2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Madrid, Spain, October 1-5, 2018

Resilient Active Information Gathering with Mobile Robots

Brent Schlotfeldt,[1] Vasileios Tzoumas,[2] Dinesh Thakur,[1] George J. Pappas[1]

***Abstract*—Applications of safety, security, and rescue in robotics, such as multi-robot target tracking, involve the execution of information acquisition tasks by teams of mobile robots. However, in failure-prone or adversarial environments, robots get attacked, their communication channels get jammed, and their sensors may fail, resulting in the withdrawal of robots from the collective task, and consequently the inability of the remaining active robots to coordinate with each other. As a result, traditional design paradigms become insufficient and, in contrast, *resilient* designs against system-wide *failures and attacks* become important. In general, resilient design problems are hard, and even though they often involve objective functions that are monotone or submodular, scalable approximation algorithms for their solution have been hitherto unknown. In this paper, we provide the first algorithm, enabling the following capabilities: *minimal communication*, i.e., the algorithm is executed by the robots based only on minimal communication between them; *system-wide resiliency*, i.e., the algorithm is valid for any number of denial-of-service attacks and failures; and *provable approximation performance*, i.e., the algorithm ensures for all monotone (and not necessarily submodular) objective functions a solution that is finitely close to the optimal. We quantify our algorithms approximation performance using a notion of curvature for monotone set functions. We support our theoretical analyses with simulated and real-world experiments, by considering an active information gathering scenario, namely, *multi-robot target tracking*.**

I. Introduction

Advances in robotic miniaturization, perception, and communication [1]–[7] envision the deployment of robots to support critical missions of *safety*, *security*, and *rescue* such as:

- *Hazardous environmental monitoring*: Deploy a team of mobile robots *to monitor* the radiation flow around a nuclear reactor after an explosion; [1]
- *Adversarial-target tracking*: Deploy a team of agile robots *to track* an adversarial target that moves in a cluttered urban environment, aiming to escape; [2]
- *Search and rescue*: Deploy a team of aerial micro-robots *to localize* people trapped in a burning building; [3]

Each of the above scenarios requires the deployment of a team of mobile robots, where each robot needs to be agile; coordinate its motion with its team in a decentralized way; and navigate itself in unknown, complex, and GPS-denied environments, with the objective of gathering the most information about a process of interest. In particular, the problem of designing the motion of a team of mobile robots to infer the state of a process is known as *active information gathering*.

But in all above mission scenarios the robots operate in failure-prone and adversarial environments, where the robots' can get attacked; their communications channels can get jammed; or their sensors can fail. Therefore, in such failure-prone or adversarial scenarios, *resilient* planning against *worst-case* and *system-wide failures and attacks* become important.

In this paper we formalize for the first time a problem of *resilient active information gathering* with mobile robots, that goes beyond the traditional objective of (non-resilient) active information gathering, and guards against worst-case failures or attacks that can cause not only the withdrawal of robots from the information gathering task, but also the inability of the remaining robots to jointly optimize their control inputs, due to disruptions to their communication network.

Resilient active information gathering with mobile robots is a computationally challenging task, since it needs to account for all possible removals of robots from the joint motion-design task, which is a problem of combinatorial complexity. In particular, this computational challenge motivates one of the primary goals in this paper, namely, to provide a scalable and provably near-optimal *approximation* algorithm for resilient active information gathering with mobile robots.

Related work. Related work on problems of information gathering focuses on the deployment of either static sensors [8]–[10] or mobile sensors (mounted on robots) [11]–[20] to monitor a target process. Among these works, the line of work [11]–[20] is the most relevant to ours, as it considers mobile sensors. In particular, [11]–[15] focus on information gathering tasks over non-Gaussian processes, whereas the remaining [16]–[20] focus on information gathering tasks over Gaussian processes. The advantage in the latter case is that open-loop robot-motion designs are optimal [18], an observation that led [18]–[20] to provide the first scalable, non-myopic robot-motion algorithms for active information gathering, along with sub-optimality guarantees. However, in all of these works, there is no resilience to failures or attacks.

In contrast to robotic motion planning, resilient optimization problems have recently received attention in the literature of set function optimization [21]–[23]. However, [21]–[23] focus on the resilient selection of a *small subset* of elements in the event of attacks or failures, whereas the information acquisition problem requires the selection of controls for *all* robots over a time horizon. In this paper, we adapt the recent results in [22], [23] and seek to bridge the gap between developments in set function optimization and robotic motion planning algorithms in order to enable critical information gathering tasks executed by a team of mobile robots.

[1]The authors are with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (email: {brentsc, pappasg}@seas.upenn.edu).

[2]At the time the paper was written, the author was with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104-6228 USA. Currently, the author is with the Laboratory for Information & Decision Systems (LIDS), and the Department of Aeronautics and Astronautics (AeroAstro), Massachusetts Institute of Technology, Cambridge, MA 02139 USA (email: vtzoumas@mit.edu).

This research is partially supported by ARL CRA DCIST W911NF-17-2-0181 and the Rockefeller Foundation.

Contributions. We make the following contributions:

- (*Problem definition*) We formalize the problem of *resilient active information gathering with mobile robots* against multi-robot denial-of-service attacks or failures. This is the first work to formalize, address, and demonstrate the importance of this problem.
- (*Solution*) We develop the first algorithm for resilient active information gathering with the following properties:
  - *minimal communication*: it terminates within the same order of communication rounds as state-of-the-art algorithms for (non-resilient) information gathering;
  - *system-wide resiliency*: it is valid for any number of denial-of-service attacks or failures;
  - *provable approximation performance*: for all monotone (not necessarily submodular) information gathering objective functions in the active robot set (non-failed robots), it ensures a solution close to the optimal.
- (*Simulations*) We conduct simulations in a variety of multi-robot multi-target tracking scenarios, varying the number of robots, targets, and failures. Our simulations validate the benefits of our approach to achieve resilient target tracking against failures or attacks.
- (*Experiments*) We conduct hardware experiments of multiple quad-rotors tracking static ground targets, to visually demonstrate the necessity of resilient robot motion planning against robotic failures or denial-of-service attacks.

All proofs of our theoretical contributions can be found in the full version of the paper, located at the authors' websites.

Notation. Calligraphic fonts denote sets (e.g., $\mathcal{A}$). Given a set $\mathcal{A}$, then $|\mathcal{A}|$ denotes $\mathcal{A}$'s cardinality; given also a set $\mathcal{B}$, then $\mathcal{A}\setminus\mathcal{B}$ denotes the set of elements in $\mathcal{A}$ that are not in $\mathcal{B}$. Given a random variable v, with mean μ and covariance Σ, then $v \sim \mathsf{N}(\mu, \Sigma)$ denotes that v is a Gaussian random variable.

