MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Dynamic Sensor Scheduling for Spatio-temporal Monitoring of Water Bodies

Deshpande, Vedang M.; Vinod, Abraham P.

TR2025-117 August 02, 2025

Abstract

We present a formulation for dynamic sensor scheduling for environment monitoring applications using multi-agent systems. The domain of interest is represented by a parameterized state-space model which captures spatio-temporal correlations of the environment. The parameters of the model are adapted online as new measurements become available. We introduce an improved greedy approach that simultaneously determines the optimal sensing locations and assigns agents to these locations; and this approach minimizes travel costs incurred by the mobile agents while satisfying the dynamic reachability constraints. We provide performance guarantees for the algorithms under certain conditions and derive their polynomial-time computational complexities. We demonstrate the proposed approach for bio-mass monitoring applications in large water bodies.

IEEE Conference on Control Technology and Applications (CCTA) 2025

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Dynamic Sensor Scheduling for Spatio-temporal Monitoring of Water Bodies

Vedang M. Deshpande†, Abraham Vinod

***Abstract*— We present a formulation for dynamic sensor scheduling for environment monitoring applications using multi-agent systems. The domain of interest is represented by a parameterized state-space model which captures spatio-temporal correlations of the environment. The parameters of the model are adapted online as new measurements become available. We introduce an improved greedy approach that simultaneously determines the optimal sensing locations and assigns agents to these locations; and this approach minimizes travel costs incurred by the mobile agents while satisfying the dynamic reachability constraints. We provide performance guarantees for the algorithms under certain conditions and derive their polynomial-time computational complexities. We demonstrate the proposed approach for bio-mass monitoring applications in large water bodies.**

I. Introduction

Monitoring of environmental processes using mobile agents has gained substantial attention in recent years [1]–[3]. Such environmental processes are spatio-temporally correlated and often span large geographical areas. However, the resources available for data collection and monitoring, such as sensors and mobile agents (e.g., aerial drones, ground and underwater vehicles), are typically limited. This raises an important question: how can we efficiently utilize these limited resources to achieve optimal monitoring performance?

Many studies have tackled this problem [4], [5], offering different approaches to address its various aspects including energy aware motion planning [4], informative path planning [5]. Recently, there has been a significant focus on development of data-driven methods for monitoring of spatio-temporal processes that are difficult to model using physical principles [5]. However, these methods may require large amount of data collection or yield suboptimal results for processes that can be modeled, at least partially, using physics.

In this paper, we consider the monitoring problem for a class of spatio-temporal processes which can be represented by state-space models wherein certain aspects of the model may be unknown or subject to uncertainties. This is motivated by the characteristic challenges encountered in bio-mass monitoring applications in large water bodies. Monitoring, prediction and control of bio-mass (e.g. algae and bacteria) in both natural and man-made water bodies is an important problem because unchecked bio-mass growth may pose serious threats to the public health and infrastructure [6], [7]. Consequently, several previous studies have dedicated efforts to develop predictions models of bio-mass growth [6], [8], [9]. However, several parameters of these models may be subject to uncertainties and require real-world measurements for model calibration efforts and accurate predictions [9].

Remote satellite images of the water bodies are routinely used for monitoring algal blooms over large geographical areas [7]. However, such measurements may be expensive to obtain, often affected by the weather conditions and provide only preliminary guidance for deploying additional local monitoring resources [7]. Therefore, monitoring efforts have to rely on physically collected samples from water bodies for accurate measurements. In practice, locations of physical measurements are often fixed a priori or selected randomly [10], which may lead to sub-optimal monitoring performance over longer durations. How to optimally deploy a dynamic team of mobile agents for the bio-mass monitoring task – remains largely an unexplored research area.

To this end, we present a formulation to optimally monitor a class of spatio-temporal processes represented by state-space models using a team of mobile agents. The primary *contribution* this work is the development of greedy algorithms that concurrently determine the optimal sensing locations and assigns agents to these locations. This approach minimizes travel costs while satisfying the dynamic constraints imposed on the mobile agents. We provide performance guarantees for the algorithms under certain conditions and derive their polynomial-time computational complexities.

The greedy methods have been widely used for solving sensor/actuator selection problems [11]–[14] because they provide tractable solutions to otherwise intractable NP-hard combinatorial optimization problems. These methods also enjoy certain performance guarantees [15]. The key novelty of this work lies in the explicit integration of dynamic constraints imposed on mobile agents into the greedy algorithm, ensuring that the chosen sensor locations result in a dynamically feasible assignment for the agents.

II. Problem Formulation

Consider a spatio-temporal environment, i.e., an environment where quantities of interest varies both in space and time. Denoting the spatial distribution of the quantites of interest by the vector $x_k \in \mathbb{R}^{n_x}$, we model its temporal evolution,

$$x_{k+1} = f(x_k, u_k; \theta) + w_k, \tag{1}$$

where the evolution function $f(\cdot)$ is parameterized by the parameter vector $\theta \in \mathbb{R}^{n_\theta}$ which may be unknown. $u_k \in \mathbb{R}^{n_u}$

† Corresponding author.
The authors are with Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA, 02139. Email: deshpande@merl.com, abraham.p.vinod@ieee.org

denotes known external signals that affect the process evolution and $w_k \in \mathbb{R}^{n_x}$ denotes additive zero mean Gaussian uncertainty of covariance Q_k which may arise due to external disturbances or unmodeled process dynamics.

The process model (1) may arise as a result of spatio-temporal discretizations of partial differential equations (PDEs) or ordinary differential equations (ODEs) that describe the underlying process over continuous domains, e.g., see (18) in Section IV. The process model (1) admits a broad class of spatio-temporal environmental processes including bio-mass growth in water bodies [9].

The monitoring of processes characterized by (1) entails accurate dynamic estimation of the state vector x_k and the parameters θ using limited sensor data. We assume the following measurement model.

$$y_k^{\{i\}} = c_i^\top x_k + \eta_k^{\{i\}}, \tag{2}$$

where $y_k^{\{i\}} \in \mathbb{R}$ denotes the measurement obtained from the i^{th} sensor and $i \in \mathcal{S}$, and the set $\mathcal{S} := \{1, 2, \cdots n_{\mathcal{S}}\}$ of cardinality $|\mathcal{S}| = n_{\mathcal{S}}$ denotes the set of *all* available sensors. $\eta_k^{\{i\}} \in \mathbb{R}$ denotes the zero mean Gaussian noise with covariance σ_i^2 that corrupts the sensor measurement.

