied with a high probability, i.e.  $\mathbb{P}\{P_{\text{TX},k}(\xi_k(t))\Upsilon(\xi_k(t)) \geq \text{SNR}_{\text{th}}\} \geq \epsilon$ , where  $\Upsilon(\xi_k(t)) = 10^{\Upsilon_{\text{dB}}(\xi_k(t))/10}$  denotes the estimated channel quality at  $\xi_k(t)$  in the non-dB domain,  $\mathbb{P}\{\cdot\}$  is the probability of the argument, and  $\epsilon$  is the required probability threshold. Equivalently, we have  $P_{\text{TX},k}(\xi_k(t)) \geq 10^{-\sigma_{\text{dB}}(\xi_k(t))Q^{-1}(\epsilon)/10}\text{SNR}_{\text{th}}/\bar{\Upsilon}(\xi_k(t))$ , where  $\bar{\Upsilon}(\xi_k(t)) = 10^{\bar{\Upsilon}_{\text{dB}}(\xi_k(t))/10}$  and  $Q^{-1}(\cdot)$  denotes the inverse of the Q function.

As mentioned previously, we consider the case where the robot only communicates to the remote station at the POIs. Moreover, we assume that each mobile robot needs a fixed time duration  $t_c$  at each POI to transmit the updated information. Then, the minimum required communication energy cost of the  $k$ th robot at the  $i$ th POI is  $P_{\text{TX},\min,i}t_c$ , where  $P_{\text{TX},\min,i} = 10^{-\sigma_{\text{dB}}(p_i)Q^{-1}(\epsilon)/10}\text{SNR}_{\text{th}}/\bar{\Upsilon}(p_i)$ .

### C. Motion Model

We consider a first-order motion power model for the mobile robot [21]:  $P_{\text{mo}} = \begin{cases} \kappa_1 v + \kappa_2 & \text{if } 0 < v \leq v_{\text{max}}, \\ 0 & \text{if } v = 0, \end{cases}$  where  $P_{\text{mo}}$  is the motion power,  $v$  denotes the velocity of the robot,  $\kappa_1$  and  $\kappa_2$  are positive constants, and  $v_{\text{max}}$  is the maximum velocity of the robot. This model is a very good fit to the Pioneer 3DX robot when the velocity is smaller than 0.9 m/s [21]. Then, the motion energy consumption for moving from  $q_1$  to  $q_2$  can be found as follows:  $E_{\text{mo}}(q_1, q_2) = \kappa_1 \|q_1 - q_2\| + \kappa_2 t_{\text{mo}}$ , where  $t_{\text{mo}}$  is the total motion time. Clearly,  $E_{\text{mo}}(q_1, q_2)$  is minimized only if  $v = v_{\text{max}}$ , i.e.  $t_{\text{mo}} = \|q_1 - q_2\|/v_{\text{max}}$ . Hence, we have the following motion energy cost model for the mobile robot:  $E_{\text{mo}}(q_1, q_2) = \kappa_{\text{M}} \|q_1 - q_2\|$ , where  $\kappa_{\text{M}} = \kappa_1 + \kappa_2/v_{\text{max}}$ . It can be seen that  $E_{\text{mo}}(q_1, q_2)$  is linearly increasing with respect to the traveling distance.

### III. OUR PROPOSED APPROACH BASED ON SPACE-FILLING CURVES

In this section, we first briefly discuss the MILP formulation that results from posing our coverage problem, as shown in [1]. Then, we show how to utilize space-filling curves to solve this problem considerably more efficiently. Finally, under certain conditions (e.g. for homogeneous POIs and robots, and large time and energy budgets), we show that our approach is at most a constant factor away from the global optimum, for which we find a mathematical expression.

#### A. MILP formulation

Let  $\mathcal{V} = \{1, \dots, m\}$  represent the set of indices of all the POIs, and  $\mathcal{V}_k$ , for  $k \in \{1, \dots, n\}$ , denote the set of indices of the POIs that are assigned to the  $k$ th mobile robot. In this paper we consider the case where  $\{\mathcal{V}_k\}_{k=1}^n$  is a partition of  $\mathcal{V}$ , i.e.  $\mathcal{V}_i \cap \mathcal{V}_j = \emptyset$ , for any  $i \neq j$ , and  $\bigcup_{k=1}^n \mathcal{V}_k = \mathcal{V}$ . In other words, we assume that each POI is assigned to one and only one robot. Also, as mentioned previously, since we are interested in designing a periodic trajectory for each robot, a Hamiltonian cycle  $\mathcal{H}_k$  needs to be found on each  $\mathcal{V}_k$ .

Define binary variables  $x_{i,k}$  and  $z_{i,j,k}$ , for  $i, j \in \mathcal{V}$  and  $k \in \{1, \dots, n\}$ . Let  $x_{i,k} = 1$  if the  $i$ th POI is assigned to the  $k$ th mobile robot, and  $x_{i,k} = 0$  otherwise. Also,  $z_{i,j,k} = 1$  if there exists an edge between the  $i$ th and  $j$ th POIs in the Hamiltonian cycle assigned to the  $k$ th mobile agent, and  $z_{i,j,k} = 0$  otherwise. Then, we have the following MILP:

