Distributed State Estimation using Intermittently Connected Robot Networks

Reza Khodayi-mehr, Yiannis Kantaros, and Michael M. Zavlanos

***Abstract*—We consider the problem of distributed state estimation using multi-robot systems. The robots have limited communication capabilities and only communicate their measurements intermittently, when they are physically close. To decrease the travel distance needed only to communicate, we divide the robots into small teams that communicate at different locations. Then, we propose a new distributed scheme that combines (i) communication schedules that ensure that the network is intermittently connected, and (ii) sampling-based motion planning for the robots in every team to collect optimal measurements and decide on a meeting time and location. This is the first distributed state estimation framework that relaxes all network connectivity assumptions and controls intermittent communication events so that the estimation uncertainty is minimized. Our results show significant improvement in estimation accuracy compared to methods that maintain end-to-end connection for all time.**

I. INTRODUCTION

Distributed state estimation (DSE) using mobile robot networks has a number of important applications, including robot localization [2], [3], SLAM [4]–[7], coverage [8], target localization [9]–[11] and tracking [12]–[17] among others. In these applications, the robots are equipped with sensing devices and collect information to minimize the uncertainty of a desired state variable. Successful accomplishment of this task critically depends on the ability of the robots to communicate with each other. Existing literature on DSE often *assumes* that the communication network remains connected for all time or over time. To the contrary, we propose a distributed control framework that allows the robots to temporarily disconnect in order to optimally collect information but also *ensures* that this information can be shared intermittently with all other robots through appropriate communication events.

DSE has been thoroughly studied in the literature. In much of this literature, e.g., [3], [5], [6], [8]–[12], [14], [16], [18], [19], the posterior distribution of the state is assumed Gaussian so that Kalman filter equations can be employed to obtain it in closed-form. Under more general assumptions, Bayesian framework is used for state estimation [2], [4], [20], [21]. None of these papers on DSE consider communication constraints. Such constraints can be in the form of limited communication ranges and rates, or delays and loss of data. For instance, a localization method for networks that are not guaranteed to be fully connected is presented in [7]. Common in all these works is the assumption that the communication network used to share the information is end-to-end connected for all time [22] or intermittently connected [7]. To the contrary, here we lift all connectivity assumptions and control the communication network so that it is intermittently connected infinitely often.

An important aspect of the multi-robot DSE problem is the design of informative paths for the robots to collect measurements. In this work, we allow the robots to temporarily disconnect and accomplish their tasks free of communication constraints. Then, the path planning problem should provide both a sequence of informative measurement locations for DSE and locations of communication events to share information. When the process that is estimated is time varying, delays during communication events should also be minimized to limit the time that robots wait without collecting any measurements. To address this challenging planning problem, we utilize sampling-based algorithms. Such algorithms have been used in [23], [24] for informative path planning. These works build upon the RRG algorithm [25] to design motion plans that maximize an information metric subject to budget constraints. Nevertheless, this approach cannot be used when the budget is unknown *a priori*, as in our work. Also, [23], [24] address single-agent motion planning problems. Applying these algorithms to multi-agent problems requires exploration of the joint space of all robots, which makes them computationally intractable for large-scale networks. By dividing the group of robots into small teams and using sampling-based methods to jointly plan the motion only of the robots in each team, our method scales to much larger problems.

Control of the network connectivity is an integral part of the DSE problem and has been widely studied. Approaches to preserve network connectivity for all time typically rely on controlling the Fiedler value of the underlying graph either in a centralized [15], [26], [27] or distributed [28]–[33] fashion. However, due to the uncertainty in wireless channels, it is often impossible to ensure all-time connectivity in practice. Moreover, all-time connectivity constraints may prevent the robots from moving freely in their environment to fulfill their tasks, and instead favor motions that maintain a reliable communication network. Motivated by this fact, intermittent communication frameworks have recently been proposed [34]–[39]. The key idea is to divide the robots into smaller teams and require that communication events take place when the robots in each team meet at a common location in space. Compared to [34]–[38], here we do not require that communication events take place when the robots in each team meet at a common location. Instead, we require that communication events take place when the robots in each team form a connected subnetwork somewhere in the continuous space, which introduces challenging connectivity constraints in the proposed planning problem for every team.

To the best of our knowledge, this is the first framework for DSE with *intermittent communication control*. We show that the proposed sampling-based solution to the informative path

This paper is a summary of the journal article [1] published in IEEE Transactions on Robotics. This work is supported in part by the ONR under grant $\#N000141812374$. Reza Khodayi-mehr and Michael M. Zavlanos are with the Department of Mechanical Engineering and Materials Science, Duke University, Durham, NC 27708, USA, {reza.khodayi.mehr, michael.zavlanos}@duke.edu. Yiannis Kantaros is with the Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104, USA, kantaros@seas.upenn.edu.

IEEE Transactions on Robotics (T-RO) paper, presented at ICRA 2020. It should be cited as a T-RO paper.

planning problem is probabilistically complete and guarantees that the robots in every team form a connected subnetwork at the same time in order to communicate, minimizing in this way the time that the robots remain idle. The proposed framework scales with the size of the robot teams and not with the size of the network. As a result, our approach can be applied to large-scale networks. Finally, we characterize the delay in propagation of information across the network as a function of the structure of the robot teams.

II. Problem Definition

Let $\mathbf{x}(t) \in \mathbb{R}^n$ denote a state variable that evolves according to the following nonlinear dynamics

$$\mathbf{x}(t+1) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{w}(t)), \tag{1}$$

where $\mathbf{u}(t) \in \mathbb{R}^{d_u}$ and $\mathbf{w}(t) \in \mathbb{R}^{d_w}$ denote the control input and the process noise at discrete time t. We assume that the process noise $\mathbf{w}(t)$ is normally distributed as $\mathbf{w}(t) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}(t))$ with covariance $\mathbf{Q}(t)$.

Consider also $N \geq 1$ mobile robots tasked with collaboratively estimating the state $\mathbf{x}(t)$ of the dynamic process (1). Let $\Omega \subset \mathbb{R}^d$ be the domain where the robots live and let $O \subset \Omega$ denote the set of obstacles. We assume that the robots collect measurements of the state $\mathbf{x}(t)$ inside the obstacle-free space $\Omega_{\text{free}} = \Omega \backslash O$ according to the following sensing model

$$\mathbf{y}(t, \mathbf{q}) = \mathbf{h}(\mathbf{x}(t), \mathbf{q}, \mathbf{v}(t)), \tag{2}$$

where $\mathbf{y} \in \mathbb{R}^m$ is the measurement vector at discrete time t taken at location $\mathbf{q} \in \Omega_{\text{free}}$ by one robot sensor and $\mathbf{v}(t) \sim \mathcal{N}(\mathbf{0}, \mathbf{R}(t))$ is the measurement noise with covariance $\mathbf{R}(t)$.

Moreover, we assume that the robots have limited communication capabilities and, therefore, they can communicate their measurements only when they are sufficiently close to each other. Specifically, every robot is able to communicate with another robot if it lies within a communication range $R \ll \texttt{diam}(\Omega)$, where $\texttt{diam}(\Omega)$ is the diameter of the domain Ω. Without loss of generality, we assume that the communication range is the same for all robots. Since the communication range is much smaller than the size of the domain in which the robots operate, requiring that all robots are connected either for all time or intermittently can significantly interfere with the tasks that they need to accomplish, especially if they have to travel to possibly remote locations in the domain. Therefore, we do not require that the robots ever form a connected network at once and, instead, we divide the robots into $M \geq 1$ teams, denoted by $\mathcal{T}_i$, where $i \in \{1, 2, \dots, M\}$, and require that every robot belongs to exactly two teams.

