Distributed Submodular Maximization on Partition Matroids for Planning on Large Sensor Networks

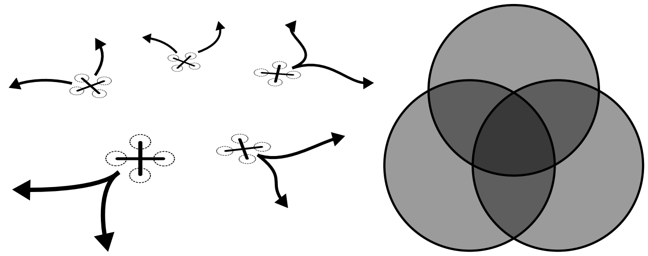
Micah Corah and Nathan Michael

Abstract— **Many problems that are relevant to sensor networks such as active sensing and coverage planning have objectives that exhibit diminishing returns and specifically are submodular. When each agent selects an action local space of actions, sequential maximization techniques for submodular function maximization obtain solutions within half of optimal even though such problems are often NP-Hard. However, adapting methods for submodular function maximization to distributed computation on sensor networks is challenging as sequential execution of planning steps is time-consuming and inefficient. Further, prior works have found that planners suffer severely impaired worst-case performance whenever large numbers of agents plan in parallel. This work develops new tools for analysis of submodular maximization problems which we apply to design of randomized distributed planners that provide constant-factor suboptimality approaching that of standard sequential planners. These bounds apply when the objective satisfies a higher-order monotonicity condition and when cumulative interactions between agents are proportional to the optimal objective value. Problems including generalizations of sensor coverage satisfy these conditions when agents have spatially local sensing actions and limited sensor range. We present simulation results for two such cases.**

I. Introduction

Many objective functions that arise in sensor planning problems such as mutual information [1, 2], objectives for sensing in hazardous environments [3], and various notions of area, set, and sensor coverage [4] are submodular. Intuitively, submodularity implies diminishing returns when constructing sets of sensing actions. This work explores multi-agent planning problems with submodular objective functions and especially variants of set and sensor coverage. We focus on settings involving networks of large or even unspecified numbers of agents seeking to maximize a global submodular objective. Proposed planners perform well in scenarios where agents have limited sensing range and have access to spatially local sets of sensing actions.

Sequential planning strategies can extend generic optimal and suboptimal local planners to the multi-agent domain while retaining suboptimality guarantees in similar scenarios [1–3, 5]. However, sequential planning scales poorly when increasing the numbers of agents while the local planning subproblems themselves may be expensive on their own. For example, active sensing problems often feature nearly infinite spaces of trajectories [1, 2]. At the same time, dynamic environments and beliefs motivate real-time planning [2] so that efficient multi-agent planning is critical



(a) Sensing agents **(b)** Area coverage

Fig. 1: (a) Consider a team of robots engaged in a sensing coverage task. Intuitively, choices made by distant agents may be decoupled. We exploited such conditions to enable efficient distributed planning. Coupling is quantified in terms of an inter-agent redundancy that applies to objectives that have a higher-order property related to submodularity. (b) This work proposes novel randomized planners and analyzes these planners for a class of objective functions that includes area coverage objectives.

for these approaches to be effective in practice for more than a few agents. We address this issue by proposing efficient distributed planners that involve fixed numbers of sequential planning steps and approach existing constant-factor performance bounds on average when cumulative interactions between agents are proportional to objective values.

While solving submodular maximization problems[1] exactly is generally hard, sequential algorithms often provide constant-factor suboptimality guarantees [6, 7]. In fact, well-known worst-case suboptimality bound of $1 - 1/e \approx 0.632$ for sequential planning with a cardinality constraint is optimal over polynomial time algorithms for the value oracle model [8] or unless $\mathbf{P} = \mathbf{NP}$ for variations such as set coverage [9]. For matroid constraints—which generalize cardinality constraints and can model joint action spaces in multi-agent problems—sequential planning obtains solutions within $1/2$ of optimal. More recent works [10, 11] propose algorithms that restore a $1-1/e$ bound but are more computationally intensive. There has also been significant interest in distributed algorithms for submodular maximization [12, 13], but these distributed approaches generally operate by apportioning ground set (all actions for all agents) across processors which is intractable for large sensor networks.