II. Problem Statement

We formalize the problem of resilient active information gathering. We begin with some basic definitions.

A. Basic definitions

We introduce standard models for the notions of *robots*, *target*, *sensors*, and *information objective function* [18].

a) Robots: Active information gathering utilizes a team of mobile robots to track the evolution of a target process. We denote the set of available robots as $\mathcal{V}$, and model each robot's dynamics as a discrete-time non-linear system:

$$x_{i,t} = f_i(x_{i,t-1}, u_{i,t-1}), \quad i \in \mathcal{V}, \quad t = 1, 2, \dots, \quad (1)$$

where the vector $x_{i,t} \in \mathbb{R}^{n_{x_{i,t}}}$ represents the state of robot i at time t, and the vector $u_{i,t} \in \mathcal{U}_{i,t}$ represents the control input, where $\mathcal{U}_{i,t}$ is a *finite* set of admissible control inputs.

b) Target: The objective of active information gathering is to track the evolution of a target process. We model the target's evolution as a standard discrete-time (possibly time-varying) linear system with additive process noise:

$$y_t = A_{t-1} y_{t-1} + w_{t-1}, \quad t = 1, 2, \dots, \quad (2)$$

where the vector $y_t \in \mathbb{R}^{n_{y_t}}$ represents the state of the target at time t, the vector $w_{t-1} \in \mathbb{R}^{n_{y_t}}$ represents process noise, and the matrix A_{t-1} has suitable dimension. In addition, we let y_0 be a random variable with covariance $\Sigma_{0|0}$, and w_{t-1} be a random variable with zero mean and covariance W_{t-1} such that w_{t-1} is independent of y_0 and of $w_{t'-1}$, for all $t' \neq t$.

c) Sensor measurements: We consider the sensor measurements to be linearly dependent on the state of the target,[1] and non-linearly dependent on the robots' state, as follows:

$$z_{i,t} = H_{i,t}(x_{i,t}) y_t + v_{i,t}(x_{i,t}), \quad i \in \mathcal{V}, \quad t = 1, 2, \dots, \quad (3)$$

where the vector $z_{i,t} \in \mathbb{R}^{n_{z_{i,t}}}$ is the measurement obtained at time t by the on-board sensor at robot i, the vector $v_{i,t}(x_{i,t}) \in \mathbb{R}^{n_{z_{i,t}}}$ represents measurement noise, and the matrix $H_{i,t}(x_{i,t})$ has suitable dimension. In addition, we let $v_{i,t}(x_{i,t})$ be a random variable with zero mean and covariance $v_{i,t}(x_{i,t})$ such that $v_{i,t}(x_{i,t})$ is independent of y_0, of $w_{t'-1}$, and of $v_{i',t'}$ for all $t' \neq t$, and $i' \neq i$.

d) Information objective function: The problem of active information gathering requires the team of robots in $\mathcal{V}$ to select their control inputs to maximize the team's tracking capability of a target. To the latter end, we assume the robots use a Kalman filtering algorithm to track the evolution of the target over a time-horizon T. Moreover, we consider the robots' collective tracking capability to be quantified by an information objective function, denoted henceforth by J, that depends *solely* on the Kalman filter's error covariances across all times $t = 1, 2, \dots, T$. Naturally, the Kalman filter's error covariances depend on the robots' control inputs, as well as on both the target process's initial condition y_0 and the robots' initial conditions $\{x_{i,0} : i \in \mathcal{V}\}$. Overall, given a time-horizon T, we define the objective function,

$$J(u_{1:T}(\mathcal{V})) \triangleq J[\Sigma_1(u_1(\mathcal{V})), \dots, \Sigma_T(u_{1:T}(\mathcal{V}))], \quad (4)$$

where $\Sigma_t(u_{1:t}(\mathcal{V}))$ denotes the Kalman filter's error covariance at time t given the robots' control inputs up to time t, namely, given $u_{1:t}(\mathcal{V}) \triangleq \{u_{i,t'} : u_{i,t'} \in \mathcal{U}_{i,t'}, \ i \in \mathcal{V}, \ t' = 1, 2, \dots, t\}$. Examples of information objective functions of the same form as in eq. (4) are the average minimum mean square error $1/T \sum_{t=1}^{T} \mathrm{trace}\,(\Sigma_t)$, the average confidence-ellipsoid volume $1/T \sum_{t=1}^{T} \det(\Sigma_t)$ [24, Appendix E], as well as information theoretic objectives such as the mutual information $I(y_t|z_{1:t})$ and conditional entropy $h(y_t|z_{1:t})$ [18], where $z_{1:t} \triangleq \{z_{i,t'} : \ i \in \mathcal{V}, \ \ t' = 1, 2, \dots, t\}$, i.e., $z_{1:t}$ is the set of measurements collected by all robots' across all times.

B. Resilient Active Information Gathering

We now define the main problem in this paper.

Problem 1 (Resilient Active Information Gathering) *Given a time horizon T, consider a set of robots $\mathcal{V}$, with dynamics per eq.* (1)*, with sensing capabilities per eq.* (3)*, and with a connected communication network. In addition, consider a target process per eq.* (2)*. Moreover, consider*

[1] This standard modeling consideration is without loss of generality whenever linearization over the current estimate of the target's state is possible.

an information gathering objective function J per eq. (4). *Finally, consider a number $\alpha \leq |\mathcal{V}|$.*

For all robots $i \in \mathcal{V}$, and for all times $t = 1, \ldots, T$, choose control inputs $u_{i,t}$ to maximize the objective function J against a worst-case *failure or attack that causes the removal of α robots from $\mathcal{V}$ at the beginning of time ($t = 0$), and the disruption of all communications among the remaining robots in $\mathcal{V}$ across all times ($t = 1, \ldots, T$). Formally,*

$$\begin{aligned} \max_{\substack{u_{i,t} \in \mathcal{U}_{i,t},\, i \in \mathcal{V}, \\ t = 1, 2 \ldots, T}} \; \min_{\mathcal{A} \subseteq \mathcal{V}} \quad & J(u_{1:T}(\mathcal{V} \setminus \mathcal{A})) : \\ \text{such that, for all } i \in \mathcal{V}, \quad & t = 1, 2, \ldots, T : \\ & y_t = A_{t-1} y_{t-1} + w_{t-1}, \\ & x_{i,t} = f_i(x_{i,t-1}, u_{i,t-1}), \\ & z_{i,t} = H_{i,t}(x_{i,t}) y_{i,t} + v_{i,t}(x_{i,t}), \\ & u_{i,t} = u_{i,t}(z_{i,1}, z_{i,2}, \ldots, z_{i,t}), \\ & |\mathcal{A}| \leq \alpha, \end{aligned} \tag{5}$$

where for any robot set $\mathcal{R} \subseteq \mathcal{V}$ and any time horizon T, we let $u_{1:T}(\mathcal{R}) \triangleq \{u_{i,t} : \; u_{i,t} \in \mathcal{U}_{i,t}, \;\; i \in \mathcal{R}, \;\; t = 1, 2, \ldots, T\}$.