Given the teams $\mathcal{T}_i$, we define the graph of teams $\mathcal{G}_\mathcal{T} = \{\mathcal{V}_\mathcal{T}, \mathcal{E}_\mathcal{T}\}$ whose set of nodes $\mathcal{V}_\mathcal{T}$ is indexed by the teams $\mathcal{T}_i$, i.e., $\mathcal{V}_\mathcal{T} = \{1, 2, \dots, M\}$, and set of edges $\mathcal{E}_\mathcal{T}$ consists of links between nodes i and j if $\mathcal{T}_i \cap \mathcal{T}_j \neq \emptyset$, i.e., if there exist a robot r_{ij} that travels between teams $\mathcal{T}_i$ and $\mathcal{T}_j$. We assume that the robots in every team $\mathcal{T}_i$ communicate when they construct a connected network. Hereafter, we denote by $\mathcal{G}_{\mathcal{T}_i} = \{\mathcal{V}_{\mathcal{T}_i}, \mathcal{E}_{\mathcal{T}_i}(t)\}$ the communication graph constructed by robots in team $\mathcal{T}_i$, where the set of nodes $\mathcal{V}_{\mathcal{T}_i}$ contains all robots in team $\mathcal{T}_i$ and the set of edges $\mathcal{E}_{\mathcal{T}_i}(t)$ collects communication links that emerge between robots in $\mathcal{T}_i$, whose pairwise distance is less than or equal to R.

To ensure that information is propagated among all robots in the network, we require that all robots in every team $\mathcal{T}_i$ form connected graphs $\mathcal{G}_{\mathcal{T}_i}$ infinitely often. For this, it is necessary to assume that the graph of teams $\mathcal{G}_\mathcal{T}$ is connected. Specifically, if $\mathcal{G}_\mathcal{T}$ is connected, then information can be propagated intermittently across teams through robots that are common to these teams and, in this way, information can reach all robots in the network.

We denote by r_{ij} a robot that belongs to teams $\mathcal{T}_i$ and $\mathcal{T}_j$ and assume that its motion is governed by the following nonlinear dynamics

$$\mathbf{p}_{ij}(t+1) = \mathbf{g}(\mathbf{p}_{ij}(t), \mathbf{u}_{ij}(t)), \tag{3}$$

where $\mathbf{p}_{ij}(t) \in \Omega_{\text{free}}$ stands for the position of robot r_{ij} and $\mathbf{u}_{ij}(t) \in \mathbb{R}^{d_r}$ stands for a control input. Without loss of generality, we assume that all robots have the same dynamics. Then, the problem that we address in this paper can be summarized as follows.

Problem 2.1: Given the dynamic process (1) and measurement model (2), and a network of N robots divided into M teams $\mathcal{T}_i$, such that $\mathcal{G}_\mathcal{T}$ is connected, determine paths $\mathbf{p}_{ij}(t)$ for all robots r_{ij}, so that (i) all teams $\mathcal{T}_i$ communicate infinitely often, and (ii) all robots collectively minimize estimation uncertainty of the state $\mathbf{x}(t)$ over time.

Throughout the paper we assume that the dynamics of the hidden state (1), the control vector $\mathbf{u}(t)$ in (1), the sensing model (2), and the process and measurement noise covariances $\mathbf{Q}(t)$ and $\mathbf{R}(t)$ are known.

III. Path Planning with Intermittent Communication

To solve Problem 2.1, we propose a distributed control framework that concurrently plans robot trajectories that minimize a desired uncertainty metric and schedules communication events during which the robots exchange their gathered information and update their beliefs. The schedules of communication events are constructed in Section III-A. The connected subnetworks associated with these communication events and the informative paths the robots follow until they communicate are determined by an online sampling-based planner discussed in Section III-B.

A. Distributed Intermittent Connectivity Control

In this section, we design infinite sequences of communication events (also called communication schedules) that ensure that robots in every team $\mathcal{T}_i$ communicate intermittently and infinitely often, for all $i \in \{1, \dots, M\}$.

Definition 3.1: The schedule of communication events of robot r_{ij}, denoted by $\texttt{sched}_{ij}$, is defined as an infinite repetition of the finite sequence

$$s_{ij} = X, \dots, X, i, X, \dots, X, j, X, \dots, X, \tag{4}$$

i.e., $\texttt{sched}_{ij} = s_{ij}, s_{ij}, \dots = s_{ij}^{\omega}$, where ω stands for the infinite repetition of the finite sequence s_{ij}.

The detailed construction of these schedules is omitted and can be found in [37, Sec. V]. In (4), i and j represent communication events for teams $\mathcal{T}_i$ and $\mathcal{T}_j$, respectively, and the discrete states X indicate that there is no communication event for robot r_{ij}. We denote by $\texttt{sched}_{ij}(k_{ij})$ the k_{ij}-th entry in the sequence $\texttt{sched}_{ij}$, where $k_{ij} \geq 1$. Hereafter, we call the indices k_{ij} *epochs*. By construction of the schedules $\texttt{sched}_{ij}$, it holds that the epochs when the communication events for a team $\mathcal{T}_i$ will occur are the same for all $r_{ij} \in \mathcal{T}_i$. Also, due to the infinite repetition of the sequence s_{ij}, communication among robots in any team $\mathcal{T}_i$ recurs every T epochs, where T is the length of s_{ij}. The schedules in Definition 3.1 ensure that all teams $\mathcal{T}_i$ communicate infinitely often [37].

Since the robots communicate intermittently, information is propagated across the network with a delay. In the following proposition, we characterize the worst-case delay. The proof is omitted due to space limitations and can be found in [1].

Proposition 3.2: The worst case delay, measured in terms of elapsed epochs k_{ij}, with which information collected by robot r_{ij} will propagate to every other robot in the network is $D_{\mathcal{G}_\mathcal{T}} = (T-1)L_{\mathcal{G}_\mathcal{T}}$, where $L_{\mathcal{G}_\mathcal{T}}$ is the longest shortest path in $\mathcal{G}_\mathcal{T}$ and T is the period of the schedules $\texttt{sched}_{ij}$.

B. Informative Path Planning

The schedules $\texttt{sched}_{ij}$ developed in Section III-A determine an abstract sequence of communication events that are not associated with any physical location in space or meeting time instant. In this section, we embed the schedules $\texttt{sched}_{ij}$ over the epochs k_{ij} into time t and space Ω.