The sensor measurement model is assumed to be linear for the simplicity of discussion and the vectors c_i are assumed to be free of uncertainty as sensor measurement models are typically known. However, these assumptions can be readily relaxed in the technical details presented in the following sections.

Since x_k denotes the process state variables discretized over a spatial grid, the elements of x_k correspond to the process variables at different grid locations. Therefore, sensors in the set $\mathcal{S}$ may correspond to different locations where a measurement can be obtained. Since environmental processes are typically modeled over large geographical areas, the number of total available sensors or sensing locations in $n_{\mathcal{S}}$ can be very large. However, the number of obtainable measurements is often constrained by the limited resources, i.e., the availability of mobile agents (e.g., aerial drones or underwater vehicles) that travel to specific locations to obtain the measurements.

In general, identifying an *optimal* time-varying (sub)set of sensors (or sensing locations) that provides best monitoring or estimation performance over the entire monitoring duration is non-trivial because it is a combinatorial NP-hard problem. For brevity of discussion, we restrict the problem scope wherein an agent can obtain a measurement from at most one sensing location at each time step. Moreover, each agent must be assigned a dynamically feasible sensing task at each time step to ensure full utilization of the available mobile agents. Therefore, in addition to optimizing the estimation performance, we are also interested in identifying a dynamically feasible assignment of agents to the sensing locations. Furthermore, agent assignments incur some assignment-costs, e.g., cost of travel from current location to the next assigned location. Therefore, it is also of interest to identify an agent-sensor assignment that incurs minimal travel costs. Finally, this problem of identifying the sensors and tasking the agents must be dynamically solved at every time step as the process (1) evolves over time. To this end, we tackle the problem of dynamic sensor scheduling which is stated below.

Problem 1. Design a tractable, iterative algorithm that at time step k determines an optimal sensor subset $\mathcal{S}_k$ of cardinality N_{ag} that: (i) minimizes the uncertainty in the estimates of x_k and θ; (ii) yields a dynamically feasible assignment of agents to the identified sensing locations; and (iii) incurs a minimal travel cost.

A. Data Assimilation using EKF

We consider an extended Kalman filter (EKF) for assimilating sensor measurements and estimating the state vector x_k and parameters θ [16]. We rewrite (1) and (2) in the following form that is amenable for simultaneous state and parameter estimation.

$$\tilde{x}_{k+1} = \begin{bmatrix} x_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} f(x_k, u_k; \theta_k) \\ \theta_k \end{bmatrix} = \tilde{f}(\tilde{x}_k, u_k) + \tilde{w}_k \tag{3a}$$

$$y_k^{\{i\}} = \tilde{c}_i{}^\top \tilde{x}_k + \eta_k^{\{i\}}, \tag{3b}$$

where $\tilde{x}_k := \begin{bmatrix} x_k^\top & \theta_k^\top \end{bmatrix}^\top$ is the augmented state vector to be estimated, $\tilde{w}_k$ denotes the augmented zero-mean Gaussian process uncertainty with block-diagonal covariance matrix $\tilde{Q}_k := \mathsf{blkdiag}(Q_k, Q_k^\theta)$. The noise covariance Q_k^θ corresponding to the parameters is typically assigned a small value. Furthermore, the sensor measurement model is updated to account for additional states in the augmented state vector, i.e, $\tilde{c}_i = \begin{bmatrix} c_i^\top & 0_{n_\theta}^\top \end{bmatrix}^\top$, where 0_{n_θ} denotes a zero vector in $\mathbb{R}^{n_\theta}$.

The posterior and prior distributions of the state estimate at time k are assumed to be Gaussian and characterized by the mean-covariance pairs $(\tilde{x}_k^+, \tilde{P}_k^+)$ and $(\tilde{x}_k^-, \tilde{P}_k^-)$ respectively. The EKF equations to estimate the augmented vector $\tilde{x}_k$ are presented next.

Using the augmented model (3), the equations corresponding to the prediction or time-update step of EKF are given as follows:

$$\tilde{x}_k^- = \tilde{f}(\tilde{x}_{k-1}, u_{k-1}) \tag{4a}$$

$$\tilde{P}_k^- = \tilde{F}_{k-1} P_{k-1}^+ \tilde{F}_{k-1}^\top + \tilde{Q}_{k-1} \tag{4b}$$

where

$$\tilde{F}_k := \frac{\partial \tilde{f}}{\partial \tilde{x}}(\tilde{x}_k, u_k) = \begin{bmatrix} \frac{\partial f}{\partial x}(x_k, u_k; \theta_k) & \frac{\partial f}{\partial \theta}(x_k, u_k; \theta_k) \\ 0 & I \end{bmatrix} \tag{5}$$

and I denotes an identity matrix of the suitable dimension.

Given the prior estimate (4), the posterior estimate is obtained by the following measurement update step of the EKF

$$\tilde{K}_k = (\tilde{P}_k^- \tilde{H}_k^\top)(\tilde{H}_k \tilde{P}_k^- \tilde{H}_k^\top + R_k)^{-1} \tag{6a}$$

$$\tilde{x}_k^+ = \tilde{x}_k^- + \tilde{K}_k \big(y_k - \tilde{H}_k \tilde{x}_k^-\big) \tag{6b}$$

$$\tilde{P}_k^+ = (I - \tilde{K}_k \tilde{H}_k) \tilde{P}_k^- \tag{6c}$$

where

$$\tilde{H}_k = \begin{bmatrix} c_{i_1} & c_{i_2} & \cdots \\ 0 & 0 & \cdots \end{bmatrix}^\top \tag{7}$$

is the measurement matrix corresponding to the set of selected sensors $\mathcal{S}_k = \{i_1, i_2, \cdots i_{N_{\text{ag}}}\} \subseteq \mathcal{S}$ at time step k.

B. Estimation Quality

The uncertainty in the estimate $\tilde{x}_k$ is commonly quantified using the volume of the confidence ellipsoid centered at the mean. This volume, characterized by the determinant of the covariance matrix of the Gaussian distribution, serves as a widely accepted measure of estimation uncertainty [12].