$$\begin{aligned}
\min \quad & E_{\text{tot}} = \sum_{k=1}^n E_k \quad (2) \\
\text{s.t.} \quad & \\
1) \quad & \alpha_{i,k} t_{i,k} - \rho_i \left( \sum_{i=1}^m t_{i,k} + \frac{1}{v_{\text{max}}} \sum_{i=1}^m \sum_{j=1, j \neq i}^m z_{i,j,k} d_{i,j} \right) \\
& \quad + (1 - x_{i,k}) \Omega \geq 0, \quad \forall i, k, \\
2) \quad & \sum_{i=1}^m t_{i,k} + \frac{1}{v_{\text{max}}} \sum_{i=1}^m \sum_{j=1, j \neq i}^m z_{i,j,k} d_{i,j} \leq T_{\text{max}}, \quad \forall k, \\
3) \quad & P_s \sum_{i=1}^m t_{i,k} + \kappa_M \sum_{i=1}^m \sum_{j=1, j \neq i}^m z_{i,j,k} d_{i,j} \\
& \quad + t_c \sum_{i=1}^m x_{i,k} P_{\text{TX}, \min, i} = E_k, \quad \forall k, \\
4) \quad & \sum_{j=1, j \neq i}^m z_{i,j,k} = x_{i,k}, \quad \sum_{j=1, j \neq i}^m z_{j,i,k} = x_{i,k}, \quad \forall i, k, \\
5) \quad & \sum_{k=1}^n x_{i,k} = 1, \quad \forall i, \quad 6) \quad \sum_{i=1}^m e_i = n, \\
7) \quad & 2 - e_i \leq u_i \leq m, \quad \forall i, \quad 8) \quad u_i - u_j - m(e_i + e_j) \\
& \quad + (m-1) \sum_{k=1}^n z_{i,j,k} \leq m-2, \quad \forall i, j \neq i, \\
9) \quad & 0 \leq t_{i,k} \leq x_{i,k} \Omega, \quad \forall i, k, \quad 10) \quad E_k \leq E_{\text{max}, k}, \quad \forall k, \\
11) \quad & z_{i,j,k}, x_{i,k}, e_i \in \{0, 1\}, \quad u_i \in \mathbb{N}, \quad \forall i, j, k.
\end{aligned}$$

Note that the objective function of (2) is the total energy cost of the mobile robots in one period, where  $E_k$  denotes the energy cost of the  $k$ th robot. Constraint 1 guarantees that the quantity of interest at each POI is bounded, where  $t_{i,k}$  is the time duration that the  $k$ th robot spends at the  $i$ th POI,  $d_{i,j} = \|p_i - p_j\|$  is the Euclidean distance between  $p_i$  and  $p_j$ , where  $\Omega$  is any sufficiently-large constant that satisfies  $\Omega \geq T_{\text{max}} \max_i \rho_i$ . Note that the third term in constraint 1 ensures that the constraint can always be satisfied if  $x_{i,k} = 0$ . Constraint 2 ensures that the period of each robot is smaller than or equal to an upper bound  $T_{\text{max}}$ . Constraint 3 denotes the energy cost of each robot, where  $P_s$  is the power cost of sensing when the robot is covering any POI and  $t_c$  is a fixed communication time. We assume that  $t_c$  is small as compared to  $t_{i,k}$ s. Constraint 4 guarantees that each POI has exactly one degree in and one degree out. Constraint 5 forces each POI to be assigned to one and only one mobile robot. Constraints 6-8 are Miller-Tucker-Zemlin (MTZ) subtour elimination constraints [1], [22], where  $e_i$  and  $u_i$  are auxiliary variables. Constraint 9 guarantees that  $t_{i,k} = 0$  if the  $i$ th POI is not assigned to the  $k$ th robot. Finally, constraint 10 forces the energy cost of the  $k$ th robot in each period to be bounded from above by its maximum energy budget  $E_{\text{max}, k}$ . See [1] for more details on how this MILP is formulated.

In the rest of the paper, we use superscript  $\star$  to denote the optimal solution of the corresponding optimization problem. Next, we present a property of (2) that will be used in the subsequent sections.

*Theorem 1 ([1]):* Consider a partition  $\{\mathcal{V}_k\}_{k=1}^n$  of  $\mathcal{V}$  and the corresponding Hamiltonian cycles  $\{\mathcal{H}_k\}_{k=1}^n$ . Then, the stability condition (constraint 1 in (2)) can be satisfied if and only if  $\phi_k = \sum_{i \in \mathcal{V}_k} \rho_i / \alpha_{i,k} < 1$ , for all  $k \in \{1, \dots, n\}$ . Moreover, if  $\phi_k < 1$ , the optimal total time and energy costs of the  $k$ th robot in one period (given Hamiltonian cycle  $\mathcal{H}_k$ ) can be characterized as follows:

$$T_k^*(\mathcal{H}_k) = \frac{1}{1 - \phi_k} \frac{d(\mathcal{H}_k)}{v_{\text{max}}}, \quad (3)$$

$$\begin{aligned}
E_k^*(\mathcal{H}_k) &= E_{\text{sen}, k}^*(\mathcal{H}_k) + E_{\text{mo}, k}^*(\mathcal{H}_k) + E_{\text{comm}, k} \\
&= \frac{P_s \phi_k}{1 - \phi_k} \frac{d(\mathcal{H}_k)}{v_{\text{max}}} + \kappa_M d(\mathcal{H}_k) + t_c \sum_{i \in \mathcal{V}_k} P_{\text{TX}, \min, i}, \quad (4)
\end{aligned}$$

where function  $d(\cdot)$  denotes the total distance of the argument in the 2D workspace,  $T_k^*(\mathcal{H}_k)$  denotes the total time cost of  $k$ th robot in one period, and  $E_{\text{sen}, k}^*(\mathcal{H}_k) = P_s \phi_k d(\mathcal{H}_k) / ((1 - \phi_k) v_{\text{max}})$ ,  $E_{\text{mo}, k}^*(\mathcal{H}_k) = \kappa_M d(\mathcal{H}_k)$  and  $E_{\text{comm}, k} = t_c \sum_{i \in \mathcal{V}_k} P_{\text{TX}, \min, i}$  are the sensing, motion and communication energy costs for the  $k$ th robot respectively.

Note that (3) and (4) hold for any stabilizing partition and its corresponding Hamiltonian cycles. Hence, we use notations  $T_k^*(\mathcal{H}_k)$ ,  $E_k^*(\mathcal{H}_k)$ ,  $E_{\text{sen}, k}^*(\mathcal{H}_k)$  and  $E_{\text{mo}, k}^*(\mathcal{H}_k)$  to indicate that these parameters are all functions of the Hamiltonian cycle  $\mathcal{H}_k$ . In order to minimize the total energy cost  $E_{\text{tot}}(\{\mathcal{H}_k\}_{k=1}^n) = \sum_{k=1}^n E_k^*(\mathcal{H}_k)$ , we need to further find the optimal partition  $\{\mathcal{V}_k^*\}_{k=1}^n$  and the corresponding minimum-length Hamiltonian cycles  $\{\mathcal{H}_k^*\}_{k=1}^n$ . In (2), this problem is tackled by imposing mixed integer linear constraints, which makes the problem computationally expen-

sive. Intuitively, the optimal partition should assign the POIs that are close to each other to one robot. This inspires us to utilize the space-filling curves, which can be used to find  $\{\mathcal{V}_k\}_{k=1}^n$  and  $\{\mathcal{H}_k\}_{k=1}^n$  efficiently and significantly simplify the problem.