We henceforth denote the problem in eq. (5) by

$$\mathsf{P}(\mathcal{V}, \alpha), \tag{6}$$

where we stress the dependence of the problem only on the set of robots $\mathcal{V}$, and the maximum number of failures or attacks α. Evidently, the (non-resilient) active information gathering problem is an instance of the problem in eq. (6) where $\alpha = 0$. Hence, Problem 1 goes beyond the objective of the (non-resilient) information gathering problem $\mathsf{P}(\mathcal{V}, 0)$, by accounting in the planning process for worst-case failures or attacks that not only may cause the removal of robots from the information gathering task, but also they may prevent the remaining robots from jointly re-planning their motion, e.g., due to the caused disruptions to the robots' communication network after the removal of the attacked or failed robots.

III. Algorithm for Resilient Active Information gathering

We present the first scalable algorithm for Problem 1 (Algorithm 1), followed by the intuition behind it.

A. Scalable algorithm for Problem 1

Algorithm 1 is composed of four steps:

a) Computation of robots' marginal contributions in the absence of attacks (step 1 of Algorithm 1): Each robot $i \in \mathcal{V}$ solves the problem of active information gathering in eq. (7), which is an instance of Problem 1 where no other robot participates in the information gathering task, and where no failures are possible; algorithms to solve such information gathering problems have been proposed in [18]–[20].

b) Computation of robot set $\mathcal{L}$ with the α largest marginal contributions in the absence of attacks (step 2 of Algorithm 1): The robots in $\mathcal{V}$ share their marginal contribution to the information gathering task, which they computed in Algorithm 1's step 1, and decide which subset $\mathcal{L}$ of them composes a set of α robots with the α largest marginal contributions; this procedure can be executed with minimal communication (at most $2|\mathcal{V}|$ communication rounds), e.g., by accumulating (through the communication network) to one robot all the marginal contributions $\{q_i : i \in \mathcal{V}\}$, and, then, by letting this robot to select the set $\mathcal{L}$, and to communicate it back to the rest of the robots.

c) Computation of control inputs for robots in $\mathcal{L}$ (step 3 of Algorithm 1): The robots in the set $\mathcal{L}$, per Algorithm 1's step 2, adopt the control inputs they computed in Algorithm 1's step 1 (e.g., using the algorithm in [18]).

d) Computation of control inputs for robots in $\mathcal{V} \setminus \mathcal{L}$ (step 4 of Algorithm 1): Given the set of robots $\mathcal{L}$, per Algorithm 1's line 2, the remaining robots in $\mathcal{V} \setminus \mathcal{L}$ jointly solve the problem of active information gathering in eq. (8), which is an instance of Problem 1 where the robots in $\mathcal{L}$ do not participate in the information gathering task, and where any attacks or failures are impossible. In particular, the robots in $\mathcal{V} \setminus \mathcal{L}$ can jointly solve the problem in eq. (8) with minimal communication (at most $2|\mathcal{V}|$ communication rounds) using the algorithm *coordinate descent* [20, Section IV].

Algorithm 1 Scalable algorithm for Problem 1.

Input: Time horizon T; set of robots $\mathcal{V}$; dynamics of robots in $\mathcal{V}$, per eq. (1); dynamics of target process, per eq. (2); sensing capabilities of robots in $\mathcal{V}$, per eq. (3); information objective function J, per eq. (4); number $\alpha \leq |\mathcal{V}|$, per Problem 1, that represents the maximum number of possible robot removals from $\mathcal{V}$.

Output: Control inputs $u_{i,t}$ for all robots $i \in \mathcal{V}$, and for all times $t = 1, 2, \ldots, T$.

1: Each robot $i \in \mathcal{V}$ computes the value of the (non-resilient) active information gathering problem

$$\mathsf{P}(\{i\}, 0), \tag{7}$$

per the notation in eq. (6), and denotes it by q_i.

2: All robots in $\mathcal{V}$ find a subset $\mathcal{L}$ of α robots among them (that is, $\mathcal{L} \subseteq \mathcal{V}$ and $|\mathcal{L}| = \alpha$), such that for all robots $i \in \mathcal{L}$ and all robots $j \in \mathcal{V} \setminus \mathcal{L}$, it is $q_i > q_j$;

3: Each robot in $\mathcal{L}$ adopts the control inputs it computed in Algorithm 1's line 1 by solving the problem in eq. (7).

4: The robots in $\mathcal{V} \setminus \mathcal{L}$ compute their control inputs by solving the following active information gathering problem

$$\mathsf{P}(\mathcal{V} \setminus \mathcal{L}, 0), \tag{8}$$

per the notation we introduced in eq. (6).

B. Intuition behind Algorithm 1

The goal of Problem 1 is to ensure the success of an information gathering task despite failures or attacks that cause the removal of α robots from the task, and, consequently, disruptions to the robot's communication network (due to the robots' previous removal), which prevent the remaining robots from jointly re-planning their motion. In this context,

Algorithm 1 aims to fulfill Problem 1's goal first by separating the set of robots $\mathcal{V}$ into two subsets —the set of robots $\mathcal{L}$, and the (remaining) set of robots $\mathcal{V} \setminus \mathcal{L}$ (Algorithm 1's line 1 and line 2),— and second by designing the robots' control inputs in each of the two sets (Algorithm 1's line 3 and line 4). In particular, Algorithm 1 aims with set $\mathcal{L}$ to capture the worst-case attack or failure to α robots among the robots in $\mathcal{V}$; equivalently, the set $\mathcal{L}$ is aimed to act as a "bait" to an attacker that selects the *best* α robots in $\mathcal{V}$ (*best* with respect to the robots' contribution towards attaining the goal of Problem 1). However, the problem of selecting the *best* α robots in $\mathcal{V}$ is a combinatorial problem, and, in general, intractable [25]. Therefore, Algorithm 1 aims to approximate the best α robots in $\mathcal{V}$ by letting the set $\mathcal{L}$ be the set of α robots with the α largest marginal contributions, and, then, it assigns to them the corresponding control inputs (Algorithm 1's line 2 and line 3). Afterwards, given the set $\mathcal{L}$, Algorithm 1 assumes the removal of the robots in $\mathcal{L}$ from $\mathcal{V}$, and coordinates the remaining robots in $\mathcal{V} \setminus \mathcal{L}$ to jointly plan their motion using a decentralized active information gathering algorithm, such as the coordinated descent algorithm proposed in [20, Section IV] (Algorithm 1's line 4).