Let k be an epoch when a communication event takes place for team $\mathcal{T}_i$, i.e., $\texttt{sched}_{ij}(k) = i$, where to simplify notation we drop dependence of the epoch k_{ij} on the robot r_{ij}. Moreover, assume that the path $\mathbf{p}_{ij}(t)$ is divided into segments indexed by the epochs and, let $\mathbf{p}_{ij}^k : [t_{0,ij}^k, t_{f,ij}^k] \rightarrow \Omega_{\text{free}}$ denote the k-th segment of path $\mathbf{p}_{ij}(t)$ of robot $r_{ij} \in \mathcal{T}_i$ starting at the discrete time $t_{0,ij}^k$ and ending at $t_{f,ij}^k$. Communication within team $\mathcal{T}_i$ during the k-th epoch takes place at the time instant $t_{f,\mathcal{T}_i}^k = \max_{r_{ij}\in\mathcal{T}_i}\left\{t_{f,ij}^k\right\}$ when all robots $r_{ij} \in \mathcal{T}_i$ arrive at the end locations $\mathbf{p}_{ij}(t_{f,ij}^k)$ of their paths $\mathbf{p}_{ij}^k$. The starting location $\mathbf{p}_{ij}(t_{0,ij}^k)$ of the path $\mathbf{p}_{ij}^k$ coincides with the location where the last communication event for robot r_{ij} took place within team $\mathcal{T}_j$; see also Figure 1. The epoch and corresponding time instant when communication for team $\mathcal{T}_j$ took place are denoted by $\bar{k}$ and $t_{f,\mathcal{T}_j}^{\bar{k}}$, respectively. Thus, robot r_{ij} starts executing the path segment $\mathbf{p}_{ij}^k$ at the time instant $t_{0,ij}^k = t_{f,\mathcal{T}_j}^{\bar{k}}$ and finishes its execution at $t_{f,ij}^k$. At time $t_{f,\mathcal{T}_i}^k$, when all robots in $\mathcal{T}_i$ communicate, they collectively design the path segments $\mathbf{p}_{ij}^{k+T}$, $\forall r_{ij} \in \mathcal{T}_i$, that will result in a connected configuration for team $\mathcal{T}_i$ at epoch $k+T$; see Figure 1.

In what follows, our goal is to design path segments $\mathbf{p}_{ij}^{k+T}$ that minimize estimation uncertainty and satisfy the following three constraints: (a) the paths $\mathbf{p}_{ij}^{k+T}$ do not intersect with the obstacles and respect the dynamics (3), (b) the end locations $\mathbf{p}_{ij}(t_{f,ij}^{k+T})$ for all robots $r_{ij} \in \mathcal{T}_i$ correspond to a connected communication graph $\mathcal{G}_{\mathcal{T}_i}$ for team $\mathcal{T}_i$, and (c) the end times $t_{f,ij}^{k+T}$ of the paths $\mathbf{p}_{ij}^{k+T}$ are the same and equal to $t_{f,\mathcal{T}_i}^{k+T}$ for all robots $r_{ij} \in \mathcal{T}_i$, so that there are no robots waiting for the arrival of other robots to communicate. Minimizing waiting times allows the robots to spend more time collecting measurements needed for estimation. To achieve

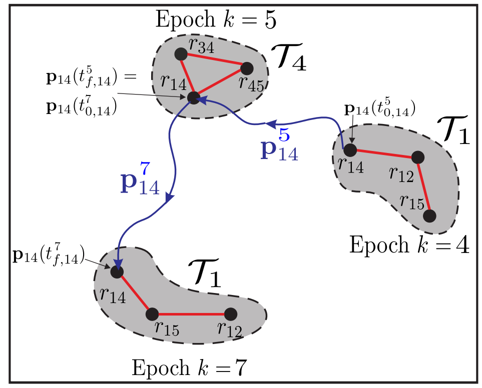

Fig. 1: Illustration of the proposed planning algorithm for robot r_{14} with $\texttt{sched}_{14} = [1,\ 4,\ X]^\omega$ and period $T=3$. The figure shows the path segments $\mathbf{p}_{14}^k$ that robot r_{14} follows, where the paths $\mathbf{p}_{14}^5$ and $\mathbf{p}_{14}^7$ were designed at epochs $k = 5 - T = 2$ and $k = 7 - T = 4$, respectively. Observe that $\mathbf{p}_{14}^6 = \emptyset$, since $\texttt{sched}_{14}(6) = X$.

this goal, we formulate an optimal control problem that the robots $r_{ij} \in \mathcal{T}_i$ need to solve in order to design paths $\mathbf{p}_{ij}^{k+T}$ when they communicate at epoch k. We define by $\mathcal{C} = \{\mathcal{G}(\mathcal{V},\mathcal{E}) \mid \lambda_2(\mathcal{L}(\mathcal{G})) > 0\}$, the set of all connected graphs $\mathcal{G}$ with vertices in the set $\mathcal{V}$ and edges in the set $\mathcal{E}$, i.e., the set of graphs $\mathcal{G}$ whose Laplacian matrix $\mathcal{L}(\mathcal{G})$ has a positive second smallest eigenvalue [40]. Then, the optimal control problem that we formulate to solve Problem 2.1 is given as

$$
\begin{aligned}
\min_{\mathbf{P}_{\mathcal{T}_i}, t_{f,\mathcal{T}_i}} & \sum_{t=t_{0,\mathcal{T}_i}}^{t_{f,\mathcal{T}_i}} \texttt{unc}(\mathbf{P}_{\mathcal{T}_i}(t)) \qquad (5)\\
\text{s.t. } & \mathbf{p}_{ij}(t_{0,ij}^{k+T}) = \mathbf{p}_{ij}(t_{f,\mathcal{T}_j}^{\bar{k}}),\ \forall r_{ij} \in \mathcal{T}_i \\
& \mathbf{p}_{ij}(t) \in \Omega_{\text{free}},\ \forall r_{ij} \in \mathcal{T}_i \\
& \mathbf{p}_{ij}(t+1) = \mathbf{g}(\mathbf{p}_{ij}(t), \mathbf{u}_{ij}(t)),\ \forall r_{ij} \in \mathcal{T}_i \\
& \mathcal{G}_{\mathcal{T}_i}^{k+T} \in \mathcal{C}, \\
& t_{f,ij} = t_{f,\mathcal{T}_i},\ \forall r_{ij} \in \mathcal{T}_i, \\
& \texttt{unc}(\mathbf{P}_{\mathcal{T}_i}(t_{f,\mathcal{T}_i})) \leq \delta.
\end{aligned}
$$

In (5), $\mathbf{P}_{\mathcal{T}_i}$ stands for the joint path of the robots $r_{ij} \in \mathcal{T}_i$ that lives in the joint space $\Omega_{\text{free}}^{|\mathcal{T}_i|}$. Projection of this path on the workspace of robot r_{ij} yields the path segment $\mathbf{p}_{ij}^{k+T}$. Also, $\texttt{unc}(\mathbf{P}_{\mathcal{T}_i}(t)))$ denotes an arbitrary positive uncertainty metric such as a scalar function of the covariance matrix [18]. Specifically, in Section IV, we employ the maximum eigenvalue of the posterior covariance as the uncertainty metric which can be computed using the extended Kalman filter (EKF). Then, the objective function measures the cumulative uncertainty in the estimation of $\mathbf{x}(t)$ after fusing information from all robots in $\mathcal{T}_i$ collected along their individual paths $\mathbf{p}_{ij}$ from $t_{0,\mathcal{T}_i} = \min\{t_{0,ij}^{k+T}\}$ up to time $t_{f,\mathcal{T}_i}$, given all earlier measurements available to team $\mathcal{T}_i$. The first constraint in (5) enforces that the paths start from the location where the previous communication event for the robots in $\mathcal{T}_i$ occurred. The second constraint in (5) requires that the paths lie in the free space. The third constraint requires that the dynamics (3) is satisfied. The fourth constraint requires that the communication graph $\mathcal{G}_{\mathcal{T}_i}^{k+T}$ constructed at epoch $k+T$, once all robots in $\mathcal{T}_i$ reach the end point of their respective paths $\mathbf{p}_{ij}^{k+T}$, is connected. The next constraint requires that all robots in