Several recent works [2, 14, 15] also address the core challenge of this work: design of parallel variants of sequential planners for multi-agent systems. Gharesifard and Smith [14] define a class of distributed planners based on directed acyclic graphs where agents perform greedy planning steps using only a subset of the decisions made by prior agents.

The authors are affiliated with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA. {micahcorah, nmichael}@cmu.edu

We gratefully acknowledge support from DOE grant DE-EM0004067.

[1] Specifically those that are submodular, monotone, and normalized.

Although they provide worst-case bounds on suboptimality, Grimsman et al. [15] provide tighter results for the same framework. However, both works obtain bounds that are inversely proportional to the maximum number of agents that may plan in parallel. In contrast, Corah and Michael [2] demonstrate that such planners can be effective when it is possible to find sets of decoupled actions while constructing incremental solutions. However, Corah and Michael [2] only provide a post-hoc bound, and propose an algorithm that does not scale to arbitrary numbers of agents as some steps remain fully sequential.

This work continues in the direction of our prior work [2] in seeking to develop efficient distributed planners that exploit problem structure. As discussed in Fig. 1, our approach is inspired by the intuition that distant agents may be decoupled. We model this idea using a concept of inter-agent redundancy which describes how much one agent's marginal gain can decrease as a result of ignoring another agent. Inter-agent redundancy works in concert with a higher-order monotonicity property referred to as supermodularity of conditioning which implies monotonically decreasing redundancies. Together, these properties enable use of pairwise redundancy to bound the effect of ignoring an agent at any step of the planning process which relates total redundancy to suboptimality. The proposed algorithm randomly partitions agents and obtains a constant-factor bound when the optimal solution is proportional to the cumulative pairwise redundancy between all agents. This condition is generally satisfied by problems involving limited sensing range and distributions of agents with bounded density, and the approach further admits features such as local adaptation and limits on communication range. Finally, we prove that a generalized variant of weighted set coverage satisfies supermodularity of conditioning and provide simulation results for two cases, area coverage and a probabilistic detection scenario.

II. Background Discussion

A. Sets and properties of set functions

Consider a set function $f : 2^\Omega \to \mathbb{R}$ where Ω is called the ground set and 2^Ω is its power set. For convenience, we treat set functions as multi-variate functions so that $f(A, B) = f(A \cup B)$ and implicitly convert elements of the ground set to subsets so that $f(x) = f(\{x\})$ whereas lowercase and uppercase variables represent elements and subsets of the ground set respectively. Subscripts will be used for indexing subsets so that $X_{1:i} = \{x_1, \ldots, x_i\} \subseteq X$. The *discrete derivative* of a set function (or the marginal gain) will be written as $f(x|X) = f(x, X) - f(X)$ for $x \in \Omega$ and $X \subseteq \Omega$. A set function is *non-decreasing* if $f(x|X) \geq 0$ and *normalized* if $f(\emptyset) = 0$. Further, f is *submodular* if, for $A \subseteq B \subseteq \Omega$ and $C \subseteq \Omega \setminus B$, then

$$f(C|B) \leq f(C|A) \qquad (1)$$

which states that marginal gains are monotonically decreasing. The negation of such a set function $-f$ is *supermodular* and *non-increasing*.

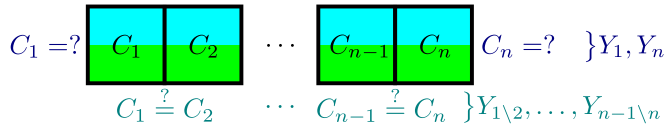


Fig. 2: This illustration depicts an example of a submodular objective where supermodularity of conditioning does not hold and generalizes common examples of when mutual information increases under conditioning [19]. In this example, boxes in the set $C = \{C_1, \ldots, C_n\}$ are colored blue or green independently and with equal probability. Sensors may observe either end directly $\{Y_1, Y_n\}$ or changes in color of adjacent boxes $\{Y_{1\setminus 2}, \ldots Y_{n-1\setminus n}\}$, and obtain a submodular mutual information reward $I(C; Y)$. Each individual observation provides one bit of information, and pairs obtain two bits and no redundancy. This only changes when considering all observations together: the color of the last box can be determined by observing the first and each change in color of subsequent boxes. Observing last box provides no additional information which violates supermodularity of conditioning (2).