IV. Performance Guarantees

We quantify Algorithm 1's performance, by bounding the number of robot communication rounds it requires, as well as, by bounding its approximation performance. To this end, we use the following two notions of curvature for set functions.[2]

A. Curvature and total curvature of monotone functions

We present the notions of *curvature* and of *total curvature* for non-decreasing set functions. We start with the notions of *monotonicity*, and of *submodularity* for set functions.

Definition 1 (Monotonicity [26]) *Consider any finite set $\mathcal{V}$. The set function $g : 2^{\mathcal{V}} \mapsto \mathbb{R}$ is* non-decreasing *if and only if for any sets $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{V}$, it holds $g(\mathcal{B}) \geq g(\mathcal{A})$.*

Definition 2 (Submodularity [26, Proposition 2.1]) *Consider any finite set $\mathcal{V}$. The set function $g : 2^{\mathcal{V}} \mapsto \mathbb{R}$ is* submodular *if and only if for any sets $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{V}$, and any element $v \in \mathcal{V}$, it holds $g(\mathcal{A} \cup \{v\}) - g(\mathcal{A}) \geq g(\mathcal{B} \cup \{v\}) - g(\mathcal{B})$.*

In words, a set function g is submodular if and only if it satisfies a diminishing returns property where for any $\mathcal{A} \subseteq \mathcal{V}$ and $v \in \mathcal{V}$, the drop $g(\mathcal{A} \cup \{v\}) - g(\mathcal{A})$ is non-increasing.

Definition 3 (Curvature of monotone submodular functions [27]) *Consider a finite set $\mathcal{V}$ and a non-decreasing submodular set function $g : 2^{\mathcal{V}} \mapsto \mathbb{R}$ such that (without loss of generality) for any elements $v \in \mathcal{V}$, it is $g(v) \neq 0$. The curvature of g is defined as follows:*

$$\kappa_g \triangleq 1 - \min_{v \in \mathcal{V}} \frac{g(\mathcal{V}) - g(\mathcal{V} \setminus \{v\})}{g(v)}. \tag{9}$$

[2]We focus on properties of set functions to quantify Algorithm 1's approximation performance by analyzing the properties of Problem 1's objective function J as a function of the remaining robot set after the removal of a subset of robots from $\mathcal{V}$ (due to failures or attacks).

Notably, the above notion of curvature implies that for any non-decreasing submodular set function g, it is $0 \leq \kappa_g \leq 1$.

Definition 4 (Total curvature of non-decreasing functions [28, Section 8]) *Consider a finite set $\mathcal{V}$ and a monotone set function $g : 2^{\mathcal{V}} \mapsto \mathbb{R}$. The total curvature of g is defined as follows:*

$$c_g \triangleq 1 - \min_{v \in \mathcal{V}} \min_{\mathcal{A}, \mathcal{B} \subseteq \mathcal{V} \setminus \{v\}} \frac{g(\{v\} \cup \mathcal{A}) - g(\mathcal{A})}{g(\{v\} \cup \mathcal{B}) - g(\mathcal{B})}. \tag{10}$$

The above notion of total curvature implies that for any non-decreasing set function g, it is $0 \leq c_g \leq 1$. Moreover, to connect the notion of total curvature with that of curvature, we note that when a function g is non-decreasing and submodular, then the two notions coincide, i.e., $c_g = \kappa_g$.

B. Performance analysis for Algorithm 1

We quantify Algorithm 1's approximation performance, as well as, the number of communication rounds it requires.

Theorem 1 (Performance of Algorithm 1) *Consider an instance of Problem 1, and the definitions:*

- *let the number $J^\star$ be the (optimal) value to Problem 1, i.e., $J^\star \triangleq \mathsf{P}(\mathcal{V}, \alpha)$;*
- *given any control inputs $u_{1:T}(\mathcal{V})$ for the robots in $\mathcal{V}$, let the set $\mathcal{A}^\star$ be a (worst-case) removal of α robots from $\mathcal{V}$, i.e., $\mathcal{A}^\star \triangleq \arg\min_{\mathcal{A} \subseteq \mathcal{V}} J[u_{1:T}(\mathcal{V} \setminus \mathcal{A})]$;*
- *given any removal of a subset of robots $\mathcal{A}$ from the robot set $\mathcal{V}$ (due to attacks or failures), call the remaining robot set $\mathcal{V} \setminus \mathcal{A}$* active robot set.

Finally, consider the robots in $\mathcal{V}$ to solve optimally the problems in Algorithm 1's step 1 and step 4, using an algorithm that terminates in ρ communication rounds.

1) (Approximation performance) *Algorithm 1 returns control inputs $u_{1:T}(\mathcal{V})$ such that:*

- *If the objective function J is non-decreasing and submodular in the active robot set, and (without loss of generality) J is non-negative and $J[u_{1:T}(\emptyset)] = 0$, then, it is*

$$\frac{J[u_{1:T}(\mathcal{V} \setminus \mathcal{A}^\star)]}{J^\star} \geq \max\left(1 - \kappa_J, \frac{1}{1+\alpha}\right), \tag{11}$$

  where κ_J is the curvature of J (Definition 3).
- *If the objective function J is non-decreasing in the active robot set, and (without loss of generality) J is non-negative and $J[u_{1:T}(\emptyset)] = 0$, then, it is*

$$\frac{J[u_{1:T}(\mathcal{V} \setminus \mathcal{A}^\star)]}{J^\star} \geq (1 - c_J)^2, \tag{12}$$

  where c_J is the total curvature of J (Definition 4).