Accordingly, we define the following objective

$$J_{\text{est}}(\mathcal{S}_k) = \mathsf{logdet}\big(P_k^+\big), \tag{8}$$

which we aim to minimize. This objective quantifies the uncertainty in posterior filter estimates as a function of the sensor set $\mathcal{S}_k$ selected at time step k. Notably, the log-determinant function, $\mathsf{logdet}(\cdot)$, is a submodular function of selected sensors, which allows us to provide certain performance guarantees for the greedy algorithms (see Section III).

C. Dynamical Constraints and Cost of Travel

We now turn our attention to ensuring that the sensor schedule generated is feasible for the mobile agents team. Specifically, we restrict the possible sensor locations in consideration to only those locations that are reachable from the current agent locations. In other words, these reachability constraints enforce physical constraints arising from dynamics of the mobile agents.

For a grid world, given the current agents team location, we can compute the set of locations reachable by the team with a pre-specified number of steps N_{steps} using any of the existing shortest path algorithms, e.g., Dijkstra, A* [17]. Here, lower N_{steps} facilitates more frequent measurements, while larger N_{steps} allows for broader coverage of the environment. As a side-product, we also obtain D_{ij}, the shortest distance between the mobile agent i's current location and a candidate sensor location j in the grid.

In our implementation, we assume that the locations or grid-cells that are within N_{steps} steps from the current location of an agent are reachable, where moving to an adjacent cell counts as one step and moving to a diagonal location is not allowed.

Thus, for a given location of the i^{th} mobile agent, we can identify C_i – the *coverage set* of the i^{th} agent, i.e., the set of all sensors that are reachable from its current location. In particular,

$$C_i := \{j | j \in \mathcal{S}, D_{ij} \leq N_{\text{steps}}\}. \tag{9}$$

For clarity of discussion, we denote assignment of i^{th} agent to the sensing location $j \in \mathcal{S}$ by an ordered tuple (i, j). We next define *dynamically feasible assignment* as follows.

Definition 1. *An assignment set*

$$\mathcal{A}_k := \{(i, j_i) | j_i \in \mathcal{S}, i = 1, 2, \cdots, N_{ag}\} \tag{10}$$

is said to be dynamically feasible if $j_i \in C_i \ \forall i \in \{1, 2, \cdots, N_{ag}\}$, *and* $j_{i_1} \neq j_{i_2} \ \forall i_1, i_2 \in \{1, 2, \cdots, N_{ag}\}$

The definition essentially enforces the constraint that every agent gets assigned a sensor that lies in its coverage set, and no two agents get assigned the same sensor location.

The cost of travel for the assignment (i, j) is simply D_{ij}. Therefore, the total cost of travel for an assignment set defined in (10) is given by

$$J_{\text{travel}}(\mathcal{A}_k) = \sum_i D_{ij_i} \tag{11}$$

which we are interested in minimizing. This measure can be readily extended to incorporate additional factors such as energy consumption, battery limitations, travel duration, and other operational costs [18].

In this work, we make the following assumptions — (i) the sensor scheduling need not account for collision avoidance between agents, and (ii) there are no obstacles/no-fly zones in the environment. The first assumption may be relaxed in practice by considering multi-agent motion planners at the cost of increased computational effort. We make the second assumption for the sake of brevity, and can be easily relaxed by appropriately defining coverage sets for the agents.

III. Main Results

Considering Problem 1, and the objectives (8) and (11), we state the *bi-criterion* optimization problem as follows

$$\min_{\mathcal{A}_k} \begin{bmatrix} J_{\text{est}} \\ J_{\text{travel}} \end{bmatrix} \quad \text{subject to } \mathcal{A}_k \text{ being dynamically feasible.} \tag{12}$$

It is noted that the set of selected sensor $\mathcal{S}_k$ is already a part of $\mathcal{A}_k$, hence, it is not written as an explicit optimization variable in (12). Problem (12) is a vector optimization which can not be directly solved. A standard approach to solve such problems is to scalarize of the objective function [19]. However, such an approach may not be tractable in this case as $\mathcal{A}_k$ is a combinatorial variable. Instead, we introduce a greedy algorithm for simultaneous sensor selection and agent assignment which yields a feasible solution to (12) in polynomial time.

The remainder of this section is structured as follows. First, we briefly review the classical greedy approach for minimization of only J_{est}. Next, we summarize an improved greedy approach to solve (12) and discuss its computational complexity.

A. Minimizing J_{est}

In this section, we briefly review the classical approach for minimization of J_{est}. That is, we only consider the first objective in (12) while ignoring J_{travel} and the assignment constraints.

First, we note that

$$\underset{\mathcal{S}_k, |\mathcal{S}_k| = N_{\text{ag}}}{\arg\min} \ J_{\text{est}}(\mathcal{S}_k) = \underset{\mathcal{S}_k, |\mathcal{S}_k| = N_{\text{ag}}}{\arg\max} \ J'_{\text{est}}(\mathcal{S}_k) \tag{13}$$

where

$$J'_{\text{est}}(\mathcal{S}_k) := \text{logdet}\big((\tilde{P}_k^+)^{-1}\big) \tag{14}$$

$$= \text{logdet}\left((\tilde{P}_k^-)^{-1} + \sum_{i\in\mathcal{S}_k} \sigma_i^{-2}\tilde{c}_i\tilde{c}_i^\top\right) \tag{15}$$

which follows from the information form of Kalman update [16], and $\sigma_i^{-2}\tilde{c}_i\tilde{c}_i^\top$ denotes the information matrix of the sensor $i \in \mathcal{S}_k$.

We note the following result.

Lemma 1. *The function $J'_{est} : 2^{\mathcal{S}} \mapsto \mathbb{R}$ is a submodular monotonic function $\forall\, \mathcal{S}_k \subseteq \mathcal{S}$.*

Proof. The proof follows similar arguments as in [12, Lemma 1]. □

Maximization of J'_{est} corresponds to the maximization of monotonic submodular function. Therefore, classical greedy methods are guaranteed to yield a solution that is within a factor of $(1 - 1/e)$ of the optimal solution [15]. We summarize the classical greedy approach [12] in Algorithm 1 for maximization of J'_{est}.