This work applies a higher-order monotonicity property which we refer to as *supermodularity of conditioning*. The set function $g_X(Y) = f(X|Y)$ describes how the marginal gain for X varies under conditioning with Y and is referred to as the *conditioning function*. A function exhibits supermodularity of conditioning if g_X is supermodular for all $X \subseteq \Omega$. Specifically, for $A \subseteq B \subseteq \Omega$ and $C \subseteq \Omega \setminus B$, then

$$g_X(C|A) \leq g_X(C|B). \qquad (2)$$

Expressed in terms of f and negated, this expression takes the form $f(X|A) - f(X|A, C) \geq f(X|B) - f(X|B, C)$ which is interpreted as stating that conditioning reduces redundancy as $A \subseteq B$. Expressions of the form $f(A) - f(A|C)$ will be referred to as expressing the pairwise redundancy of A and C. Properties like submodularity of conditioning have not yet been used extensively in the literature on optimization of submodular functions although a few recent works study the same and similar properties [16–18].

Weighted set cover is an example of a submodular function that satisfies supermodularity of conditioning as will be shown later. Weighted set cover objectives have been studied extensively and have hardness results for sequential planning [9] and tightness results for distributed planning [15]. Such results demonstrate that satisfying supermodularity of conditioning does not on its own impact hardness of an optimization problem. At the same time, some relevant objectives do not necessarily exhibit supermodularity of conditioning. Figure 2 describes one such a scenario for a submodular mutual information objective.[2]

B. Independence constraints and partition matroids

Optimization problems involving submodular objectives often incorporate a constraint on admissible subsets of Ω. Consider a constraint represented by the tuple $\mathcal{M} = (\Omega, \mathcal{I})$ where $\mathcal{I}$ is a collection of subsets of Ω such that $X \in \mathcal{I}$ implies $X \subseteq \Omega$. Most commonly considered constraints are special cases of *independence systems* in which $\mathcal{I}$ must be non-empty and satisfy a heredity property so that for all

[2] For this reason, Corollary 2.1 of our prior work [2] is incorrect. One of our contributions is to identify sufficient conditions for an analogous bound.

$X_1 \in \mathcal{I}$ then $X_2 \subseteq X_1$ implies $X_2 \in \mathcal{I}$. An independence system is a *matroid* if $\mathcal{I}$ satisfies an exchange property so that for all $X_1, X_2 \in \mathcal{I}$ such that if $|X_1| > |X_2|$ there exists some $x \in X_1 \setminus X_2$ so that $X_2 \cup \{x\} \in \mathcal{I}$. *Partition matroids* are especially relevant as they describe multi-agent problems where the joint action space is a product of local action spaces. Let $\{\mathcal{X}_1, \ldots, \mathcal{X}_n\}$ partition Ω into blocks $\mathcal{X}_i$ so that $\bigcup_{i=1}^n \mathcal{X}_i = \Omega$ and $\mathcal{X}_i \cap \mathcal{X}_j = \emptyset$ for $i \neq j$. Then $\mathcal{I} = \{X \subseteq \Omega \mid |X \cap \mathcal{X}_i| \leq \ell_i\}$ for $\ell_i \geq 0$ defines a partition matroid.

III. Problem Statement

Consider a multi-agent planning problem with agents $\mathcal{A} = \{1, \ldots, n\}$ where each agent $i \in \mathcal{A}$ is associated with a set of actions $\mathcal{X}_i$ which is also a block of the partition matroid $\mathcal{M}$. Agents may select at most one action so that $|X \cap \mathcal{X}_i| \leq 1$ for each joint solution $X \in \mathcal{I}$. Further, the agents are engaged in a sensing task with an objective f that satisfies the conditions outlined in Sec. II-A and so seek to solve

$$X^\star \in \arg\max_{X \in \mathcal{I}} f(X). \tag{3}$$

As shown by Nemhauser et al. [7] the local greedy heuristic obtains an approximate solution $X^g = \{x_1^g, \ldots, x_n^g\}$ by recursively applying a greedy maximization step,