2) (Communication rounds) *Algorithm 1 terminates in at most $2|\mathcal{V}| + \rho$ communication rounds.*

Theorem 1 implies on Algorithm 1's performance:

a) Near-optimality: For both monotone submodular and merely monotone information objective functions, Algorithm 1 guarantees a value for Problem 1 which is finitely close to the

optimal. For example, per ineq. (11), the approximation factor of Algorithm 1 is bounded by $1/(1+\alpha)$, which, for any finite number of robots $|\mathcal{V}|$, is non-zero.

b) Approximation difficulty: For both monotone submodular and merely monotone information objective functions, when the curvature κ_J or the total curvature c_J, respectively, tend to zero, Algorithm 1 becomes exact since for $\kappa_J \to 0$ and $c_J \to 0$ the terms $1-\kappa_J$ and $1-c_J$ in ineq. (11) and ineq. (12) tend to 1. Overall, Algorithm 1's curvature-dependent approximation bounds make a first step towards separating the classes of monotone submodular and merely monotone information objective functions into functions for which Problem 1 can be approximated well (low curvature functions), and functions for which it cannot (high curvature functions).

Overall, Theorem 1 quantifies Algorithm 1's approximation performance when the robots in $\mathcal{V}$ solve optimally the problems in Algorithm 1's step 1 and step 4. However, the problems in Algorithm 1's step 1 and step 4 are computationally challenging, and only approximation algorithms are known for their solution, among which the recently proposed *coordinate descent* [20, Section IV]; in particular, coordinate descent has the advantages of being scalable and of having provable approximation performance. We next quantify Algorithm 1's performance when the robots in $\mathcal{V}$ solve the problem in Algorithm 1's step 4 using *coordinate descent* (we refer the reader to Appendix A for a description of coordinate descent).

Proposition 1 *Consider an instance of Problem 1, and the notation introduced in Theorem 1. Finally, consider the robots in $\mathcal{V}$ to solve the problem in Algorithm 1's step 1 optimally, and the problem in Algorithm 1's step 4 using* coordinate descent *[20, Section IV].*

1) (Approximation performance) *Algorithm 1 returns control inputs $u_{1:T}(\mathcal{V})$ such that:*

- *If the objective function J is non-decreasing and submodular in the active robot set, and (without loss of generality) J is non-negative and $J[u_{1:T}(\emptyset)] = 0$, then, it is*

$$\frac{J[u_{1:T}(\mathcal{V}\setminus\mathcal{A}^\star)]}{J^\star} \geq \frac{\max\left(1-\kappa_J, 1/(1+\alpha)\right)}{2}. \quad (13)$$

- *If the objective function J is non-decreasing in the active robot set, and (without loss of generality) J is non-negative and $J[u_{1:T}(\emptyset)] = 0$, then, it is*

$$\frac{J[u_{1:T}(\mathcal{V}\setminus\mathcal{A}^\star)]}{J^\star} \geq \frac{(1-c_J)^3}{2}. \quad (14)$$

2) (Communication rounds) *Algorithm 1 terminates in at most $3|\mathcal{V}|$ communication rounds.*

Proposition 1 implies on Algorithm 1's performance:

a) Approximation performance for low curvature: For both monotone submodular and merely monotone information objective functions, when the curvature κ_J or the total curvature c_J, respectively, tend to zero, Algorithm 1 recovers the same approximation performance as that of the state-of-the-art algorithms for (non-resilient) active information gathering, which Algorithm 1 calls as subroutines. For instance, for submodular objective functions, the algorithm for active information gathering *coordinate descent* [20, Section IV] has approximation performance at least $1/2$ the optimal [20, Theorem 2]. And per Proposition 1, when Algorithm 1 calls this algorithm as a subroutine, it yields approximation performance at least $(1-\kappa_J)/2$ the optimal, which tends to $1/2$ for $\kappa_J \to 0$.

b) Approximation performance for no failures or attacks: For submodular objective functions, and for zero number of failures ($\alpha = 0$), Algorithm 1's approximation performance becomes the same as that of the state-of-the-art algorithms for (non-resilient) active information gathering. In particular, for submodular objective functions, the *coordinate descent* algorithm has approximation performance at least $1/2$ the optimal; at the same time Algorithm 1 also has approximation performance at least $1/2$ the optimal for $\alpha = 0$ (per ineq. (13)).

c) Minimal communication: Even though Algorithm 1 goes beyond the objective of (non-resilient) active information gathering by accounting for attacks or failures, it terminates within the same order of communication rounds as state-of-the-art algorithms for (non-resilient) active information gathering. In particular, the coordinate descent algorithm for active information gathering terminates in at most $|\mathcal{V}|$ rounds, and, per Proposition 1, when Algorithm 1 calls coordinate descent as a subroutine, it terminates in at most $3|\mathcal{V}|$ rounds; evidently, $|\mathcal{V}|$ and $3|\mathcal{V}|$ have the same order.

Summary of theoretical results. Overall, Algorithm 1 is the first algorithm for Problem 1, and enjoys the following:

- *minimal communication*: Algorithm 1 terminates within the same order of communication rounds as state-of-the-art algorithms for (non-resilient) information gathering;
- *system-wide resiliency*: Algorithm 1 is valid for any number of denial-of-service attacks and failures;
- *provable approximation performance*: Algorithm 1 ensures for all monotone and (possibly) submodular objective functions a solution finitely close to the optimal.

V. Application: Multi-target tracking with mobile robots

We motivate the importance of Problem 1, as well as, demonstrate the performance of Algorithm 1, by considering an application of active information gathering, namely, *multi-target tracking with mobile robots*. In particular, the application's setting is as follows: a team $\mathcal{V}$ of mobile robots is tasked with tracking the position of M moving targets. In more detail, each robot moves according to unicycle dynamics on SE(2), discretized with a sampling period τ, as follows:

$$\begin{pmatrix} x^1_{t+1} \\ x^2_{t+1} \\ \theta_{t+1} \end{pmatrix} = \begin{pmatrix} x^1_t \\ x^2_t \\ \theta_t \end{pmatrix} + \begin{pmatrix} \nu\,\mathrm{sinc}(\frac{\omega\tau}{2})\cos(\theta_t + \frac{\omega\tau}{2}) \\ \nu\,\mathrm{sinc}(\frac{\omega\tau}{2})\sin(\theta_t + \frac{\omega\tau}{2}) \\ \tau\omega \end{pmatrix}.$$

The set of admissible controls is given by the motion primitives $\mathcal{U} \triangleq \{(\nu,\omega) : \nu \in \{1,3\}$ m/s, $\omega \in \{0, \pm 1, \pm 3\}$ rad/s$\}$.