Algorithm 1 GREEDY SENSOR SELECTION (CLASSICAL)

1: Input: $\mathcal{S}$
2: Initialize: $\tilde{P}_k^{(+)} = \tilde{P}_k^-$, $\mathcal{S}_k = \emptyset$
3: **for** $j = 1, 2, \cdots, N_{\text{ag}}$ **do**
4: $\quad \mathcal{S}_k^{\text{avail}} \leftarrow \mathcal{S} \setminus \mathcal{S}_k$
5: $\quad s^* \leftarrow \arg\max_{s\in\mathcal{S}_k^{\text{avail}}} \text{logdet}\left((\tilde{P}_k^+)^{-1} + \sigma_s^{-2}\tilde{c}_s\tilde{c}_s^\top\right)$
6: $\quad \mathcal{S}_k \leftarrow \mathcal{S}_k \cup \{s^*\}$
7: $\quad (\tilde{P}_k^+)^{-1} \leftarrow (\tilde{P}_k^+)^{-1} + \sigma_{s^*}^{-2}\tilde{c}_{s^*}\tilde{c}_{s^*}^\top$
8: **end for**

In each iteration of the greedy algorithm, a sensor is selected that results in in the largest increment in the objective J'_{est}. This is effectively implemented by updating the covariance matrix in the last line of the algorithm for every selected sensor which utilizes the sequential measurements assimilation of the Kalman update [16] for a simplified evaluation of objective increments in Step 5.

Algorithm 1 ignores J_{travel} and does not guarantee the dynamic feasibility of the chosen sensor set. In the following subsection, we present an improved greedy algorithm that addresses these limitations of Algorithm 1.

B. Simultaneous Greedy Sensor Selection and Agent Assignment

We introduce a greedy approach in Algorithm 2 to solve (12). A greedy selection of $\mathcal{S}_k$ to maximize the objective function J'_{est} entails, in each iteration, selection of one sensor that results in the largest increment in the objective function, until N_{ag} sensors have been selected. Heuristics incorporated in Algorithm 2 aim to minimize the travel cost while guaranteeing that the selected sensor set yields a dynamically feasible assignment of agents and the sensors.

The input to Algorithm 2 is the coverage sets of all agents at the current time step as defined in (9), and it runs for N_{ag} iterations to select one sensor in each iteration. The set of available sensors to choose from, $\mathcal{S}_k^{\text{avail}}$, is determined to be all sensors in $\bigcup C$ except $\mathcal{S}_k$ – the sensors already selected in the previous iterations of the algorithm. This ensures that no sensor gets selected more than once.

In lines 5–7 of the algorithm, a sensor is selected that yields the maximum increment in the objective value, and ties, if any, are settled arbitrarily; and the covariance matrix is updated for the selected sensor.

In lines 8–11 of the algorithm, the selected sensor is assigned to one of the nearest agents that is currently untasked and reachable from the selected sensor. Note that D_{is^*} denotes the distance of i^{th} agent from the sensor s^*, and it is our objective to minimize the travel cost incurred by the mobile agent. The coverage set of the assigned agent is removed from C in line 11 to ensure that each agent gets assigned exactly one sensor.

Algorithm 2 GREEDY SENSOR SELECTION WITH SIMULTANEOUS AGENT ASSIGNMENT

1: Input: $C = \{C_1, C_2 \cdots C_{N_{\text{ag}}}\}$
2: Initialize: $\tilde{P}_k^{(+)} = \tilde{P}_k^-$, $\mathcal{S}_k = \emptyset$, $\mathcal{A}_k = \emptyset$
3: **for** $j = 1, 2, \cdots, N_{\text{ag}}$ **do**
4: $\quad \mathcal{S}_k^{\text{avail}} \leftarrow \bigcup C \setminus \mathcal{S}_k$
5: $\quad s^* \leftarrow \arg\max_{s\in\mathcal{S}_k^{\text{avail}}} (\tilde{c}_s^\top \tilde{P}_k^+ \tilde{c}_s)/\sigma_s^2$
6: $\quad \mathcal{S}_k \leftarrow \mathcal{S}_k \cup \{s^*\}$
7: $\quad \tilde{P}_k^+ \leftarrow \tilde{P}_k^+ - \tilde{P}_k^+\tilde{c}_{s^*}\left(1 + \frac{\tilde{c}_{s^*}^\top \tilde{P}_k^+ \tilde{c}_{s^*}}{\sigma_{s^*}^2}\right)^{-1} \tilde{c}_{s^*}^\top \tilde{P}_k^+$
8: $\quad \mathcal{N} \leftarrow \{i \mid s^* \in C_i \in C\}$
9: $\quad i^* \leftarrow \arg\min_{i\in\mathcal{N}} D_{is^*}$
10: $\quad$ Task agent i^* to sense s^*: $\mathcal{A} \leftarrow \mathcal{A} \cup \{(i^*, s^*)\}$
11: $\quad C \leftarrow C \setminus \{C_{i^*}\}$
12: **end for**

Note that Steps 5–7 of Algorithms 1 and 2 are equivalent. Using matrix determinant lemma,

$$\det\left((\tilde{P}_k^+)^{-1} + \sigma_i^{-2}\tilde{c}_i\tilde{c}_i^\top\right) = \left(1 + \frac{\tilde{c}_i^\top \tilde{P}_k^+ \tilde{c}_i}{\sigma_i^2}\right) \det\left((\tilde{P}_k^+)^{-1}\right), \tag{16}$$

the Step 5 of Algorithm 1 can be equivalently reduced to Step 5 of Algorithm 2. Similarly, Step 7 of Algorithm 1 can be equivalently written in standard covariance update form shown in Step 7 of Algorithm 2. These equivalent substitutions reduce the overall time complexity of the algorithm, which is discussed next.

Proposition 1. *Algorithm 2 has the worst-case time complexity of*

$$O\left(N_{ag} n_C (n_x + n_\theta)^2 + n_C N_{ag}^3\right),$$

where $n_C := |\bigcup C|$.

Proof. Step 5 is a quadratic term with a complexity of $O((n_x + n_\theta)^2)$. This term is computed $O(n_C)$ times in each iteration. Step 7 contains outer product of vectors and matrix addition, both of which have a computational complexity of $O((n_x + n_\theta)^2)$. Therefore, Steps 5 and 7 together have a

total complexity of $O(n_C(n_x + n_\theta)^2)$ per iteration. In each iteration, Step 8 checks set membership which has the worst-case complexity of $O(n_C N_{\text{ag}}^2)$. Step 9 finds the minimum in a vector with the worst-case complexity of $O(n_C)$. Therefore, total complexity of all iterations of Algorithm 2 is $O(N_{\text{ag}} n_C(n_x + n_\theta)^2 + n_C N_{\text{ag}}^3)$. □

The heuristics incorporated in Algorithm 2 to satisfy the dynamic feasibility of assignment mean that its optimality guarantees are not yet established. Nevertheless, it performs well in practice, as evidenced in the numerical results section. However, some performance guarantees can be offered under specific conditions, which are discussed in the following subsection.