$$x_i^g \in \arg\max_{x \in \mathcal{X}_i} f(x | X_{1:i-1}^g), \tag{4}$$

and obtains the following bound

$$f(X^\star) \leq 2 f(X^g). \tag{5}$$

IV. Greedy Planning on Directed Acyclic Graphs

Applying (4) on a large network of agents is time-consuming as each agent must wait to receive the incremental solution from the previous agents before beginning computation. Gharesifard and Smith [14] propose a related class of planners where agents may ignore the decisions of previous agents according to a directed acyclic graph. Rather than executing the planning step in (4), these planners obtain the solution $X^d = \{x_1^d, \ldots, x_n^d\}$ by evaluating

$$x_i^d \in \arg\max_{x \in \mathcal{X}_i} f(x | X_{\mathcal{N}_i}^d) \tag{6}$$

using incremental solutions from $\mathcal{N}_i \subseteq \{1, \ldots, i-1\}$, the set of in-neighbors of agent i in the directed acyclic graph. This model can be used to design planers where distant agents do not communicate or where subsets of agents execute their planning steps in parallel. Prior works studying such planners examine worst-case behavior for objectives that are submodular and monotonic [14, 15]. These fail to obtain constant-factor suboptimality when given only a fixed number of sequential planning steps but an arbitrary number of agents. Instead, this work examines sufficient conditions for a distributed planner with a fixed number of sequential planning steps to approach the performance of a sequential planner (5) on average. We begin by analyzing (6) based on the redundancy of sensing actions between pairs of agents.

V. Analysis using Inter-Agent Redundancy

The performance of the distributed planner will be analyzed by bounding decreases in marginal gains due to failure to condition on choices by prior agents i.e. $f(x_i^d | X_{\mathcal{N}_i}^d) - f(x_i^d | X_{1:i-1}^d)$. Supermodularity of conditioning enables derivation of bounds on such changes in marginal gains using pairwise redundancies between elements.

Define the *inter-agent redundancy graph* as a weighted, undirected graph $\mathcal{G} = (\mathcal{A}, \mathcal{E}, \mathcal{W})$ with agents as vertices, edges $\mathcal{E} = \{(i,j) \mid i,j \in \mathcal{A}, i \neq j\}$,[3] and weights

$$\mathcal{W}(i,j) = w_{ij} = \max_{x_i \in \mathcal{X}_i,\, x_j \in \mathcal{X}_j} f(x_i) - f(x_i | x_j). \tag{7}$$

This connects the notion of redundancy to the multi-agent planning problem via maximum inter-agent redundancies which are undirected as $f(x_i) - f(x_i|x_j) = f(x_j) - f(x_j|x_i)$.

Decreases in marginal gains can be bounded using the pairwise redundancies from the inter-agent redundancy graph using the following lemma.

Lemma 1 (Pairwise redundancy bound): Consider disjoint subsets $A, B, C \subseteq \Omega$. Then

$$f(A|B) - f(A|B,C) \leq \sum_{c \in C} f(c) - f(c|A). \tag{8}$$

Proof: Lemma 1 follows from the chain rule and supermodularity of conditioning. Given an ordering $A = \{a_i, \ldots, a_{|A|}\}$, construct a telescoping sum $f(A|B) - f(A|B,C) = \sum_{i=1}^{|A|} f(a_i | A_{1:i-1}, B) - f(a_i | A_{1:i-1}, B, C)$. Then, by supermodularity of conditioning (2) $f(A|B) - f(A|B,C) \leq \sum_{a \in A} f(a) - f(a|C)$. Then (8) follows by symmetry of A and C in the expansion of the left-hand side. ■

The total weight of the redundancy graph will characterize suboptimality for our approach. We will refer to a problem defined according to (3) as *α-redundant* for $\alpha > 0$ if

$$\alpha f(X^\star) \geq \sum_{(i,j) \in \mathcal{E}} w_{ij}. \tag{9}$$

Instances of (3) with finite objective values and numbers of agents are all α-redundant for some α although specific values are not guaranteed in general. We will use α-redundancy to absorb additive terms proportional to graph weights into constant-factor multiplicative bounds in terms of α.

A. Analysis of distributed planners using inter-agent redundancy

The inter-agent redundancy graph defined in the previous section can be applied in the analysis of distributed planners (6) using a similar approach as in our previous work [2]. Let $\hat{\mathcal{N}}_i = \{1, \ldots, i-1\} \setminus \mathcal{N}_i$ be the set of preceding agents that are ignored at step i of the assignment process according to (6) so that the set of all deleted edges is $\tilde{\mathcal{E}} = \{(i,j) \mid i \in \mathcal{A},\, j \in \hat{\mathcal{N}}_i\}$. Then the planner suboptimality can be bounded in terms of the weights of these deleted edges.