The targets move according to double integrator dynamics, corrupted with additive Gaussian noise. For M targets, their state at time t is $y_t = \left[y_{t,1}^\top y_{t,2}^\top, \ldots, y_{t,M}^\top\right]^\top$ where $y_{t,m}$

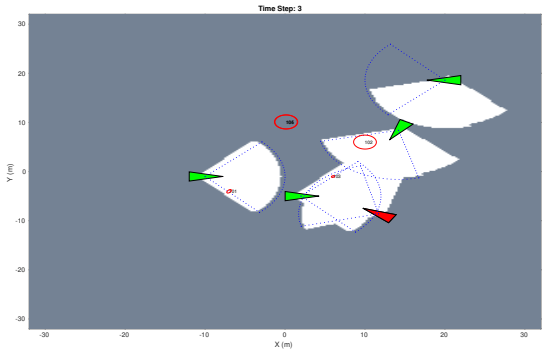


Fig. 1: Simulation environment depicting five robots. The jammed robot is indicated in red.

contains the planar coordinates and velocities of the m-th target, denoted by $(y^1, y^2, \dot{y}^1, \dot{y}^2)$. The model is as follows:

$$y_{t+1,m} = \begin{bmatrix} I_2 & \tau I_2 \\ 0 & I_2 \end{bmatrix} y_{t,m} + w_t, \quad w_t \sim \mathsf{N}\left(0, q \begin{bmatrix} \tau^3/3 I_2 & \tau^2/2 I_2 \\ \tau^2/2 I_2 & \tau I_2 \end{bmatrix}\right).$$

The sensor observation model consists of a range and bearing for each target $m \in \{0, \ldots, M-1\}$:

$$z_{t,m} = h(x_t, y_{t,m}) + v_t, \quad v_t \sim \mathsf{N}\left(0, V(x_t, y_{t,m})\right);$$
$$h(x, y_m) = \begin{bmatrix} r(x, y_m) \\ \alpha(x, y_m) \end{bmatrix} := \begin{bmatrix} \sqrt{(y^1 - x^1)^2 + (y^2 - x^2)^2} \\ \tan^{-1}((y^2 - x^2)(y^1 - x^1)) - \theta \end{bmatrix}.$$

We note that since the sensor observation model is non-linear, we linearize it around the predicted target trajectory $y \neq x$:

$$\nabla_y h(x, y_m) = \frac{1}{r(x, y_m)} \begin{bmatrix} (y^1 - x^1) & (y^2 - x^2) & 0_{1x2} \\ -\sin(\theta + \alpha(x, y_m)) & \cos(\theta + \alpha(x, y_m)) & 0_{1x2} \end{bmatrix}.$$

The observation model for the joint target state can then be expressed as a block diagonal matrix containing the linearized observation models for each target along the diagonal, i.e.,

$$H \triangleq \operatorname{diag}\left(\nabla_{y_1} h(x, y_1), \ldots, \nabla_{y_M} h(x, y_M)\right).$$

The sensor noise covariance grows linearly in range and in bearing, up to σ_r^2, and σ_b^2, where σ_r and σ_b are the standard deviation of the range and the bearing noise, respectively. The model here also includes a limited range and field of view, denoted by the parameters r_{sense} and ψ, respectively.

For the information objective function we use the average log determinant of the covariance matrix, which is equivalent to conditional entropy for Gaussian variables [18]. Notably, this objective function is merely monotone, and not necessarily submodular [29]. Overall, we solve an instance of Problem 1 with the above constraints, and the objective function:

$$J \triangleq \frac{1}{T} \sum_{t=1}^{T} \log\det(\Sigma_t),$$

where Σ_t is the Kalman filtering covariance at time t [18].[3]

[3]We remark that the problem scenario is dependent on a prior distribution of the target's initial conditions y_0 and $\Sigma_{0|0}$. Notwithstanding, if a prior distribution is unknown, an exploration strategy can be incorporated to find the targets by placing exploration landmarks at the map frontiers [20].

Finally, we use the algorithms *ARVI* and *coordinate descent* described in [19] and [20], for step 1 and step 4 of Algorithm 1, respectively.

A. Simulations on multi-target tracking with mobile robots

We use simulations to evaluate the performance of our Algorithm 1 across different scenarios. In particular, we vary the number of robots, n, the number of targets M, and the number of attacks α. In each of these scenarios we compare the performance of the resilient Algorithm 1 with that of the non-resilient algorithm *coordinate descent* [20, Section IV]. To this end, we consider two information performance measures: the average entropy and average root mean square error (RMSE) per target, averaged over the robots in the team.

We describe the parameters of the simulation: the robots and targets in the environment are restricted to move inside a 64x64 meter environment, as in Fig. 1. For the evaluation, we fix the initial positions of both the robots and targets, and the robots are given a prior distribution of the targets before starting the simulation. The targets start with a zero velocity, and in the event that a target leaves the environment its velocity is reflected to remain in bounds. Across all simulations we fix the remaining parameters as follows: $T = 25$ steps, $\tau = 0.5$ s, $r_{sense} = 10$ m, $\psi = 94°$, $\sigma_r = .15$ m, $\sigma_b = 5°$, $q = .001$. Finally, we run Algorithm 1 in a receding horizon fashion, re-planning every 25 time-steps for a total of $T_{max} = 500$ steps, and average the results of each simulation configuration over 10 monte-carlo trials. The simulation was implemented in C++, and runs on a laptop with an Intel Core i7 CPU.

We emphasize that the robots are forced to execute the entire 25-step trajectory without re-planning, due to the jamming attack that occurs at the onset of every planning phase. Thus, the simulation consists of a sequence of instances of Problem 1, where after every 25-steps the network is re-connected, before a new attack occurs, allowing the next problem to begin. This continues until the 500 total time-steps have completed.

Our results are depicted in Fig. 2 and Table I. We observe in Fig. 2 that the performance of the resilient Algorithm 1 is superior both with respect to the average entropy and the RMSE error per target. We observe that as the number of jamming attacks grows, Algorithm 1's benefits becomes more pronounced, as the non-resilient algorithm suffers in performance..

Table I suggests that the resilient Algorithm 1 achieves a lower average error than the non-resilient algorithm, and, crucially, is highly effective in reducing the peak estimation error; in particular, Algorithm 1 achieves a performance that is 2 to 30 times better in comparison to the performance achieved by the non-resilient algorithm. We also observe that the impact of Algorithm 1 is most prominent when the number of attacks is large relative to the size of the robot team.