### B. Our Proposed Approach Based on Space-Filling Curves

As mentioned in Section II-A, the position of each POI,  $p_i$ , can be mapped to the unit circle  $\mathcal{C}$  by using  $\text{SF}^{-1}(p_i)$ . Then, instead of finding the optimal partition  $\{\mathcal{V}_k\}_{k=1}^n$  and the corresponding Hamiltonian cycles  $\{\mathcal{H}_k\}_{k=1}^n$  in square  $[0, S]^2$ , we can simplify the problem by solving it in  $\mathcal{C}$ .

Let  $\{\mathcal{V}_{\text{SF},k}\}_{k=1}^n$  and  $\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n$  denote the partition and the corresponding Hamiltonian cycles that are found in  $[0, S]^2$  respectively, based on our approach in  $\mathcal{C}$ . Then, given a partition  $\{\mathcal{V}_{\text{SF},k}\}_{k=1}^n$ ,  $\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n$  can be easily found by ordering the corresponding POIs in  $\mathcal{C}$ , and translating this ordering back to  $[0, S]^2$ . Note that as mentioned in Section II-A, the resulting  $\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n$  may not necessarily be the minimum-length Hamiltonian cycles. Next, we propose a strategy to find the partition  $\{\mathcal{V}_{\text{SF},k}\}_{k=1}^n$ . As mentioned in Section II-A, one of the most attractive properties of the space-filling curves is the *locality* property. Thus, in order to minimize the length of the Hamiltonian cycle for each robot, it is intuitive to group the POIs that are ‘‘close’’ together in  $\mathcal{C}$ . Hence, we partition  $\mathcal{V}$  such that, if two POIs in  $\mathcal{C}$  are grouped together, then all the POIs in between them should belong to the same subset. Fig. 4 shows examples of how the POIs are partitioned.

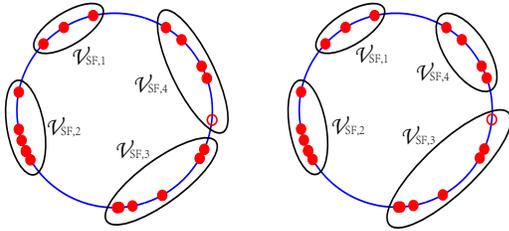


Fig. 4. Examples to show how the POIs are partitioned in  $\mathcal{C}$ . The filled and empty circles represent the POIs, which are assigned to four robots. The empty circle in the left figure denotes a boundary POI of  $\mathcal{V}_{\text{SF},4}$ .

Based on the partition strategy above, we propose Algorithm 1 to minimize  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)$ .<sup>4</sup> Note that in the second step of the algorithm, the choice of the initial partition is not unique. It can be any partition that satisfies our locality grouping condition. In the fourth step,  $\Omega_i$ , for  $i \in \{1, \dots, 4\}$ , are any sufficiently-large constants satisfying the following conditions:  $\Omega_1 > \sum_{k=1}^n E_{\text{max},k}$ ,  $\Omega_2 > \max_k E_{\text{max},k}/(\kappa_M \min_{i,j} d_{i,j})$ ,  $\Omega_3 > \max_k E_{\text{max},k}/(\min_{i,j} d_{i,j}/v_{\text{max}})$  and  $\Omega_4 > \max_k E_{\text{max},k}/(\min_{i,k} \rho_i/\alpha_{i,k})$ . The goal of the fourth step is to guarantee that the algorithm will always seek for feasible solutions. In the sixth step, we say  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)$  is converged if it cannot be decreased anymore by the process from the seventh through the twelfth steps. Note that since  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)$  is bounded

<sup>4</sup>Note that  $\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n$  are the Hamiltonian cycles found in  $[0, S]^2$ . Hence,  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)$  can be calculated by simply substituting  $\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n$  into  $E_{\text{tot}}(\{\mathcal{H}_k\}_{k=1}^n)$ .

from below by the global optimum, this repeating process is guaranteed to stop after some steps of iterations. In the seventh step of the algorithm, the boundary POIs of  $\mathcal{V}_{\text{SF},k}$  can be found by calculating  $\max_{i \in \mathcal{V}_{\text{SF},k}} \{\text{SF}^{-1}(p_i)\}$  and  $\min_{i \in \mathcal{V}_{\text{SF},k}} \{\text{SF}^{-1}(p_i)\}$ . In the eighth step, a neighbor subset of a boundary POI is a subset to which the boundary POI can be assigned other than its current subset. For example, Fig. 4 (left) shows a partition with the POI marked by the empty circle denoting a boundary POI of  $\mathcal{V}_{\text{SF},4}$ . Clearly,  $\mathcal{V}_{\text{SF},3}$  is its neighbor subset since this POI can also be assigned to  $\mathcal{V}_{\text{SF},3}$ . If the total energy cost is smaller after assigning this POI to  $\mathcal{V}_{\text{SF},3}$ , then the algorithm updates  $\mathcal{V}_{\text{SF},3}$  and  $\mathcal{V}_{\text{SF},4}$ , as shown in Fig. 4 (right). By checking all the boundary POIs one by one in each iteration,  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)$  is reduced until it converges. It is worth mentioning that this algorithm is not intended to optimally partition the POIs in  $\mathcal{C}$ , since this will still be a computationally-intensive combinatorial optimization problem. Thus, the result of the algorithm depends on the choice of the initial partition and the choice of the boundary POI in each iteration. As expected, there is no guarantee that it will converge to the optimal partition.