C. Common Coverage Sets

We consider a special scenario in which coverage sets are identical for all agents, i.e., $C_1 = C_2 = \cdots = C_{N_{\text{ag}}}$. Such a scenario may arise when all agents are located in the close proximity of each other or when allowable N_{steps} used in identifying reachable sets is sufficiently large to cover all candidate sensor locations.

When $C_1 = C_2 = \cdots = C_{N_{\text{ag}}}$, we can decouple the problem of sensor selection from agent assignment since any set $\mathcal{S}_k \subseteq \bigcup C$ of cardinality N_{ag} will result in a dynamically feasible assignment. In such a case, we may adopt a two-step approach summarized in Algorithm 3.

Algorithm 3 GREEDY SENSOR SELECTION FOLLOWED BY LINEAR ASSIGNMENT

1: Input: $C = \{C_1, C_2 \cdots C_{N_{\text{ag}}}\}$
2: Initialize: $\tilde{P}_k^{(+)} = \tilde{P}_k^-$, $\mathcal{S}_k = \emptyset$
3: **for** $j = 1, 2, \cdots, N_{\text{ag}}$ **do**
4: $\quad \mathcal{S}_k^{\text{avail}} \leftarrow \bigcup C \setminus \mathcal{S}_k$
5: $\quad s^* \leftarrow \arg\max_{s \in \mathcal{S}_k^{\text{avail}}} (\tilde{c}_s^\top \tilde{P}_k^+ \tilde{c}_s)/\sigma_s^2$
6: $\quad \mathcal{S}_k \leftarrow \mathcal{S}_k \cup \{s^*\}$
7: $\quad \tilde{P}_k^+ \leftarrow \tilde{P}_k^+ - \tilde{P}_k^+ \tilde{c}_{s^*} \left(1 + \frac{\tilde{c}_{s^*}^\top \tilde{P}_k^+ \tilde{c}_{s^*}}{\sigma_{s^*}^2}\right)^{-1} \tilde{c}_{s^*}^\top \tilde{P}_k^+$
8: **end for**
9: Given $\mathcal{S}_k$, solve (17) for an optimal assignment

We first greedily select N_{ag} sensors by ignoring the assignment Steps 8–11 of the Algorithm 2. Subsequently, we solve a *linear assignment problem* to assign the selected sensor locations to individual agents to minimize the total cost of travel. Recall that linear assignment problems are a special class of integer linear programs, that admit polynomial time solutions [20].

In our application, consider the scenario of N_{ag} mobile agents that must be assigned N_{ag} selected sensing locations. We formulate an assignment problem with the following structure,

$$\begin{aligned} \text{minimize} \quad & \textstyle\sum_{1 \le i \le N_{\text{ag}}, 1 \le j \le N_{\text{ag}}} D_{ij} Z_{ij} \\ \text{subject to} \quad & Z \in \{0,1\}^{N_{\text{ag}} \times N_{\text{ag}}}, \\ & 1_{1 \times N_{\text{ag}}} Z \le 1, Z 1_{N_{\text{ag}} \times 1} \le 1. \end{aligned} \tag{17}$$

Specifically, given a collection of costs $D \in \mathbb{R}^{N_{\text{ag}} \times N_{\text{ag}}}$ where D_{ij} is the shortest distance between the i^{th} agent and the j^{th} selected sensor location, the linear assignment problem seeks to identify an assignment Z where $Z_{ij} = 1$ implies that mobile agent i is assigned to the sensing location j, and the constraints $1_{1 \times N_{\text{ag}}} Z \le 1$, $Z 1_{N_{\text{ag}} \times 1} \le 1$ ensures that each sensing location is assigned at most one mobile agent and each mobile agent is assigned at most one sensing location, respectively.

Optimality of Algorithm 3:: Steps 3–8 of the Algorithm 3 is the classical greedy approach to maximize a monotonic submodular function, therefore, it is guaranteed to yield a solution that is within a factor of $(1 - 1/e)$ of the optimal solution [15]. Problem (17) can be solved exactly in polynomial time.

Proposition 2. *Algorithm 3 has the worst-case time complexity of*

$$O\left(N_{ag} n_C (n_x + n_\theta)^2 + N_{ag}^2 \log(N_{ag})\right),$$

where $n_C := |\bigcup C|$.

Proof. Complexity of steps 3–8 is $O(N_{\text{ag}} n_C(n_x + n_\theta)^2)$ (see Proposition 1), and Problem (17) has the worst-case complexity of $O(N_{\text{ag}}^2 \log(N_{\text{ag}}))$ [21]. □

IV. NUMERICAL RESULTS

A. System Model:

For this study, we consider the following dynamical model for bio-mass (e.g., algae) growth in water bodies (see [6], [8], [9] for more details),

$$\dot{x}(t) = a \frac{u(t) - x(t)}{u(t)} x(t) + \sum_{i=1}^{2} b_i \frac{\partial^2 x(t)}{\partial z_i^2}. \tag{18}$$

where $x(t)$ denotes the bio-mass density at an arbitrary location, a, b_i are uncertain parameters, $u(t)$ is a known signal that accounts for effects due to external disturbances, and z_i is the i^{th} spatial dimension.

We note that (18) is a simplified representation of bio-mass growth in water bodies. Nonetheless, it captures fundamental characteristics of bio-mass growth. The rate of change of bio-mass density is a spatio-temporal function of the current density. The first term on right hand side of (18) resembles the logistic population growth whose carrying capacity is limited by $u(t)$. The carrying capacity or the maximum sustainable bio-mass density in water bodies is affected by many external factors including available amount of nutrients, ambient lighting conditions and temperatures, amount of precipitation, etc. In favor of a simple model, we have lumped these external effects into the signal $u(t)$ which is assumed to be known.

The second term in (18) accounts for the diffusion of bio-mass from high-density locations to low-density locations. We assume a simple case in which the density varies over two-dimensional spatial domain (e.g., the water surface), and ignore the third spatial dimension. We note that, in reality, the gravity plays an important role in diffusion of mass density,

especially, in deep water bodies and should be accounted for in realistic models.