B. Experiments on multi-target tracking with mobile robots

We implement Algorithm 1 in a multi-UAV scenario with two quadrotors tracking the positions of two static ground targets, shown in Fig. 3. The UAV trajectories are computed

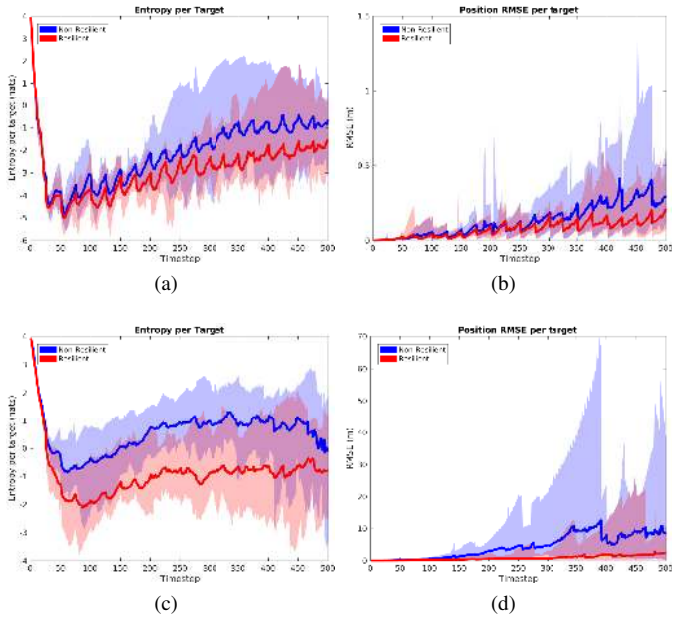


Fig. 2: The figures depict the average entropy and position RMSE error (root mean square error) per target, averaged over the robots. Figs. (a-b) were obtained from a simulation with 10 robots, 10 targets, with 2 jamming attacks. Figs. (c-d) have the same configuration but up to 6 jamming attacks. The blue colors correspond to the non-resilient algorithm, and the red colors correspond to the resilient algorithm. The shaded regions are the spread between the minimum and maximum values of the information measure, and the solid lines are the mean value. The plots are the aggregate of ten trials, each executed over 500 time-steps.

	Mean RMSE (m)		Peak RMSE (m)	
	NR	Resilient	NR	Resilient
$n = 5,\ M = 10$				
$\alpha = 1$	0.28	0.19	9.62	2.09
$\alpha = 2$	1.47	0.68	26.07	15.71
$\alpha = 4$	10.67	4.9	225.47	103.82
$n = 10,\ M = 5$				
$\alpha = 2$	0.35	0.14	57.65	1.87
$\alpha = 4$	0.39	0.28	6.66	3.17
$\alpha = 6$	2.07	0.65	93.27	15.63
$n = 10,\ M = 10$				
$\alpha = 2$	0.13	0.08	1.4	1.32
$\alpha = 4$	0.24	0.23	4.19	2.66
$\alpha = 6$	4.39	1.2	69.77	26.4

TABLE I: The table depicts the estimation performance, measured by average and peak RMSE error per tracked target, for a variety of configurations. The number n denotes the number of mobile sensors, (i.e., $n = |\mathcal{V}|$), M denotes the number of moving targets, and α denotes the number of failures. NR denotes the non-resilient algorithm, while Resilient is Algorithm 1. All results are across 500 timesteps, averaged over ten trials per configuration.

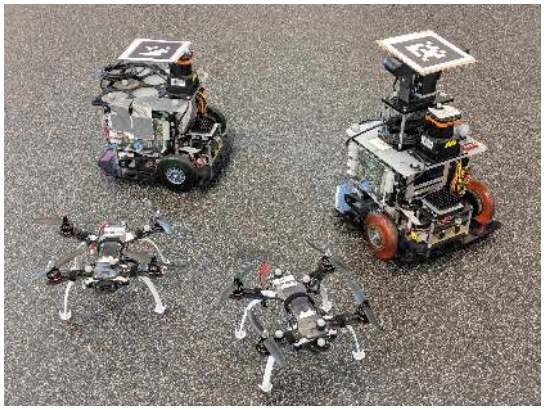


Fig. 3: The experimental setup with two quad-rotors equipped with Qualcomm Flight™, and two Scarabs as ground targets.

off-board but in *real-time* on a laptop with an Intel Core i7 CPU. The UAVs are localized using the Vicon Motion Capture system. The UAVs are quad-rotors equipped with Qualcomm Flight™. The UAVs use Vicon pose estimates to generate noisy measurements corresponding to a downward facing camera which has a 360° field-of-view, and a 1 meter sensing radius. The UAVs move in a 4x8 meter testing laboratory environment with no obstacles. One robot is jammed at all times.

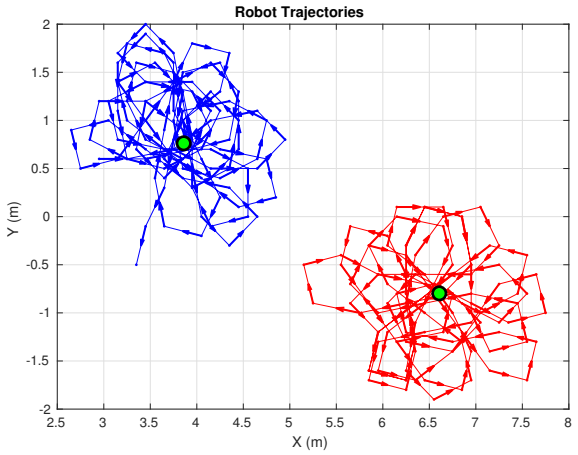
The goal of the hardware experiments is to acquire a visual interpretation of the properties of the trajectories designed using the resilient Algorithm 1. To isolate the effect of resilience, we simplify the problem to static targets (i.e. stationary) and to the smallest possible team, i.e., 2 robots.

We observe from the experiments that the trajectories planned by the UAVs under the non-resilient algorithm stick to the target they are closest to, whereas under the resilient Algorithm 1, the UAVs switch amongst the two targets (Fig. 4). Intuitively, the reason is that the resilient algorithm always assumes that one of the robots will fail, in which case the optimal strategy for one UAV is to track two targets is to switch amongst the targets, whereas the non-resilient algorithm assumes that none of the robots will fail, in which case the optimal strategy for two UAVs is to allocate themselves to the closest target. When there is the possibility of one UAV failing, switching amongst the targets is preferable, since both robots will acquire information about both targets.