#### Algorithm 1 Our proposed approach based on space-filling curves

- 1: map  $p_i$  to  $\mathcal{C}$ , for  $i \in \{1, \dots, m\}$
- 2: initially partition the POIs into a  $\{\mathcal{V}_{\text{SF},k}\}_{k=1}^n$  that satisfies our proposed grouping condition
- 3: order the mapped POIs in each  $\mathcal{V}_{\text{SF},k}$  to obtain  $\mathcal{H}_{\text{SF},k}$
- 4: for each  $k$ , 1) if  $\phi_k < 1$ , let  $\tilde{E}_k^*(\mathcal{H}_{\text{SF},k}) = E_k^*(\mathcal{H}_{\text{SF},k}) + \mathbb{I}(E_k^*(\mathcal{H}_{\text{SF},k}) > E_{\text{max},k})(\Omega_1 + \Omega_2(E_k^*(\mathcal{H}_{\text{SF},k}) - E_{\text{max},k})) + \mathbb{I}(T_k^*(\mathcal{H}_{\text{SF},k}) > T_{\text{max}})(\Omega_1 + \Omega_3(T_k^*(\mathcal{H}_{\text{SF},k}) - T_{\text{max}}))$ , where  $T_k^*(\mathcal{H}_{\text{SF},k})$  and  $E_k^*(\mathcal{H}_{\text{SF},k})$  are calculated by substituting  $\mathcal{H}_{\text{SF},k}$  into (3) and (4) respectively; 2) if  $\phi_k \geq 1$ , let  $\tilde{E}_k^*(\mathcal{H}_{\text{SF},k}) = \Omega_1 + \Omega_4(\phi_k - 1)$
- 5: calculate  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n) = \sum_{k=1}^n \tilde{E}_k^*(\mathcal{H}_{\text{SF},k})$
- 6: while  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)$  is not converged
- 7: find the POIs on the boundary of  $\mathcal{V}_{\text{SF},k}$
- 8: get a new partition  $\{\mathcal{V}_{\text{SF},k}\}_{k=1}^n$  by assigning one of the boundary POI to its neighbor subset
- 9: find new  $\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n$  and  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)$  similar to steps 3-5
- 10: if the new partition results in a smaller energy
- 11: update  $\{\mathcal{V}_{\text{SF},k}\}_{k=1}^n$ ,  $\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n$  and  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)$
- 12: end if
- 13: end while
- 14: if the final solution satisfies constraints 1, 2 and 10 in (2)
- 15: return the solution
- 16: else
- 17: cannot find a feasible solution
- 18: end if

*Remark 1:* Note that we refer to a solution that satisfies constraints 1, 2 and 10 in (2) as a *feasible* solution in Algorithm 1. Our goal is then to find a feasible solution that minimizes the total cost. In general, Algorithm 1 may not be able to find a feasible solution though even if it exists.

Note that due to the large penalty of violating the constraints, if the algorithm finds a feasible solution in some iteration, then it will never iterate back to an infeasible solution. Hence, we are guaranteed to have a feasible solution if the initial partition is a feasible one. Such an initial partition can be easily found, for instance, if the POIs and robots are homogeneous, i.e.  $\rho_i = \rho$ , for all  $i$ , and  $\alpha_{i,k} = \alpha$ , for all  $i, k$ , and  $E_{\max,k}$  and  $T_{\max}$  are large, as we shall discuss later.

### C. Performance Analysis of the Proposed Approach Based on Space-Filling Curves

In this section, we show that, under certain conditions, the performance of our approach is at most a constant factor away from the global optimum. We first show the following two lemmas that will be used to prove our main result.

*Lemma 1:* Consider the following optimization problem:

$$\begin{aligned} \max \quad & J = \sum_{i=1}^k \mu(x_{2i-1}, x_{2i}) \\ \text{s.t.} \quad & 0 \leq x_1 \leq x_2 \leq \dots \leq x_{2k} \leq 1. \end{aligned}$$

The optimal solution is  $x_{2i}^* = x_{2i+1}^* = i/k$ , for  $i \in \{1, \dots, k-1\}$ ,  $x_1^* = 0$  and  $x_{2k}^* = 1$ . Moreover, we have  $J^* = \sqrt{k}$ .

*Proof:* Assume that  $x_{2i-1}^*$ ,  $x_{2i}^*$ ,  $x_{2i+1}^*$  and  $x_{2i+2}^*$  are the optimal points that maximize  $J$  for some  $i \in \{1, \dots, k-1\}$ . Clearly,  $x_{2i}$  and  $x_{2i+1}$  only impact the terms  $\mu(x_{2i-1}, x_{2i}) + \mu(x_{2i+1}, x_{2i+2})$ . Hence,  $x_{2i}^*$  and  $x_{2i+1}^*$  must maximize  $\mu(x_{2i-1}, x_{2i}) + \mu(x_{2i+1}, x_{2i+2})$  for fixed  $x_{2i-1}^*$  and  $x_{2i+2}^*$ . As a result, we have  $x_{2i}^* = x_{2i+1}^* = (x_{2i-1}^* + x_{2i+2}^*)/2$ . This implies that  $|x_{2i-1}^* - x_{2i}^*| = |x_{2i+1}^* - x_{2i+2}^*|$ , for  $i \in \{1, \dots, k-1\}$ . Therefore, we have  $|x_{2i-1}^* - x_{2i}^*| = 1/k$ , for  $i \in \{1, \dots, k\}$ , i.e.  $x_{2i}^* = x_{2i+1}^* = i/k$ , for  $i \in \{1, \dots, k-1\}$ ,  $x_1^* = 0$  and  $x_{2k}^* = 1$ . Moreover, we have  $J^* = \sum_{i=1}^k \sqrt{1/k} = \sqrt{k}$ . ■

*Lemma 2:* We have the following inequalities for  $\sum_{k=1}^n d(\mathcal{H}_k^*)$  and  $\sum_{k=1}^n d(\mathcal{H}_{\text{SF},k})$ , where  $\mathcal{H}_k^*$  denotes the minimum-length Hamiltonian cycle for the  $k$ th robot:

1)  $\sum_{k=1}^n d(\mathcal{H}_k^*) \geq d(\mathcal{H}^*) - W(n)$ , where  $\mathcal{H}^*$  is the minimum-length Hamiltonian cycle to visit all the POIs and  $W(n)$  is the largest possible minimum-length Hamiltonian cycle to visit  $n$  points in the square;

2)  $\sum_{k=1}^n d(\mathcal{H}_{\text{SF},k}) \leq d(\mathcal{H}_{\text{SF}}) + C_{\text{SF}}S\sqrt{n}$ , where  $\mathcal{H}_{\text{SF}}$  is the Hamiltonian cycle to visit all the POIs in the 2D workspace, after ordering the POIs in  $\mathcal{C}$  and mapping them back to the 2D workspace (see Section II-A).<sup>5</sup>

*Proof:* To prove the first part of the lemma, consider a partition  $\{\mathcal{V}_k\}_{k=1}^n$  of  $\mathcal{V}$ . First, choose an arbitrary POI in each  $\mathcal{V}_k$ , for  $k \in \{1, \dots, n\}$ . Then, find a minimum-length tour  $\mathcal{L}^*$  that visits these POIs. Clearly, we have  $d(\mathcal{L}^*) \leq W(n)$ . Moreover, it is straightforward to see that  $\sum_{k=1}^n d(\mathcal{H}_k^*) + d(\mathcal{L}^*) \geq d(\mathcal{H}^*)$ . Hence, we have  $\sum_{k=1}^n d(\mathcal{H}_k^*) \geq d(\mathcal{H}^*) - W(n)$ .