We discretize (18) over a uniformly spaced two-dimensional $n_{\text{grid}} \times n_{\text{grid}}$ grid or cells-matrix. We stack the columns of this cells-matrix which results in a discretized state vector $x_k \in \mathbb{R}^{n_x}$, $n_x = n_{\text{grid}}^2$, whose elements correspond to densities in different cells and its evolution can be written in the form (1). The diffusion terms i.e. the second derivatives are approximated using finite differences. All variables, parameters and signals are assumed to be non-dimensional, and we sample the temporal evolution such that $\Delta t = t_{k+1} - t_k = 1$. The external signal $u(t)$ and parameters $\theta := [a, b_1, b_2] \in \mathbb{R}^3$ are assumed to identical for all grid cells.

B. Sensors:

For this case study, the set of all available sensors $\mathcal{S}$ corresponds to the all possible locations or grid cells where bio-mass density can be measured. Thus, there are total $|\mathcal{S}| = n_x$ sensing locations. In (2), $c_i \in \mathbb{R}^{n_x}$ is a standard basis vector whose i^{th} element is unity. In other words, the i^{th} sensor measures bio-mass density at the cell corresponding to i^{th} element of x_k. Mobile agents are assumed to equipped with onboard instrumentation to measure densities and transmit the measurement data to a centralized planner.

C. Results:

Case: Single Agent: We consider a case where the (18) is discretized in a 3×3 grid, i.e., $n_{\text{grid}} = 3, n_x = 9$. We assume there is a single mobile agent that can travel one step (left, right, up or down) between two successive measurement instances, i.e., $N_{\text{ag}} = 1, N_{\text{steps}} = 1$. Following covariance values were used in the implementation of the filter: $Q_k = 0.1I$, $Q_k^\theta = 0.05I$, $R_k = 0.01I$, and $\tilde{P}_0 = 0.2I$. Values of the parameters were set to $a^* = 0.2, b_1^* = 0.05, b_2^* = 0.05$ as a reference for simulating sensor measurements. The external signal $u(t)$ was set to $u(t_k) = 15 + 5\sin(t_k)$.

The objective is to monitor the discretized domain using the single agent, i.e., to identify the next *optimal* cell location for measuring the bio-mass density. The state trajectories estimated using the Algorithm 2 and a random policy are shown in Fig. 1, and the sensing schedule, i.e., path traversed by the agent in first 10 steps is shown in Fig. 2. Since Algorithm 2 identifies one-step locally optimal sensor location in each iteration, we note that for this case where $N_{\text{ag}} = 1$, the greedy approach yields the globally optimal solution.

Fig. 1 highlights the efficacy of an optimal sensor schedule as it aims to minimize the overall uncertainty across all states at each step. On the other hand, the state uncertainty with a random schedule - which ignores any uncertainty measure - can be large for some states. For example, consider the cells x_7 and x_8 which were not measured in first 10 steps of the random schedule. As seen in Fig. 1, the uncertainty in these two cells is quite large around $t_k = 10$ for a random schedule, whereas the greedy schedule makes measurements of these cells to minimize the overall uncertainty. The worst case uncertainty in state and parameter estimates quantified

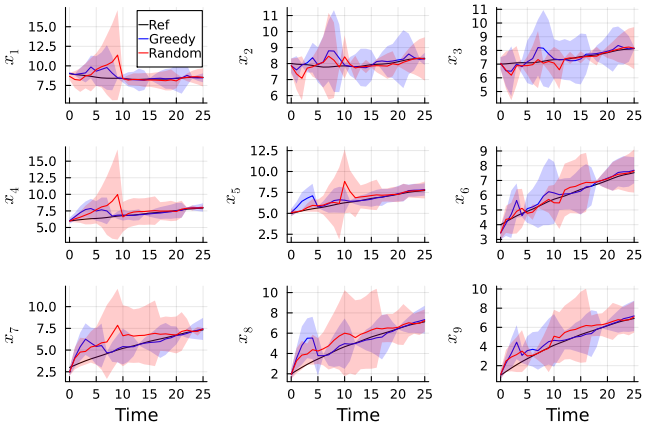

Fig. 1: Estimated bio-mass density in different cells. Shaded area shows the 1σ confidence interval.

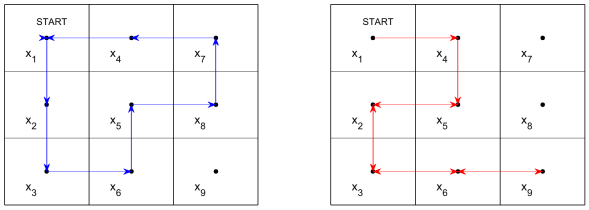

Fig. 2: Path traversed by the mobile agent in first 10 steps using greedy approach (left) and random policy (right) starting at the top left corner of the grid.

by $\det(\tilde{P}_k^+)$ across all time steps was found to be 8.6×10^{-4} for the greedy, and 4.1×10^{-2} for the random schedule.

Case: Multiple Agents: We consider the case where the (18) is discretized in a 5×5 grid, i.e., $n_{\text{grid}} = 5, n_x = 25$ and there are $N_{\text{ag}} = 3$ mobile agents. It is assumed that the agents can travel a maximum of $N_{\text{steps}} = 5$ between two successive measurements. All other parameter values were set identical to the previous case. In order to study performance of the proposed algorithm, we consider two additional sensor scheduling approaches that are variants of the Algorithm 2 with some randomizations. In the first approach, referred to as *Greedy-Rand*, we modify Step 9 of the Algorithm 2 to randomly select an agent from the set $\mathcal{N}$ instead of choosing

Fig. 3: Boxplot of the cumulative sum $\sum_k \text{logdet}((P_k^+)^{-1})$ (normalized by values of Algorithm 2).

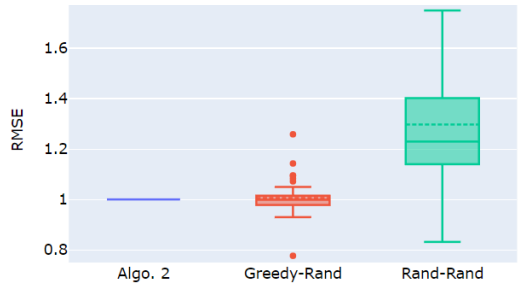

Fig. 4: Boxplot of estimation RMSE (normalized by values of Algorithm 2).