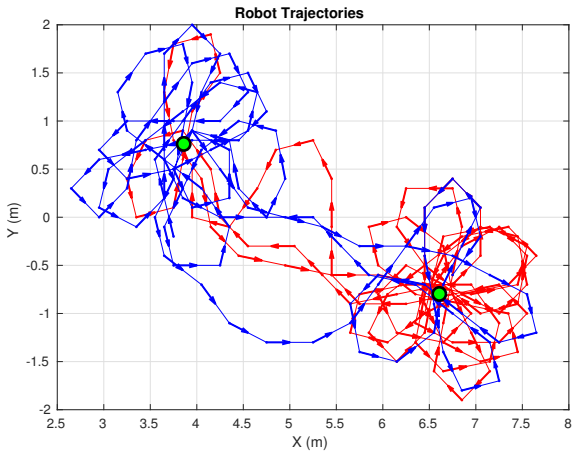
VI. Concluding Remarks & Future Work

We made the first steps to ensure the success of critical active information gathering tasks against failures and denial-of-service attacks, per Problem 1. In particular, we provided the first algorithm for Problem 1, and proved it guarantees near-optimal performance against system-wide failures, even with minimal robot communication. We motivated the need for resilient active information gathering, and showcased the success of our algorithm, with simulated and real-world experiments in a series of multi-robot target tracking scenarios.

This paper opens a number of avenues for future research, both in theory and in applications. Future work in theory includes the resilient design of the robot's communication network against network-wide failures, to balance the trade-off between *minimal communication* and *connectedness*, which is necessitated in scenarios that are both resource constrained



(a)



(b)

Fig. 4: The plot in (a) depicts the experimental robot trajectories in the non-resilient algorithm. The figure in (b) depicts the resilient algorithm. The targets are in green.

(e.g., where bandwidth or battery is limited), and failure-prone (e.g., where attacks can disrupt communication links). Future work in applications includes the experimental testing of resilient active information gathering with mobile robots in environmental monitoring, search and rescue scenarios, and simultaneous localization and mapping.

VII. Acknowledgements

We thank Nikolay A. Atanasov for inspiring discussions.

References

[1] M. Michini, M. A. Hsieh, E. Forgoston, and I. B. Schwartz, "Robotic tracking of coherent structures in flows," *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 593–603, 2014.

[2] C. Nieto-Granda, J. G. Rogers III, and H. Christensen, "Multi-robot exploration strategies for tactical tasks in urban environments," in *SPIE Defense, Security, and Sensing*. International Society for Optics and Photonics, 2013, pp. 87 410B–87 410B.

[3] V. Kumar and N. Michael, "Opportunities and challenges with autonomous micro aerial vehicles," *The International Journal of Robotics Research*, vol. 31, no. 11, pp. 1279–1291, 2012.

[4] S. Karaman and E. Frazzoli, "High-speed flight in an ergodic forest," in *IEEE Intern. Confer. on Robotics and Automation*, 2012, pp. 2899–2906.

[5] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[6] T. Cieslewski, E. Kaufmann, and D. Scaramuzza, "Rapid exploration with multi-rotors: A frontier selection method for high speed flight," in *IEEE/RSJ Int. Conf. on Intel. Robots and Systems*, 2017, pp. 2135–2142.

[7] M. Santos, Y. Diaz-Mercado, and M. Egerstedt, "Coverage control for multirobot teams with heterogeneous sensing capabilities," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 919–925, 2018.

[8] A. Krause, "Optimizing sensing: Theory and applications," Ph.D. dissertation, Carnegie Mellon University, 2008.

[9] J. L. Williams, "Information theoretic sensor management," Ph.D. dissertation, Massachusetts Institute of Technology, 2007.

[10] V. Tzoumas, A. Jadbabaie, and G. J. Pappas, "Near-optimal sensor scheduling for batch state estimation," in *IEEE 55th Conference on Decision and Control*, 2016, pp. 2695–2702.

[11] G. M. Hoffmann and C. J. Tomlin, "Mobile sensor network control using mutual information methods and particle filters," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 32–47, 2010.

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

[13] P. Dames, M. Schwager, V. Kumar, and D. Rus, "A decentralized control policy for adaptive information gathering in hazardous environments," in *IEEE Conference on Decision and Control*, 2012, pp. 2807–2813.

[14] P. Dames and V. Kumar, "Autonomous localization of an unknown number of targets using teams of mobile sensors," *IEEE Trans. on Autom. Science and Eng.*, vol. 12, no. 3, pp. 850–864, 2015.

[15] B. Charrow, V. Kumar, and N. Michael, "Approximate representations for multi-robot control policies that maximize mutual information," *Autonomous Robots*, vol. 37, no. 4, pp. 383–400, 2014.

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

[17] C. M. Kreucher, "An information-based approach to sensor resource allocation," Ph.D. dissertation, University of Michigan, 2005.

[18] N. Atanasov, J. Le Ny, K. Daniilidis, and G. J. Pappas, "Information acquisition with sensing robots: Algorithms and error bounds," in *IEEE International Confer. on Robotics and Automation*, 2014, pp. 6447–6454.

[19] B. Schlotfeldt, D. Thakur, N. Atanasov, V. Kumar, and G. J. Pappas, "Anytime planning for decentralized multi-robot active information gathering," *IEEE Robotics and Automation Letters*, 2018.

[20] N. Atanasov, J. Le Ny, K. Daniilidis, and G. J. Pappas, "Decentralized active information acquisition: Theory and application to multi-robot slam," in *IEEE Int. Conf. on Rob. and Autom.*, 2015, pp. 4775–4782.

[21] J. B. Orlin, A. S. Schulz, and R. Udwani, "Robust monotone submodular function maximization," in *Inter. Conference on Integer Programming and Combinatorial Optimization*, 2016, pp. 312–324.

[22] V. Tzoumas, K. Gatsis, A. Jadbabaie, and G. J. Pappas, "Resilient monotone submodular function maximization," in *IEEE Conference on Decision and Control*, 2017, pp. 1362–1367.

[23] V. Tzoumas, A. Jadbabaie, and G. J. Pappas, "Resilient Monotone Sequential Maximization," *ArXiv e-prints: 1803.07954*.

[24] D. P. Bertsekas, *Dynamic programming and optimal control, Vol. I*. Athena Scientific, 2005.

[25] U. Feige, "A threshold of $ln(n)$ for approximating set cover," *Journal of the ACM*, vol. 45, no. 4, pp. 634–652, 1998.

[26] G. Nemhauser, L. Wolsey, and M. Fisher, "An analysis of approximations for maximizing submodular set functions – I," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[27] M. Conforti and G. Cornuéjols, "Submodular set functions, matroids and the greedy algorithm," *Discrete Applied Mathematics*, vol. 7, no. 3, pp. 251 – 274, 1984.

[28] M. Sviridenko, J. Vondrák, and J. Ward, "Optimal approximation for submodular and supermodular optimization with bounded curvature," *Math. of Operations Research*, vol. 42, no. 4, pp. 1197–1218, 2017.

[29] S. T. Jawaid and S. L. Smith, "Submodularity and greedy algorithms in sensor scheduling for linear dynamical systems," *Automatica*, vol. 61, pp. 282–288, 2015.