To prove the second part of the lemma, consider a partition  $\{\mathcal{V}_{\text{SF},k}\}_{k=1}^n$  of  $\mathcal{V}$  which satisfies the grouping constraint mentioned in Section III-B. Let  $x_{\min,k} = \min_{i \in \mathcal{V}_{\text{SF},k}} |\text{SF}^{-1}(p_i)|$

<sup>5</sup>Note that  $d(\mathcal{H}_{\text{SF},k})$  and  $d(\mathcal{H}_{\text{SF}})$  are the total distances of  $\mathcal{H}_{\text{SF},k}$  and  $\mathcal{H}_{\text{SF}}$  in the 2D workspace respectively.

and  $x_{\max,k} = \max_{i \in \mathcal{V}_{\text{SF},k}} |\text{SF}^{-1}(p_i)|$ . Without loss of generality, we define the mapping  $\text{SF}(\cdot)$  and label  $\mathcal{V}_{\text{SF},k}$ s in a way such that  $0 \leq x_{\min,1} \leq x_{\max,1} \leq x_{\min,2} \leq x_{\max,2} \leq \dots \leq x_{\max,n} \leq 1$ . By translating the corresponding partition and ordering back to the 2D workspace, it is straightforward to see that  $\sum_{k=1}^n d(\mathcal{H}_{\text{SF},k}) \leq d(\mathcal{H}_{\text{SF}}) + C_{\text{SF}}S \sum_{k=1}^n \mu(x_{\min,k}, x_{\max,k})$ . By applying Lemma 1, we then have  $\sum_{k=1}^n d(\mathcal{H}_{\text{SF},k}) \leq d(\mathcal{H}_{\text{SF}}) + C_{\text{SF}}S\sqrt{n}$ . ■

The following two lemmas from the TSP literature asymptotically characterize  $d(\mathcal{H}^*)$ ,  $d(\mathcal{H}_{\text{SF}})$  and  $W(n)$ .

*Lemma 3* ([23], [24]): Assume that the POIs are i.i.d. according to an absolutely continuous pdf  $f(q)$  defined over  $\mathcal{W}$ . Define  $I_{\mathcal{W}}(f) = \int_{\mathcal{W}} \sqrt{f(q)} dq$ . Also, let  $\mathbb{E}\{\cdot\}$  represent the mean value of the argument. Then,

1) there exists a constant  $\theta_{\text{TSP}}$  such that  $\lim_{m \rightarrow \infty} \mathbb{E}\{d(\mathcal{H}^*)\}/\sqrt{m} = \theta_{\text{TSP}}I_{\mathcal{W}}(f)$ ;

2) there exists constants  $\theta_{\text{SF},\text{TSP}}^-$  and  $\theta_{\text{SF},\text{TSP}}^+$  such that  $\liminf_{m \rightarrow \infty} \mathbb{E}\{d(\mathcal{H}_{\text{SF}})\}/\sqrt{m} \geq \theta_{\text{SF},\text{TSP}}^- I_{\mathcal{W}}(f)$  and  $\limsup_{m \rightarrow \infty} \mathbb{E}\{d(\mathcal{H}_{\text{SF}})\}/\sqrt{m} \leq \theta_{\text{SF},\text{TSP}}^+ I_{\mathcal{W}}(f)$ , where  $\theta_{\text{SF},\text{TSP}}^+ \neq \theta_{\text{SF},\text{TSP}}^-$  and  $\theta_{\text{SF},\text{TSP}}^+ - \theta_{\text{SF},\text{TSP}}^- < 2 \times 10^{-4}$ .

*Lemma 4* ([25]): There exists a constant  $\theta_{\text{TSP},\text{worst}}$  such that  $\lim_{n \rightarrow \infty} W(n)/\sqrt{n} \leq \theta_{\text{TSP},\text{worst}}S$ , where  $2/(12)^{1/4} \leq \theta_{\text{TSP},\text{worst}} \leq \sqrt{2}$ .

For the case that our approach can find a feasible solution, we show that the performance of our approach is at most a constant factor away from the global optimum, which we characterize mathematically. One example of such a case, where our approach will find a feasible solution, is when the POIs and robots are homogeneous, and  $E_{\max,k}$  and  $T_{\max}$  are large.

*Theorem 2:* Let  $E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)$  and  $E_{\text{tot}}^*(\{\mathcal{H}_k^*\}_{k=1}^n)$  denote the total energy cost of a feasible solution by applying our proposed algorithm and the global optimum that results from directly solving (2) respectively. Then, we have the following inequality:

$$\frac{E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)}{E_{\text{tot}}^*(\{\mathcal{H}_k^*\}_{k=1}^n)} \leq \frac{\beta_{\text{SF}}(d(\mathcal{H}_{\text{SF}}) + C_{\text{SF}}S\sqrt{n}) + E_{\text{comm}}}{\beta^*(d(\mathcal{H}^*) - W(n)) + E_{\text{comm}}}, \quad (5)$$