Algorithm 1 Sampling-based Informative Path Planning

1: Set $\mathcal{V}_s = \{\mathbf{v}_0\}$, $\mathcal{E}_s = \emptyset$, and $\mathcal{X}_g^i = \emptyset$;
2: **for** $s = 1, \dots, n_{\text{sample}}$ **do**
3: Sample Ω_{free} to acquire $\mathbf{v}_{\text{rand}}$;
4: Find the nearest node $\mathbf{v}_{\text{nearest}} \in \mathcal{V}_s$ to $\mathbf{v}_{\text{rand}}$;
5: Steer from $\mathbf{v}_{\text{nearest}}$ toward $\mathbf{v}_{\text{rand}}$ to select $\mathbf{v}_{\text{new}}$;
6: **if** `CollisionFree`$(\mathbf{v}_{\text{nearest}}, \mathbf{v}_{\text{new}})$ **then**
7: Update the set of vertices $\mathcal{V}_s = \mathcal{V}_s \cup \{\mathbf{v}_{\text{new}}\}$;
8: Build the set $\mathcal{V}_{\text{near}} = \{\mathbf{v} \in \mathcal{V}_s \mid \|\mathbf{v} - \mathbf{v}_{\text{new}}\| < r\}$;
9: Extend the tree towards $\mathbf{v}_{\text{new}}$;
10: Rewire the tree;
11: **end if**
12: **end for**
13: Find $\mathbf{v}_{\text{end}} \in \mathcal{X}_g^i$ with smallest uncertainty;
14: Return the path $\mathbf{P}_{\mathcal{T}_i}^{k+T} = (\mathbf{v}_0, \dots, \mathbf{v}_{\text{end}})$;
15: Project $\mathbf{P}_{\mathcal{T}_i}^{k+T}$ onto the workspace of r_{ij} to get $\mathbf{p}_{ij}^{k+T}$;

$\mathcal{T}_i$ terminate the execution of their path segments $\mathbf{p}_{ij}^{k+T}$ at time instants $t_{f,ij}$ that are the same and equal to $t_{f,\mathcal{T}_i}$, i.e., no waiting time. The final constraint in (5) requires that the uncertainty metric $\texttt{unc}(\mathbf{P}_{\mathcal{T}_i}(t_{f,\mathcal{T}_i}))$ at time $t_{f,\mathcal{T}_i}$ is below a specified threshold $\delta > 0$. This ensures that the robots explore the environment sufficiently before they communicate again.

C. Sampling-based Solution to Planning Problem

To solve (5) we propose the sampling-based Algorithm 1 that is built upon the RRT* algorithm [25]. Application of this algorithm requires definition of a cost function and a goal set $\mathcal{X}_g^i$. Referring to (5), the cost of the joint path $\mathbf{P}_{\mathcal{T}_i}$ that lives in $\Omega_{\text{free}}^{|\mathcal{T}_i|}$ is defined as $\texttt{Cost}(\mathbf{P}_{\mathcal{T}_i}) = \sum_{t=t_{0,\mathcal{T}_i}}^{t_{f,\mathcal{T}_i}} \texttt{unc}(\mathbf{P}_{\mathcal{T}_i}(t))$. The goal set captures the constraints in (5). We define this set for team $\mathcal{T}_i$ as follows:

$$\mathcal{X}_g^i = \{\mathbf{v} \in \Omega_{\text{free}}^{|\mathcal{T}_i|} \mid \text{(i) } \lambda_2(\mathcal{L}(\mathcal{G}_{\mathcal{T}_i}^{k+T}(\mathbf{v}))) > 0, \\ \text{(ii) } \min_{r_{ij} \in \mathcal{T}_i} t_{ij}(\mathbf{v}) = \max_{r_{ij} \in \mathcal{T}_i} t_{ij}(\mathbf{v}), \text{ (iii) } \texttt{unc}(\mathbf{v}) \leq \delta\}, \quad (6)$$

where $t_{ij}(\mathbf{v})$ denotes the time instant where robot r_{ij} arrives at its respective location in node $\mathbf{v}$ of the tree.

Algorithm 1 generates a tree denoted by $\mathcal{G}_s = \{\mathcal{V}_s, \mathcal{E}_s\}$ that resides in $\Omega_{\text{free}}^{|\mathcal{T}_i|}$, where $\mathcal{V}_s$ denotes its set of nodes and $\mathcal{E}_s$ denotes its set of edges. This tree is initialized as $\mathcal{V}_s = \{\mathbf{v}_0\}$, $\mathcal{E}_s = \emptyset$ [line 1, Alg. 1] where the root $\mathbf{v}_0 \in \Omega_{\text{free}}^{|\mathcal{T}_i|}$ of the tree is selected so that it matches the positions $\mathbf{p}_{ij}(t_{0,ij}^{k+T})$ of the robots $r_{ij} \in \mathcal{T}_i$ in the joint space $\Omega_{\text{free}}^{|\mathcal{T}_i|}$. This ensures that the first constraint in (5) is satisfied. In other words, the tree generated by Algorithm 1 is rooted at the end point of the previous paths $\mathbf{p}_{ij}^k$. The tree $\mathcal{G}_s = \{\mathcal{V}_s, \mathcal{E}_s\}$ is built incrementally by adding new samples $\mathbf{v} \in \Omega_{\text{free}}^{|\mathcal{T}_i|}$ to $\mathcal{V}_s$ and corresponding edges to $\mathcal{E}_s$, based on three operations: '`Sample`' [line 3, Alg. 1], '`Extend`' [line 9, Alg. 1], and '`Rewire`' [line 10, Alg. 1]. After taking n_{sample} samples, where $n_{\text{sample}} \geq 1$ is user-specified, Algorithm 1 terminates and returns the node $\mathbf{v}_{\text{end}} \in \mathcal{X}_g^i$ with the smallest cost. Then, the path $\mathbf{P}_{\mathcal{T}_i}^{k+T} = (\mathbf{v}_0, \dots, \mathbf{v}_{\text{end}})$ that connects $\mathbf{v}_{\text{end}}$ to the root $\mathbf{v}_0$ of the tree can be obtained [lines 13-14, Alg. 1]. The individual path segments $\mathbf{p}_{ij}^{k+T}$ of the robots are obtained by projecting the joint path $\mathbf{P}_{\mathcal{T}_i}^{k+T}$ onto the space of each robot. Note that the paths start at different initial times $t_{0,ij}^{k+T}$ but end at the same time $t_{f,\mathcal{T}_i}^{k+T}$ since $\mathbf{v}_{\text{end}}$ belongs to the goal set.