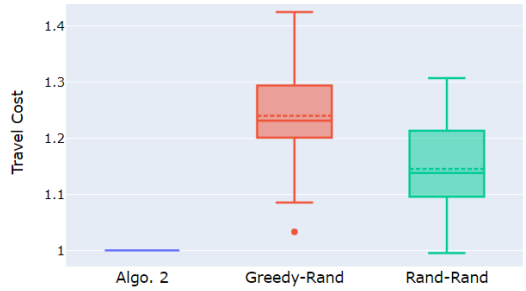

Fig. 5: Boxplot of travel cost (normalized by values of Algorithm 2).

one of the agents with smallest travel costs. In the second approach, referred to as *Rand-Rand*, we modify both Steps 5 and 9 of the Algorithm 2 to randomly select a sensor location from $\mathcal{S}_k^{\text{avail}}$ and then assign an agent randomly. Only sensor and/or agent selection steps are modified in both Greedy-Rand and Rand-Rand approaches while other steps are identical to Algorithm 2. Therefore, both Greedy-Rand and Rand-Rand approaches yield a solution with a dynamically feasible assignment. We compared the performance of different approaches using 50 Monte-Carlo runs of EKF wherein each filter run was initialized with random agent locations and state initial conditions; and the initializations were identical for different approaches in the same run. Each filter trajectory consisted of 25 times steps. Fig. 3, Fig. 4, and Fig. 5 show boxplots of various performance metrics for the three approaches under consideration; and their values were normalized by the values of Algorithm 2 in each filter run. While the median is showed by solid lines in boxplots, the dashed lines denote the mean.

Fig. 3 shows $\sum_k \mathsf{logdet}((P_k^+)^{-1})$, the cumulative sum in a filter run of the objective function J'_{est} that we aimed to maximize. Fig. 4 shows the root mean square error (RMSE) of filter estimates calculated with respect to reference trajectories. While the estimation performances of Algorithm 2 and Greedy-Rand were found to be comparable, Rand-Rand approach demonstrated relatively poorer performance especially in terms of the estimation RMSE. Thus, Fig. 3 and Fig. 4 underscore the efficacy of a greedy approach for an optimal sensor selection. Fig. 5 shows the cumulative total travel cost for all agents in a filter run. Algorithm 2 demonstrated the smallest travel cost as it assigns one of

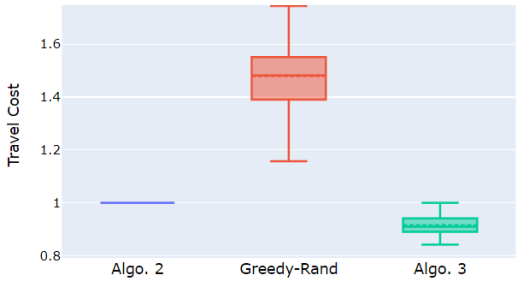

Fig. 6: Boxplot of travel cost (normalized by values of Algorithm 2, $N_{\text{steps}} = \infty$).

the closest agents to the selected sensor location in every iteration of the algorithm. On the other hand, both Greedy-Rand and Rand-Rand rely on random agent assignments, thus, incurred significantly higher travel costs.

We consider a special case discussed in Section III-C. In particular, we assume that an agent can reach any sensing location in the domain from its current location, i.e., $N_{\text{steps}} = \infty$. In such a scenario, coverage sets of different agents are identical and equal to $\mathcal{S}$. We compared the performance of Algorithm 2, Greedy-Rand, and Algorithm 3 for this case. All three approaches demonstrated identical estimation performance (not shown), since they all select identical sensor sets at each measurement instance. On the other hand, travel costs for these approaches were found to be significantly different and are shown in Fig. 6. Algorithm 3, which optimally assigns agents *after* sensors have already been selected, incurred the smallest travel cost. Approximately 10% higher travel cost incurred by Algorithm 2 can be attributed to potentially sub-optimal (nearest) agent assignment it makes simultaneously with the greedy sensor selection in each iteration. While Algorithm 3 yields a feasible one-to-one assignment solution under specific conditions, e.g., identical coverage sets, Algorithm 2 is guaranteed to provide a feasible solution for arbitrary coverage sets.

Scalability: Fig. 7 and Fig. 8 show scalability results for estimation performance and computational cost of Algorithm 2. Fig. 7 illustrates the mean cumulative sum $\sum_k \mathsf{logdet}((P_k^+)^{-1})$, averaged over 50 Monte Carlo EKF runs with random initializations. Increasing the number of allowable steps N_{steps} generally improves estimation performance, particularly when the number of agents N_{ag} is small, due to greater flexibility in sensor selection. However, this benefit diminishes as N_{ag} increases. Estimation performance improves significantly with more agents, as this enables richer data collection from the environment. Nonetheless, the marginal gains in performance decrease with each additional agent.

Fig. 8 presents the average computation time of Algorithm 2 per filtering step. As expected, the computational time increases with both the number of grid cells and the number of agents, due to the growing complexity of the sensor selection problem. The algorithm maintains compute times within a few seconds even for large grid sizes and a modest number ($\leq$ 10) of agents – a configuration rep-

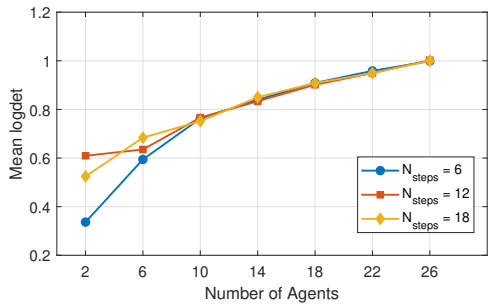

Fig. 7: Variation of the mean cumulative sum $\sum_k \text{logdet}((P_k^+)^{-1})$ with respect to the number of agents, for fixed $n_x = 100$ and different values of N_{steps}. All values are normalized by the 'best case' with $N_{\text{ag}} = 26$ and $N_{\text{steps}} = 18$.