where  $\beta_{\text{SF}} = P_s \phi_{\text{SF},\text{st}} / ((1 - \phi_{\text{SF},\text{st}})v_{\max}) + \kappa_{\text{M}}$ ,  $\phi_{\text{SF},\text{st}} = \min_j \max_{k \in \{1, \dots, n\}} \phi_{\text{SF},\text{st},k}^j$ ,  $\phi_{\text{SF},\text{st},k}^j = \sum_{i \in \mathcal{V}_{\text{SF},\text{st},k}^j} \rho_i / \alpha_{i,k}$ ,  $\{\mathcal{V}_{\text{SF},\text{st},k}^j\}_{k=1}^n$  is the  $j$ th partition that satisfies constraints 1, 2 and 10 in (2) during the iterative process of the algorithm,  $\beta^* = P_s \phi_{\min}^* / ((1 - \phi_{\min}^*)v_{\max}) + \kappa_{\text{M}}$ ,  $\phi_{\min}^* = \min_{k \in \{1, \dots, n\}} \phi_k^*$ ,  $\phi_k^* = \sum_{i \in \mathcal{V}_k^*} \rho_i / \alpha_{i,k}$ ,  $\{\mathcal{V}_k^*\}_{k=1}^n$  is the optimal partition from directly solving (2) and  $E_{\text{comm}} = t_c \sum_{i=1}^m P_{\text{TX},\text{min},i}$ .

Moreover, if the POIs are i.i.d. according to an absolutely continuous pdf  $f(q)$  defined over  $\mathcal{W}$ , and  $m$  is sufficiently large, we then have the following inequality:

$$\begin{aligned} & \frac{\mathbb{E}\{E_{\text{tot}}(\{\mathcal{H}_{\text{SF},k}\}_{k=1}^n)\}}{\mathbb{E}\{E_{\text{tot}}^*(\{\mathcal{H}_k^*\}_{k=1}^n)\}} \\ & \leq \frac{\beta_{\text{SF}}(\theta_{\text{SF},\text{TSP}}^+ I_{\mathcal{W}}(f) + C_{\text{SF}}S\sqrt{n/m}) + \sqrt{m}K_{\mathcal{W}}(f)}{\beta^*(\theta_{\text{TSP}}I_{\mathcal{W}}(f) - \theta_{\text{TSP},\text{worst}}S\sqrt{n/m}) + \sqrt{m}K_{\mathcal{W}}(f)} \end{aligned}$$

$$\leq \underbrace{\frac{\beta_{\text{SF}}(\theta_{\text{SF,TSP}}^+ I_{\mathcal{W}}(f) + C_{\text{SF}} S \sqrt{n/m}) + \sqrt{m} K_{\mathcal{W}}(f)}{\kappa_{\text{M}}(\theta_{\text{TSP}} I_{\mathcal{W}}(f) - \sqrt{2} S \sqrt{n/m}) + \sqrt{m} K_{\mathcal{W}}(f)}}_{\eta}, \quad (6)$$

where  $K_{\mathcal{W}}(f) = t_c \int_{\mathcal{W}} f(q) 10^{-\sigma_{\text{ab}}(q) Q^{-1}(\epsilon)/10} \text{SNR}_{\text{th}} / \bar{\Upsilon}(q) dq$  is the average communication cost at each POI.

*Proof:* The sum of the optimal total motion and sensing energy costs has the following lower bound, based on Theorem 1:  $\sum_{k=1}^n E_{\text{mo},k}^*(\mathcal{H}_k^*) + E_{\text{sen},k}^*(\mathcal{H}_k^*) = \sum_{k=1}^n (P_s \phi_k^* / ((1 - \phi_k^*) v_{\text{max}}) + \kappa_{\text{M}}) d(\mathcal{H}_k^*) \geq (P_s \phi_{\text{min}}^* / ((1 - \phi_{\text{min}}^*) v_{\text{max}}) + \kappa_{\text{M}}) \sum_{k=1}^n d(\mathcal{H}_k^*) \geq \beta^* (d(\mathcal{H}^*) - W(n))$ , where the last inequality follows by using the first part of Lemma 2. Similarly, we have the following upper bound for the sum of the total motion and sensing costs by using our algorithm:  $\sum_{k=1}^n E_{\text{mo},k}^*(\mathcal{H}_{\text{SF},k}) + E_{\text{sen},k}^*(\mathcal{H}_{\text{SF},k}) = \sum_{k=1}^n (P_s \phi_{\text{SF},k} / ((1 - \phi_{\text{SF},k}) v_{\text{max}}) + \kappa_{\text{M}}) d(\mathcal{H}_{\text{SF},k}) \leq (P_s \phi_{\text{SF,st}} / ((1 - \phi_{\text{SF,st}}) v_{\text{max}}) + \kappa_{\text{M}}) \sum_{k=1}^n d(\mathcal{H}_{\text{SF,st}}^{j^*}) \leq \beta_{\text{SF}} (d(\mathcal{H}_{\text{SF}}) + C_{\text{SF}} S \sqrt{n})$  where the last inequality follows by using the second part of Lemma 2, and  $\mathcal{H}_{\text{SF,st}}^{j^*}$  is the Hamiltonian cycle of the  $k$ th robot on  $\mathcal{V}_{\text{SF,st},k}^{j^*}$  for  $j^* = \arg \min_j \max_{k \in \{1, \dots, n\}} \phi_{\text{SF,st},k}^j$ . As a result, we have (5) in the theorem.

The second part of the theorem follows straightforwardly by applying Lemmas 3 and 4. Note that the second inequality of (6) holds since  $\beta^* > \kappa_{\text{M}}$ . ■

There are some interesting properties of (5) and (6). First, it can be observed that the tightness of the upper bound depends on the *locality* property of the space-filling curves, i.e. constant  $C_{\text{SF}}$  in (1). Also, it depends on how close the heuristic TSP tour  $\mathcal{H}_{\text{SF}}$  is to the global optimum  $\mathcal{H}^*$ .