The sampling, steer, and extend operation in Algorithm 1 are executed as in the RRT* algorithm [25]. Specifically, at every iteration of Algorithm 1, a new sample $\mathbf{v}_{\text{rand}}$ is drawn from the joint space $\Omega_{\text{free}}^{|\mathcal{T}_i|}$ [line 3, Alg. 1]. Among all the nodes in the set $\mathcal{V}_s$, we find the nearest node to $\mathbf{v}_{\text{rand}}$, denoted by $\mathbf{v}_{\text{nearest}}$ [line 4, Alg. 1]. Then, we define a steering function that given the robot dynamics (3) and the nodes $\mathbf{v}_{\text{rand}}$ and $\mathbf{v}_{\text{nearest}}$ returns a point $\mathbf{v}_{\text{new}}$ [line 5, Alg. 1] that (i) minimizes the distance $\|\mathbf{v}_{\text{new}} - \mathbf{v}_{\text{rand}}\|$, (ii) satisfies $\|\mathbf{v}_{\text{nearest}} - \mathbf{v}_{\text{new}}\| \leq \epsilon$, for some $\epsilon > 0$, and (iii) minimizes the time difference $\max_{r_{ij} \in \mathcal{T}_i} t_{ij}(\mathbf{v}_{\text{new}}) - \min_{r_{ij} \in \mathcal{T}_i} t_{ij}(\mathbf{v}_{\text{new}})$. Given the new point $\mathbf{v}_{\text{new}}$, Algorithm 1 next examines if the tree $\mathcal{G}_s$ can be extended towards $\mathbf{v}_{\text{new}}$ [lines 6-9, Alg. 1] so that (i) $\mathbf{v}_{\text{new}}$ is reached from one of the nodes in $\mathcal{V}_{\text{near}} = \{\mathbf{v} \in \mathcal{V}_s \mid \|\mathbf{v} - \mathbf{v}_{\text{new}}\| < r\}$ in an obstacle-free way, where $r > 0$ is selected as in [25] and (ii) the cost $\texttt{Cost}(\mathbf{v}_{\text{new}})$ of reaching $\mathbf{v}_{\text{new}}$ from the root of the tree is minimized. Recall that $\texttt{Cost}(\mathbf{v}_{\text{new}})$ corresponds to the cost of the path that connects the node $\mathbf{v}_{\text{new}}$ to the root $\mathbf{v}_0$ and can be computed using the EKF; see [1] for more details.

The main difference with RRT* lies in the rewiring step. Particularly, after extending the tree $\mathcal{G}_s$ towards $\mathbf{v}_{\text{new}}$, the rewiring process follows that checks if it is possible to further reduce the cost of the nodes of the tree by rewiring them through $\mathbf{v}_{\text{new}}$. In particular, for every node $\mathbf{v}_{\text{near}} \in \mathcal{V}_{\text{near}}$ that (i) satisfies $\min_{r_{ij} \in \mathcal{T}_i} t_{ij}(\mathbf{v}_{\text{near}}) = \max_{r_{ij} \in \mathcal{T}_i} t_{ij}(\mathbf{v}_{\text{near}})$, and (ii) can be connected through an obstacle-free path to $\mathbf{v}_{\text{new}}$, we compute $\overline{\texttt{Cost}}(\mathbf{v}_{\text{near}})$ assuming that $\mathbf{v}_{\text{near}}$ was connected to $\mathbf{v}_{\text{new}}$. Then we rewire $\mathbf{v}_{\text{near}}$ if the cost $\overline{\texttt{Cost}}(\mathbf{v}_{\text{near}})$ using $\mathbf{v}_{\text{new}}$ as its parent is less than the current cost $\texttt{Cost}(\mathbf{v}_{\text{near}})$ and if there are control inputs $\mathbf{u}_{ij}$ that can still drive all robots to $\mathbf{v}_{\text{near}}$ at the same time.

Completeness and Optimality: Our proposed sampling-based algorithm is probabilistically complete, i.e., if there exist paths $\mathbf{p}_{ij}^{k+T}$ that terminate at the goal set (6), then Algorithm 1 will find them with probability 1, as $n_{\text{sample}} \to \infty$. To show this result, recall that RRT* is probabilistically complete given the functions '`Steer`' and '`Extend`' [25]. The only requirement in the '`Steer`' function is that the node $\mathbf{v}_{\text{new}}$ is closer to $\mathbf{v}_{\text{rand}}$ than $\mathbf{v}_{\text{nearest}}$ is, which is trivially satisfied by Algorithm 1. Finally, since the '`Extend`' function is the as same as the extend operation of RRT*, we conclude that Algorithm 1 is probabilistically complete. Nevertheless, Algorithm 1 is not asymptotically optimal, since rewiring a node that belongs to $\mathcal{V}_{\text{near}}$ takes place only if its cost after rewiring decreases and all robots can still arrive at this node simultaneously. On the other hand, in the rewiring step of RRT* the time constraint considered here is not present.

D. Integrated Intermittent Communication-aware Planning

The integrated algorithm is summarized as follows. All robots r_{ij} follow the path $\mathbf{p}_{ij}^k$ while collecting measurements. When robot r_{ij} reaches the final waypoint of $\mathbf{p}_{ij}^k$, all robots in team $\mathcal{T}_i$ form a connected communication network, simultaneously. When this happens, the robots exchange the information/measurements they have collected since their last communication event. Given the new set of measurements, the state estimate and the respective covariance are updated using the EKF. Finally, the robots in team $\mathcal{T}_i$ compute the paths $\mathbf{p}_{ij}^{k+T}$ that will allow them to reconnect at epoch $k+T$.

IV. Numerical Experiments

In this section, we demonstrate the performance of the proposed algorithm for a target tracking problem in a non-convex environment. Specifically, we consider $N=8$ robots and 8 targets that reside in a $10\times10\times5\text{m}^3$ indoor environment. The targets move in this 3D environment while avoiding the obstacles and are modeled by a linear time-invariant dynamics $\mathbf{x}_a(t+1)=\mathbf{A}_a\mathbf{x}_a(t)+\mathbf{B}_a\mathbf{u}_a(t)+\mathbf{w}_a(t)$, where a denotes the target index, $\mathbf{x}_a(t)\in\mathbb{R}^3$, and $\mathbf{w}_a(t)\sim(\mathbf{0},\mathbf{Q}_a(t))$. We also consider ground robots that live in Ω, which is the 2D projection of the space of targets. The robots are governed with the following first-order dynamics $\mathbf{p}_{ij}(t+1)=\mathbf{p}_{ij}(t)+\mathbf{u}_{ij}(t)$, where $\mathbf{p}_{ij}(t)\in\mathbb{R}^2$, $\mathbf{u}_{ij}\in\mathbb{R}^2$, and $\|\mathbf{u}_{ij}\|\leq 0.1$m. The communication range of the robots is $R=0.2\text{m}\ll\texttt{diam}(\Omega)$. The robots need to move in Ω_{free} to estimate the 3D positions $\mathbf{x}_a(t)$ of all targets.