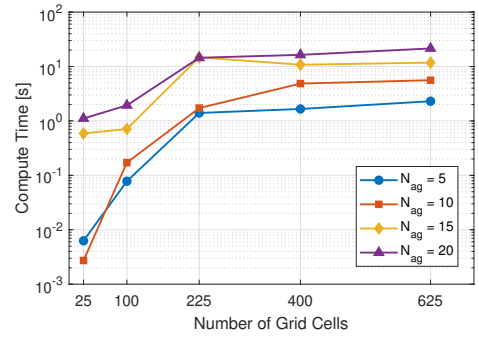

Fig. 8: Average computation time of Algorithm 2 as a function of the number of grid cells, for a fixed $N_{\text{steps}} = 5$ and varying N_{ag}.

resentative of typical deployment scenarios – demonstrating its practical viability for spatio-temporal monitoring applications.

V. Conclusions

In this paper, we formulated a dynamic sensor scheduling approach for monitoring a class of spatio-temporal environmental processes. We proposed an improved greedy algorithm that optimally selects sensing locations and assigns agents to these locations, minimizing travel costs while satisfying the dynamic reachability constraints. Although performance guarantees for the proposed algorithm were provided under specific conditions, general guarantees remain unestablished and warrant further investigation. Nonetheless, the proposed approach demonstrated promising monitoring performance in simulation results.

References

[1] M. Dunbabin and L. Marques, "Robots for environmental monitoring: Significant advancements and applications," *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 24–39, 2012.

[3] B. Potter, G. Valentino, L. Yates, T. Benzing, and A. Salman, "Environmental monitoring using a drone-enabled wireless sensor network," in *2019 Systems and Information Engineering Design Symposium (SIEDS)*, 2019, pp. 1–6.

[2] J. Burgués and S. Marco, "Environmental chemical sensing using small drones: A review," *Science of The Total Environment*, vol. 748, p. 141172, 2020.

[4] P. Thaker, S. D. Cairano, and A. P. Vinod, "Bandit-based multi-agent search under noisy observations*," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 2780–2785, 2023, 22nd IFAC World Congress. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896323017834

[5] K.-C. Ma, L. Liu, H. K. Heidarsson, and G. S. Sukhatme, "Data-driven learning and planning for environmental sampling," *Journal of Field Robotics*, vol. 35, no. 5, pp. 643–661, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21767

[6] A. B. Janssen, J. H. Janse, A. H. Beusen, M. Chang, J. A. Harrison, I. Huttunen, X. Kong, J. Rost, S. Teurlincx, T. A. Troost, D. van Wijk, and W. M. Mooij, "How to model algal blooms in any lake on earth," *Current Opinion in Environmental Sustainability*, vol. 36, p. 1–10, Feb. 2019. [Online]. Available: http://dx.doi.org/10.1016/j.cosust.2018.09.001

[7] B. A. Schaeffer, S. W. Bailey, R. N. Conmy, M. Galvin, A. R. Ignatius, J. M. Johnston, D. J. Keith, R. S. Lunetta, R. Parmar, R. P. Stumpf, E. A. Urquhart, P. J. Werdell, and K. Wolfe, "Mobile device application for monitoring cyanobacteria harmful algal blooms using sentinel-3 satellite ocean and land colour instruments," *Environmental Modelling & Software*, vol. 109, pp. 93–103, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364815218302482

[8] J. Zhang, J. Shi, and X. Chang, "A model of algal growth depending on nutrients and inorganic carbon in a poorly mixed water column," *Journal of Mathematical Biology*, vol. 83, no. 2, July 2021. [Online]. Available: http://dx.doi.org/10.1007/s00285-021-01640-z

[9] A. Thornton, T. Weinhart, O. Bokhove, B. Zhang, D. Sar, van der, K. Kumar, M. Pisarenco, M. Rudnaya, V. Savcenco, J. Rademacher, J. Zijlstra, A. Szabelska, J. Zyprych, M. Schans, van der, V. Timperio, and F. Veerman, "Modeling and optimization of algae growth," in *Proceedings of the 72nd European Study Group Mathematics with Industry (SWI 2010, Amsterdam, The Netherlands, January 25-29, 2010)*. Centrum voor Wiskunde en Informatica, 2010, pp. 54–85.

[10] Y. Fan, Z. Yang, W. Huai, H. Dai, and Y. Zhai, "Dynamic distribution monitoring and biomass estimation of aquatic vegetation in jupiá hydropower station, brazil," *Journal of Hydrology: Regional Studies*, vol. 51, p. 101606, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214581823002938

[11] V. M. Deshpande and R. Bhattacharya, "Guaranteed robust performance of $\mathcal{H}_\infty$ filters with sparse and low precision sensing," *IEEE Transactions on Automatic Control*, vol. 69, no. 2, pp. 1029–1036, 2024.

[12] M. Shamaiah, S. Banerjee, and H. Vikalo, "Greedy sensor selection: Leveraging submodularity," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 2572–2577.

[13] H. Zhang, R. Ayoub, and S. Sundaram, "Sensor selection for kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms," *Automatica*, vol. 78, pp. 202–210, 2017.

[14] V. M. Deshpande, R. Bhattacharya, and K. Subbarao, "Sensor placement with optimal precision for temperature estimation of battery systems," *IEEE Control Systems Letters*, vol. 6, pp. 1082–1087, 2022.

[15] G. L. Nemhauser and L. A. Wolsey, "Best Algorithms for Approximating the Maximum of a Submodular Set Function," *Mathematics of Operations Research*, vol. 3, no. 3, pp. 177–188, 1978, publisher: INFORMS.

[16] J. L. Crassidis and J. L. Junkins, *Optimal Estimation of Dynamic Systems*. Chapman and Hall/CRC, 2011.

[17] S. M. LaValle, *Planning Algorithms*. Cambridge, UK: Cambridge University Press, 2006.

[18] S. Nayak, M. Greiff, A. Raghunathan, S. Di Cairano, and A. P. Vinod, "Data-driven monitoring with mobile sensors and charging stations using multi-arm bandits and coordinated motion planners," in *2024 American Control Conference (ACC)*. IEEE, 2024, pp. 299–305.

[19] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[20] R. Burkard, M. Dell'Amico, and S. Martello, *Assignment Problems*. USA: Society for Industrial and Applied Mathematics, 2009.

[21] M. L. Fredman and R. E. Tarjan, "Fibonacci heaps and their uses in improved network optimization algorithms," *J. ACM*, vol. 34, no. 3, p. 596–615, July 1987. [Online]. Available: https://doi.org/10.1145/28869.28874