If the POIs are i.i.d. and the number of the POIs is large, it has been estimated that  $\theta_{\text{TSP}} \approx 0.712$  [26] and  $\theta_{\text{SF,TSP}}^+ \approx \theta_{\text{SF,TSP}}^- \approx 0.956$  [14]. Moreover, the stability constraint affects both parameters  $\beta_{\text{SF}}$  and  $\beta^*$ . In general,  $\beta_{\text{SF}}$  and  $\beta^*$  are unknown. However, if  $\rho_i = \rho$ , for all  $i$ ,  $\alpha_{i,k} = \alpha$ , for all  $i, k$ , and  $T_{\text{max}}$  and  $E_{\text{max},k}$  are large, it is straightforward to see that  $\phi_{\text{SF,st}}$  can be chosen as  $(1 + \lfloor m/n \rfloor) \rho / \alpha$  by initially partitioning the POIs equally among the robots (the second step of the algorithm), where  $\lfloor \cdot \rfloor$  denotes the largest integer that is smaller than  $m/n$ . This will be a feasible solution for the original problem given that such a solution exists. Finally, it can be seen that, as the number of POIs becomes large, the motion and communication costs linearly increase with respect to  $\sqrt{m}$  and  $m$  respectively. Hence, as  $m$  becomes large, the communication cost becomes dominating. As a result, as  $m$  increases, the gap between our approach and the global optimum decreases, and it goes to 0 as  $m$  goes to infinity.

*Remark 2:* Note that, as the number of POIs increases, we may also need to increase the number of robots to guarantee that it is feasible to stabilize  $\Psi_i(t)$  at each POI.

#### IV. SIMULATION RESULTS

Consider the case where the workspace is a  $50 \text{ m} \times 50 \text{ m}$  square region with the coordinates shown in Fig. 5 (left

and right). The channel in the workspace is generated with the following real channel parameters from downtown San Francisco [27]: path loss exponent is 4.57, standard deviation of shadowing is 4, decorrelation distance of shadowing is 3.52 m, and the multipath fading is Rician fading with Rician parameter of 5.9. The channel is extended to populate the space of interest using our probabilistic channel simulator [28]. We assume that the robots have 5% a priori channel samples gathered in the same environment. Then, the robots use the probabilistic channel assessment framework of [19], [20] to predict the channel quality at each POI in the workspace. We choose SNR threshold ( $\text{SNR}_{\text{th}}$ ) as 30 dB, receiver noise power as  $-104 \text{ dBm}$ , and  $t_c = 1 \text{ s}$ . We also use the real motion parameters of the Pioneer 3DX robot as follows:  $\kappa_1 = 7.4$ ,  $\kappa_2 = 0.29$  and  $v_{\text{max}} = 1 \text{ m/s}$  [21]. Moreover, we choose  $\rho_i = 1$ ,  $\alpha_{i,k} = 50$  for all  $i, k$ , and  $P_s = 0.01$ . Then, our approach can find a feasible initial solution, as discussed earlier. The MILP of (2) is solved using IBM ILOG CPLEX Studio in MATLAB [13]. Furthermore, we use Sierpiński curves in our proposed heuristic approach.

Fig. 5 (left and right) show examples of the solutions obtained by solving MILP and by using our approach respectively. In this simulation, there are 3 mobile robots and 24 POIs. As expected, the partitions and the Hamiltonian cycles are different. In this example, our proposed approach only takes 0.7% of the time it takes to solve the MILP, while consuming 23.5% more energy.

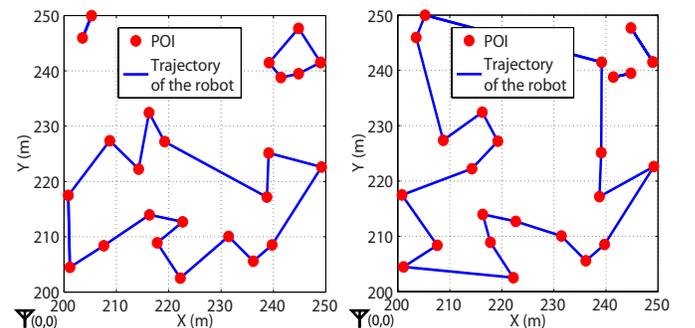


Fig. 5. Left and right figures show the solutions obtained by solving the MILP and by using our heuristic approach respectively. The position of the remote station is marked in both figures at  $(0, 0)$ . It can be seen that the partitions and the Hamiltonian cycles are different. In this example, our proposed approach only takes 0.7% of the time it takes to solve the MILP, while consuming 23.5% more energy.

Fig. 6 (left) shows the average run time of using our approach as compared to solving the MILP. In this simulation, there are 3 mobile robots. The result is averaged over 50 runs, where the positions of the POIs are drawn from a uniform distribution over the workspace. The desktop used to run the simulation has a 3GHz Duo Core CPU and 2GB memory. It can be seen that as the dimension of the problem increases, i.e. the number of POIs is large, using our heuristic approach is significantly faster than solving the MILP. Fig. 6 (right) compares the average performance of the two approaches. As can be seen, the total energy cost of our approach is only about 26% more than the optimum.

Finally, Fig. 7 shows the upper bound of the ratio of the total energy cost of our proposed algorithm to the global

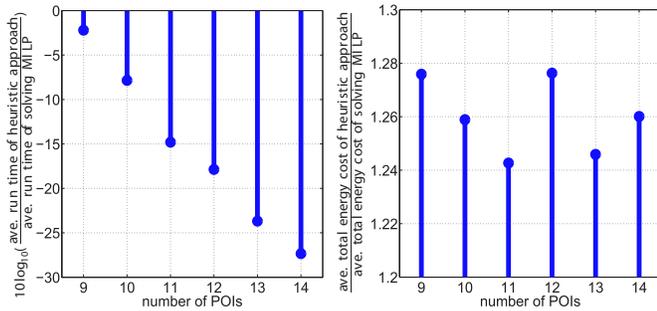


Fig. 6. Comparisons of the average run times (left) and average energy costs (right) of the MILP approach and our approach. It can be seen that our approach takes significantly less run time especially when the number of POIs is large. Also, the average total energy cost of our approach is only about 26% more than the optimum.

optimum ( $\eta$  of (6)) in this environment. In this example, we choose  $f(q)$  as a uniform pdf. It can be seen that this bound is tight. Also, the gap between our approach and the global optimum becomes smaller as the number of POIs increases.

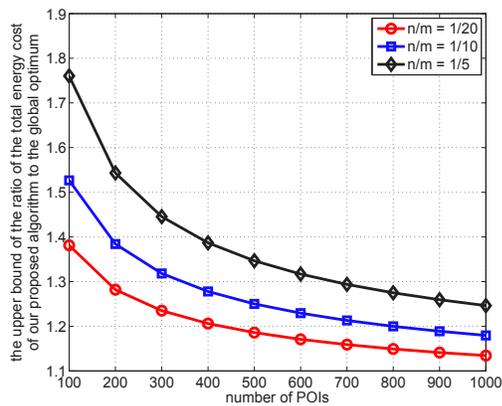


Fig. 7. The figure shows the upper bound of the ratio of the total energy cost of our proposed algorithm to the global optimum ( $\eta$  of (6)). It can be seen that the bound is tight. Also, the gap between our approach and the global optimum becomes smaller as the number of POIs increases.