We equip the robots with omnidirectional, range-only, line-of-sight sensors with limited range of 5m. Every robot can take noisy measurements of its distance from all targets that lie within its sight and range. Specifically, the measurement associated with robot r_{ij} and target a is given by

$$y_{ij,a}=\ell_{ij,a}(t)+v(t)\quad\text{if}\quad(\ell_{ij,a}(t)\leq 5)\wedge(\mathbf{x}_a(t)\in\texttt{FOV}_{ij}),\tag{7}$$

where $\ell_{ij,a}(t)=||\mathbf{p}_{ij}(t)-\mathbf{x}_a(t)||$, $v(t)\sim\mathcal{N}(0,\sigma^2(t))$, $\texttt{FOV}_{ij}$ is the field-of-view of robot r_{ij}, and

$$\sigma(t)=\begin{cases}0.01, & \text{if } \ell_{ij,a}(t)\leq 1,\\ 0.045\,\ell_{ij,a}(t)-0.035, & \text{if } 1<\ell_{ij,a}(t)\leq 3,\\ 0.1, & \text{if } 3<\ell_{ij,a}(t)\leq 5.\end{cases}\tag{8}$$

This model captures the fact that the range readings become less accurate as the distance increases and is designed to motivate the robots to approach the targets. Observe in (8) that the standard deviation $\sigma(t)$ depends on the position of the robot r_{ij} and the true position of target a. This means that the measurement noise covariance is unknown and, therefore, during the execution of Algorithm 1, we estimate $\sigma(t)$ using the predicted position of the targets. Despite that, in the following case studies we show that Algorithm 1 can synthesize paths that outperform competitive methods.

As mentioned in Section III-B, we select the cost function as $\texttt{unc}(\mathbf{P}_{\mathcal{T}_i}(t))=\lambda_n(\mathbf{C}_{\mathcal{T}_i}(t))$, where $\lambda_n(\cdot)$ denotes the maximum eigenvalue and $\mathbf{C}_{\mathcal{T}_i}$ is the joint covariance matrix of all targets computed using only local information available to team $\mathcal{T}_i$. In this way, we minimize the worst-case uncertainty of localizing the targets. The goal set is constructed as in (6) where for the problem at hand, the constraint (iii) in (6) is defined as $\lambda_n(\mathbf{C}_{\mathcal{T}_i}^{a_i}(t_{f,\mathcal{T}_i}))\leq\delta$, in which a_i is the most uncertain target from the view point of team $\mathcal{T}_i$ and $\mathbf{C}_{\mathcal{T}_i}^{a_i}(t)$ is the a_i-th diagonal block in the covariance matrix $\mathbf{C}_{\mathcal{T}_i}$. In other words, we require each team $\mathcal{T}_i$ to design paths so that the uncertainty of the most uncertain target, determined according to the local information of team $\mathcal{T}_i$, drops below a threshold δ. In what follows, we select $\delta=0.12^2\text{m}^2$, i.e., team $\mathcal{T}_i$ must localize the most uncertain target with uncertainty not worse than 0.12m.

Finally, the sampling function defined in Section III-C, is constructed as follows. First, we draw a pre-defined number of samples from a uniform distribution f_1 defined over $\Omega_{\text{free}}^{|\mathcal{T}_i|}$ that allows the team $\mathcal{T}_i$ to explore the domain. Then, we switch to a distribution f_2 that enables the robots to get closer and form a connected configuration. This distribution is constructed in two steps. Specifically, we first draw a possible meeting location $\mathbf{q}\in\Omega_{\text{free}}$ from a uniform distribution. Then, we construct the joint vector $\boldsymbol{\mu}=[\mathbf{q}^T,\dots,\mathbf{q}^T]^T\in\Omega_{\text{free}}^{|\mathcal{T}_i|}$ and draw the desired samples from $f_2(\mathbf{v})=\mathcal{N}(\boldsymbol{\mu},(2R)^2\mathbf{I})$.

We assume that the network of $N=8$ robots is divided into $M=8$ teams defined as $\mathcal{T}_1=\{r_{12},r_{18}\}$, $\mathcal{T}_2=\{r_{12},r_{23}\}$, $\mathcal{T}_3=\{r_{23},r_{34}\}$, $\mathcal{T}_4=\{r_{14},r_{45}\}$, $\mathcal{T}_5=\{r_{56},r_{45}\}$, $\mathcal{T}_6=\{r_{56},r_{67}\}$, $\mathcal{T}_7=\{r_{67},r_{78}\}$, and $\mathcal{T}_8=\{r_{18},r_{78}\}$. Given the decomposition of the network into $M=8$ teams, we construct the following communication schedules $\texttt{sched}_{ij}$ for all robots r_{ij}, as discussed in Section III-A:

$$\begin{bmatrix}\texttt{sched}_{12}\\ \texttt{sched}_{23}\\ \texttt{sched}_{34}\\ \texttt{sched}_{45}\\ \texttt{sched}_{56}\\ \texttt{sched}_{67}\\ \texttt{sched}_{78}\\ \texttt{sched}_{81}\end{bmatrix}=\begin{bmatrix}1 & 2\\ 3 & 2\\ 3 & 4\\ 5 & 4\\ 5 & 6\\ 7 & 6\\ 7 & 8\\ 1 & 8\end{bmatrix}^{\omega}.$$

Notice that the period of these schedules is $T=2$. At epoch $k=1$ the robots in teams $\mathcal{T}_1$, $\mathcal{T}_3$, $\mathcal{T}_5$, $\mathcal{T}_5$, $\mathcal{T}_7$ communicate and decide on their next paths so that they re-connect after $T=2$ epochs, i.e., at epoch $k=3$. A video simulation of the trajectories of the robots and targets and the sequence of meeting times and locations can be found in [41].

In what follows, we compare our proposed Algorithm 1 to an algorithm that preserves network connectivity for all time and an intermittent connectivity heuristic that employs the same decomposition of robots into teams, but selects random meeting locations for each team. Specifically, for the connectivity preserving approach, we assume that the network of $N=8$ robots maintains a fixed connected configuration throughout the whole experiment. Then, to design informative paths for this network, we apply the RRT* algorithm to design paths for the geometric center of this configuration, so that the uncertainty of the most uncertain target drops below a threshold δ. Once the connected network travels along the resulting path, the RRT* algorithm is executed again to find the next informative path. Notice that the paths constructed for the all-time connected network are asymptotically optimal. On the other hand, for the heuristic approach the path segments $\mathbf{p}_{ij}^k$ are selected to be the geodesic paths that connect the initial locations to the randomly selected meeting location. The geodesic paths are generated using the toolbox in [42]. When traveling along the paths designed by this heuristic, the time constraint is not satisfied and the robots need to wait at their meeting locations for their team members.

Figure 2 shows the evolution of the localization error $e_\ell(t)=\|\hat{\mathbf{x}}_g(t)-\mathbf{x}(t)\|$, and the uncertainty metric $\lambda_n(\mathbf{C}(t))$

TABLE I: Comparison with Alternative Approaches

	4 Targets			8 Targets		
	all-time	heuristic	intermittent	all-time	heuristic	intermittent
$\bar{e}_\ell$ (m)	1.030 $\pm$ 0.466	1.993 $\pm$ 0.566	0.879 $\pm$ 0.461	2.517 $\pm$ 0.810	1.201 $\pm$ 0.570	0.929 $\pm$ 0.561
$\bar{\lambda}$ (m^2)	0.226 $\pm$ 0.061	0.405 $\pm$ 0.181	0.167 $\pm$ 0.084	0.563 $\pm$ 0.129	0.334 $\pm$ 0.161	0.177 $\pm$ 0.068

as a function of time t for both approaches compared to our method. In the error metric $e_\ell(t)$, $\mathbf{x}(t)$ is a vector that stacks the true positions of all targets and $\hat{\mathbf{x}}_g(t)$ stands for the estimate of $\mathbf{x}(t)$ and corresponding covariance matrix $\mathbf{C}(t)$ assuming global fusion of all measurements taken by all robots up to time t. Notice that our method outperforms the algorithm that requires all robots to remain connected for all time. The reason is that our proposed algorithm allows the robots to disconnect in order to visit multiple targets simultaneously, which cannot happen using an all-time connectivity algorithm. Observe also that our algorithm performs better than the heuristic approach which justifies the use of a sampling-based algorithm to solve (5) instead of a computationally inexpensive method that picks random meeting locations that are connected to each other through geodesic paths.

The heuristic approach performs better in Figure 2 than the algorithm that enforces end-to-end network connectivity for all time; this is not always the case. To elaborate more, we compare the performance of the algorithms for 4 and 8 targets in Table I in terms of the average localization error $\bar{e}_\ell = 1/t_{\text{end}} \sum_{t=0}^{t_{\text{end}}} \|\hat{\mathbf{x}}_g(t) - \mathbf{x}(t)\|$, and the average uncertainty $\bar{\lambda} = 1/t_{\text{end}} \sum_{t=0}^{t_{\text{end}}} \lambda_n(\mathbf{C}(t))$, where t_{end} is the total number of discrete time steps t. Observe that as we decrease the number of targets, the all-time connectivity algorithm performs better than the heuristic approach. The reason for this is that the all-time connectivity approach forces the network of robots to visit the targets sequentially. Thus, for larger number of targets it takes longer to revisit a specific target and this results in uncertainty spikes. On the other hand, the heuristic approach selects the meeting locations randomly. Thus, as we populate the domain with targets, the paths designed by the heuristic approach cross nearby a target with a larger probability which improves its performance. Our proposed algorithm outperforms both approaches regardless of the number of targets. In Table I, for the case of 4 targets, we have used a network with $N = 4$ robots divided in the following teams $\mathcal{T}_1 = \{r_{12}, r_{14}\}$, $\mathcal{T}_2 = \{r_{12}, r_{23}\}$, $\mathcal{T}_3 = \{r_{23}, r_{34}\}$, and $\mathcal{T}_4 = \{r_{34}, r_{14}\}$.

Note that $\hat{\mathbf{x}}_g(t)$ is a fictitious estimate, as it is based on global fusion of measurements taken by all robots. Nevertheless, even the local estimation results for our proposed method are considerably better than the results of competitive methods shown in Figure 2. Particularly, for 8 targets the numerical error values for team $\mathcal{T}_2$, as an example, are $\bar{e}_\ell = 1.044 \pm 0.628$m and $\bar{\lambda} = 0.177 \pm 0.071$m^2, respectively; compare with Table I. This is important since in practice, a user can have access to the information of an individual robot and not the whole network, as this would defeat the purpose of assuming robots with limited communication range. In a similar scenario in the case of all-time connectivity, the whole network needs to be recalled to an access point to collect information from the robots, which would disrupt the estimation task altogether.

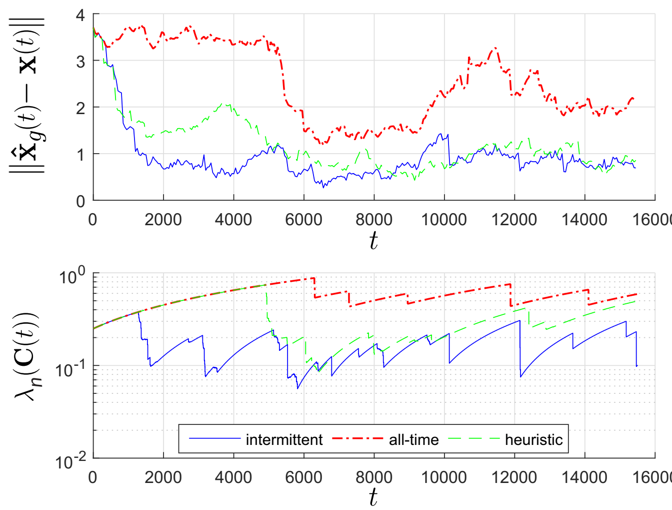

Fig. 2: Comparison of the evolution of the localization error $e_\ell(t) = \|\hat{\mathbf{x}}_g(t) - \hat{\mathbf{x}}(t)\|$ and the uncertainty metric $\lambda_n(\mathbf{C}(t))$ between our proposed control framework, an all-time connectivity algorithm, and a heuristic approach.

Finally, let $\mathcal{G}^1_\mathcal{T}$ denote the graph of teams that we have considered so far for $N = 8$ robots. In [1], we also examine the performance of our algorithm for a second denser graph $\mathcal{G}^2_\mathcal{T}$ with $M = 5$ teams. When the teams are constructed as in $\mathcal{G}^1_\mathcal{T}$, the proposed algorithm can better estimate the positions of the targets. The reason is that the communication schedules $\texttt{sched}_{ij}$ in $\mathcal{G}^1_\mathcal{T}$ allow more teams to visit multiple targets at the same time compared to when the teams are determined according to $\mathcal{G}^2_\mathcal{T}$. However, as predicted by Proposition 3.2, the delay for the graph $\mathcal{G}^2_\mathcal{T}$ is $D_{\mathcal{G}^2_\mathcal{T}} = (3-1)\times 2 = 4$ which is smaller than $D_{\mathcal{G}^1_\mathcal{T}} = (2-1)\times 5 = 5$. Consequently, appropriate selection of the graph $\mathcal{G}_\mathcal{T}$ depends on the specific problem and should achieve a balance between less uncertainty and smaller delays; see [1] for more details.

V. Conclusion

In this paper, we considered the problem of distributed state estimation using intermittently connected mobile robot networks. Robots were assumed to have limited communication capabilities and, therefore, exchanged their information intermittently only when they were sufficiently close to each other. To the best of our knowledge, this is the first framework for concurrent state estimation and intermittent communication control that does not rely on any network connectivity assumption and can be applied to large-scale networks. We presented simulation results that demonstrated significant improvement in estimation accuracy compared to methods that maintain an end-to-end connected network for all time. The improvement becomes more considerable as information becomes localized and sparse and the communication and sensing ranges of the robots become smaller compared to the size of the domain.

REFERENCES

[1] R. Khodayi-mehr, Y. Kantaros, and M. M. Zavlanos, "Distributed state estimation using intermittently connected robot networks," *IEEE Transactions on Robotics*, vol. 35, no. 3, pp. 709–724, June 2019.

[2] D. Fox, W. Burgard, H. Kruppa, and S. Thrun, "A probabilistic approach to collaborative multi-robot localization," *Autonomous Robots*, vol. 8, no. 3, pp. 325–344, 2000.

[3] S. I. Roumeliotis and G. A. Bekey, "Distributed multirobot localization," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 781–795, 2002.

[4] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part i," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 2, pp. 99–110, 2006.

[5] E. D. Nerurkar, K. J. Wu, and S. I. Roumeliotis, "C-KLAM: Constrained keyframe-based localization and mapping," in *IEEE International Conference on Robotics and Automation*, Hong Kong, China, May-June 2014, pp. 3638–3643.

[6] N. Atanasov, J. Le Ny, K. Daniilidis, and G. J. Pappas, "Decentralized active information acquisition: theory and application to multi-robot slam," in *Proceedings of IEEE International Conference on Robotics and Automation*. IEEE, 2015, pp. 4775–4782.