## V. CONCLUSIONS

In this paper, we considered a dynamic coverage problem where a team of mobile robots needs to cover a set of POIs in a time-varying environment, while maintaining their connectivity to a remote station in a realistic communication environment. Our goal was to minimize the total energy cost of the robots, including motion, sensing and communication energy costs, while guaranteeing that 1) each POI is assigned to one of the robots, and 2) a periodic trajectory is designed for each robot to keep the quantity of interest at every POI bounded. We proposed a computationally-efficient approach to solve this problem by utilizing space-filling curves. Under certain conditions, we further mathematically proved that the performance of our heuristic approach is at most a constant factor away from the global optimum. Finally, our simulation results confirmed that our approach is considerably faster than solving the original MILP, especially when the dimension of the problem is high. Moreover, they showed that the energy consumption of our approach is only around 26% more than the optimum.

## REFERENCES

- [1] A. Ghaffarkhah and Y. Mostofi. Dynamic coverage of time-varying fading environments. *ACM Trans. on Sensor Networks*, 2013.
- [2] F. Bullo, E. Frazzoli, M. Pavone, K. Savla, and S. L. Smith. Dynamic vehicle routing for robotic systems. *Proc. of the IEEE*, 2011.
- [3] M. M. Zavlanos, M. B. Egerstedt, Y. C. Hu, and G. J. Pappas. Graph-theoretic connectivity control of mobile robot networks. *Proceedings of the IEEE*, 99(9):1525 – 1540, July 2011.
- [4] Cory Dixon and Eric W. Frew. Optimizing cascaded chains of unmanned aircraft acting as communication relays. *IEEE Journal on Selected Areas in Communications*, 30(5):883–898, 2012.
- [5] J. Fink, A. Ribeiro, and V. Kumar. Motion planning for robust wireless networking. In *IEEE Int'l Conf. on Robotics and Automation*, 2012.
- [6] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems, January 2004.
- [7] J. Cortes, S. Martinez, and F. Bullo. Coordination and geometric optimization via distributed dynamical systems. *SIAM Journal on Control and Optimization*, 44(5):1543–1574, 2005.
- [8] D. Bhaduria, O. Tekdas, and V. Isler. Robotic data mules for collecting data over sparse sensor fields. *Journal of Field Robotics*, 28(3):388–404, 2011.
- [9] S.S. Ponda, L.B. Johnson, A.N. Kopeikin, H.-L. Choi, and J.P. How. Distributed planning strategies to ensure network connectivity for dynamic heterogeneous teams. *IEEE Journal on Selected Areas in Communications*, 30(5):861–869, 2012.
- [10] Y. Wang and I. Hussein. Awareness coverage control over large-scale domains with intermittent communications. *IEEE Transactions on Automatic Control*, 55(8):1850–1859, 2010.
- [11] S. L. Smith, M. Schwager, and D. Rus. Persistent robotic tasks: Monitoring and sweeping in changing environments. *IEEE Transaction on Robotics*, 28(2):410–426, April 2012.
- [12] M. Bader. *Space-Filling Curves: An Introduction with Applications in Scientific Computing*, volume 9. Springer, 2012.
- [13] Ibm ilog cplex optimizer. <http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/>.
- [14] J. Bartholdi III and L. Platzman. An  $O(N \log N)$  planar travelling salesman heuristic based on spacefilling curves. *Operations Research Letters*, 1(4):121–125, 1982.
- [15] J. Bartholdi III, L. Platzman, R. Collins, and W. Warden. A minimal technology routing system for meals on wheels. *Interfaces*, 13(3):1–8, 1983.
- [16] J. Bartholdi III and L. Platzman. Heuristics based on spacefilling curves for combinatorial problems in euclidean space. *Management Science*, 34(3):291–305, 1988.
- [17] M. Goetschalckx and C. Jacobs-Blecha. The vehicle routing problem with backhauls. *European J. of Operational Research*, 42(1):39–51, 1989.
- [18] R. Bowerman, P. Calamai, and G. Brent Hall. The spacefilling curve with optimal partitioning heuristic for the vehicle routing problem. *European Journal of Operational Research*, 76(1):128–142, 1994.
- [19] Y. Mostofi, M. Malmirchegini, and A. Ghaffarkhah. Estimation of communication signal strength in robotic networks. In *Proc. of the 50th IEEE Int'l Conf. on Robotics and Automation*, May 2010.
- [20] M. Malmirchegini and Y. Mostofi. On the spatial predictability of communication channels. *IEEE Trans. on Wireless Comm.*, 2012.
- [21] Y. Mei, Y.-H. Lu, Y. C. Hu, and C. S. G. Lee. Deployment of mobile robots with energy and timing constraints. *IEEE Transactions on Robotics*, 22(3):507–522, June 2006.
- [22] C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *J. of the ACM*, 1960.
- [23] J. Steele. Subadditive euclidean functionals and nonlinear growth in geometric probability. *The Annals of Probability*, 9(3):365–376, 1981.
- [24] L. Platzman, and J. Bartholdi III. Spacefilling curves and the planar travelling salesman problem. *J. ACM*, 36(4):719–737, Oct. 1989.
- [25] K.J. Supowit, E.M. Reingold, and D.A. Plaisted. The travelling salesman problem and minimum matching in the unit square. *SIAM Journal on Computing*, 12(1):144–156, 1983.
- [26] A.G. Percus and O.C. Martin. Finite size and dimensional dependence in the euclidean traveling salesman problem. *Physical Review Letters*, 76(8):1188, 1996.
- [27] W. M. Smith. *Urban Propagation Modeling for Wireless Systems*. PhD thesis, 2004.
- [28] A. Gonzalez-Ruiz, A. Ghaffarkhah, and Y. Mostofi. A comprehensive overview and characterization of wireless channels for networked robotic and control systems. *Journal of Robotics*, 2011.