[7] K. Y. Leung, T. D. Barfoot, and H. H. Liu, "Decentralized cooperative SLAM for sparsely-communicating robot networks: A centralized-equivalent approach," *Journal of Intelligent & Robotic Systems*, vol. 66, no. 3, pp. 321–342, 2012.

[8] P. Jalalkamali and R. Olfati-Saber, "Information-driven self-deployment and dynamic sensor coverage for mobile sensor networks," in *Proceedings of American Control Conference*. IEEE, 2012, pp. 4933–4938.

[9] J. Vander Hook, P. Tokekar, and V. Isler, "Algorithms for cooperative active localization of static targets with mobile bearing sensors under communication constraints," *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 864–876, 2015.

[10] C. Freundlich, P. Mordohai, and M. M. Zavlanos, "Optimal path planning and resource allocation for active target localization," in *Proceedings of American Control Conference*. IEEE, 2015, pp. 3088–3093.

[11] C. Freundlich, Y. Zhang, and M. M. Zavlanos, "Distributed hierarchical control for state estimation with robotic sensor networks," *IEEE Transactions on Control of Network Systems*, vol. PP, no. 99, pp. 1–1, December 2017.

[12] S. Martínez and F. Bullo, "Optimal sensor placement and motion coordination for target tracking," *Automatica*, vol. 42, no. 4, pp. 661–668, 2006.

[13] G. Hollinger, S. Singh, J. Djugash, and A. Kehagias, "Efficient multi-robot search for a moving target," *The International Journal of Robotics Research*, vol. 28, no. 2, pp. 201–219, 2009.

[14] T. H. Chung, J. W. Burdick, and R. M. Murray, "A decentralized motion coordination strategy for dynamic target tracking," in *Proceedings of IEEE International Conference on Robotics and Automation*. IEEE, 2006, pp. 2416–2422.

[15] J. Derenick, J. Spletzer, and A. Hsieh, "An optimal approach to collaborative target tracking with performance guarantees," *Journal of Intelligent & Robotic Systems*, vol. 56, no. 1, pp. 47–67, 2009.

[16] R. Olfati-Saber, "Distributed tracking for mobile sensor networks with information-driven mobility," in *Proceedings of American Control Conference*. IEEE, 2007, pp. 4606–4612.

[17] G. Huang, K. Zhou, N. Trawny, and S. I. Roumeliotis, "A bank of maximum a posteriori (MAP) estimators for target tracking," *IEEE Transactions on Robotics*, vol. 31, no. 1, pp. 85–103, 2015.

[18] C. Freundlich, S. Lee, and M. M. Zavlanos, "Distributed active state estimation with user-specified accuracy," *IEEE Transactions on Automatic Control*, vol. 63, no. 2, pp. 418–433, 2018.

[19] R. Olfati-Saber, "Distributed kalman filtering for sensor networks," in *Proceedings of IEEE Conference on Decision and Control*. IEEE, 2007, pp. 5492–5498.

[20] A. Makarenko and H. Durrant-Whyte, "Decentralized data fusion and control in active sensor networks," in *International Conference on Information Fusion*, vol. 1, 2004, pp. 479–486.

[21] M. E. Campbell and N. R. Ahmed, "Distributed data fusion: Neighbors, rumors, and the art of collective knowledge," *IEEE Control Systems*, vol. 36, no. 4, pp. 83–109, 2016.

[22] B. J. Julian, M. Angermann, M. Schwager, and D. Rus, "Distributed robotic sensor networks: An information-theoretic approach," *International Journal of Robotics Research*, vol. 31, no. 10, pp. 1134–1154, 2012.

[23] G. A. Hollinger and G. S. Sukhatme, "Sampling-based motion planning for robotic information gathering." in *Robotics: Science and Systems*. Citeseer, 2013, pp. 72–983.

[24] ——, "Sampling-based robotic information gathering algorithms," *The International Journal of Robotics Research*, vol. 33, no. 9, pp. 1271–1287, 2014.

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

[26] Y. Kim and M. Mesbahi, "On maximizing the second smallest eigenvalue of a state-dependent graph laplacian," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 116–120, 2006.

[27] M. M. Zavlanos and G. J. Pappas, "Potential fields for maintaining connectivity of mobile networks," *IEEE Transactions on Robotics,*, vol. 23, no. 4, pp. 812–816, 2007.

[28] M. Ji and M. B. Egerstedt, "Distributed coordination control of multi-agent systems while preserving connectedness." *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 693–703, August 2007.

[29] M. Zavlanos and G. Pappas, "Distributed connectivity control of mobile networks," *IEEE Transactions on Robotics*, vol. 24, no. 6, pp. 1416–1428, 2008.

[30] P. Yang, R. A. Freeman, G. J. Gordon, K. M. Lynch, S. S. Srinivasa, and R. Sukthankar, "Decentralized estimation and control of graph connectivity for mobile sensor networks," *Automatica*, vol. 46, no. 2, pp. 390–396, 2010.

[31] E. Montijano, J. Montijano, and C. Sagues, "Adaptive consensus and algebraic connectivity estimation in sensor networks with Chebyshev polynomials," in *50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, USA, December 2011, pp. 4296–4301.

[32] M. Franceschelli, A. Gasparri, A. Giua, and C. Seatzu, "Decentralized estimation of Laplacian eigenvalues in multi-agent systems," *Automatica*, vol. 49, no. 4, pp. 1031–1036, 2013.

[33] L. Sabattini, N. Chopra, and C. Secchi, "Decentralized connectivity maintenance for cooperative control of mobile robotic systems," *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1411–1423, 2013.

[34] M. M. Zavlanos, "Synchronous rendezvous of very-low-range wireless agents," in *49th IEEE Conference on Decision and Control*, Atlanta, GA, USA, December 2010, pp. 4740–4745.

[35] Y. Kantaros and M. M. Zavlanos, "Distributed intermittent connectivity control of mobile robot networks," *Transactions on Automatic Control*, vol. 62, no. 7, pp. 3109–3121, July 2017.

[36] ——, "Simultaneous intermittent communication control and path optimization in networks of mobile robots," in *55th IEEE Conference on Decision and Control*, Las Vegas, NV, December 2016, pp. 1794–1795.

[37] Y. Kantaros, M. Guo, and M. M. Zavlanos, "Temporal logic task planning and intermittent connectivity control of mobile robot networks," *IEEE Transactions on Automatic Control*, pp. 1–16, 2019. [Online]. Available: DOI:10.1109/TAC.2019.2893161

[38] Y. Kantaros and M. Zavlanos, "Distributed intermittent communication control of mobile robot networks under time-critical dynamic tasks," in *IEEE International Conference on Robotics and Automation*, Brisbane, Australia, May 2018, pp. 5028–5033.

[39] M. Guo and M. M. Zavlanos, "Multi-robot data gathering under buffer constraints and intermittent communication," *IEEE Transactions on Robotics*, no. 4, pp. 1082–1097, 2018.

[40] C. Godsil and G. Royle, *Algebraic Graph Theory*. New York: Springer-Verlag, 2001.

[41] "Simulation video - https://vimeo.com/262568673."

[42] K. J. Obermeyer and Contributors, "The VisiLibity library," `http://www.VisiLibity.org`, 2008, release 1.