Theorem 2 (Suboptimality of distributed planning): The suboptimality of a planner obeying (6) can be bounded

[3] Being undirected, (i,j) and (j,i) are the same edge.

using the cumulative weight of deleted edges as

$$f(X^\star) \leq 2f(X^d) + \sum_{(i,j)\in\tilde{\mathcal{E}}} w_{ij}. \tag{10}$$

Proof: Theorem 2 follows from application of pairwise redundancy on the inter-agent redundancy graph to the standard proof technique for sequential maximization,

$$\begin{aligned} f(X^\star) &\leq f(X^\star, X^d) \\ &= f(X^d) + \sum_{i=1}^{n} f(x_i^\star | X^d, X^\star_{1:i-1}) \\ &\leq f(X^d) + \sum_{i=1}^{n} f(x_i^\star | X^d_{\mathcal{N}_i}) \\ &\leq f(X^d) + \sum_{i=1}^{n} f(x_i^d | X^d_{\mathcal{N}_i}) \\ &= 2f(X^d) + \sum_{i=1}^{n} \left(f(x_i^d | X^d_{\mathcal{N}_i}) - f(x_i^d | X^d_{1:i-1}) \right). \end{aligned} \tag{11}$$

The first inequality follows from f being non-decreasing, the second from submodularity, and the third by greedy choice according to (6). The equalities result from telescoping sums and the chain rule. The main result (10) follows from application of (8) and (7) to the sum in (11) and the definition of $\tilde{\mathcal{E}}$. ■

VI. Randomized Distributed Planners

Let us now apply the analysis from the previous section to design of randomized distributed planners. A set of agents can execute planning steps in parallel if no pair of agents in the set shares an edge in the planner model (6). Considering this, we construct distributed planners by partitioning the agents and eliminating edges within blocks of the partition and then bound suboptimality for randomly assigned partitions. Finally, we present conditions for such planners to scale to an arbitrary number of agents and to admit features such as limited communication range.

A. Distributed planning on partitioned agents

Consider a planner where subsets of agents plan in parallel according to a partition $\{D_1, \ldots, D_{n_d}\}$ of agents $\mathcal{A} = \cup_{i=1}^{n_d} D_i$. In such a planner, n_d corresponds to the maximum number of sequential planning steps. Let d_i map each agent i to its block in the partition so that $i \in D_{d_i}$, and let the total ordering of agents respect a partial ordering induced by ordering the blocks of the partition so that $i < j$ implies $d_i \leq d_j$. We construct a planner (6) from the partition and ordering of agents and blocks by eliminating neighbors that share the same block from the complete directed acyclic graph

$$\mathcal{N}_i = \{1, \ldots, i-1\} \setminus D_{d_i}, \quad \hat{\mathcal{N}_i} = \{1, \ldots, i-1\} \cap D_{d_i}. \tag{12}$$

Ideally, the partition would minimize the cumulative weight of edges eliminated in the subgraphs of the blocks. However, that is equivalent to maximizing the weight of edges outside of the subgraphs which is the Max k-Cut problem on the inter-agent redundancy graph. Finding exact solutions is intractable because Max k-Cut is NP-Complete [20]. Therefore, the next section proposes randomized approaches that produce approximate solutions.

B. Planning with random partitions

As observed by Andersson [21], a random partition obtains a trivial $\frac{n_d-1}{n_d}$ by observing that edges are removed uniformly at random. The approach presented here is similar and is presented from the perspective of individual agents.

Consider a distributed planner as defined by (6) where agents share partial solutions with neighbors given a partition of the agents as in (12). Let each agent select its partition index d_i independently and uniformly at random from $\{1, \ldots, k_i\}$ so that $n_d = \max_{i\in\mathcal{A}} k_i$. We consider two policies for selection of k_i based on the weights of the redundancy graph (7) and a per-agent budget for additive suboptimality $\gamma > 0$. For *global adaptive* planners agents $i \in \mathcal{A}$ select from a fixed number of partition indices proportional to the total redundancy so that

$$k_i = n_d = \left\lceil \frac{1}{n\gamma} \sum_{(i,j)\in\mathcal{E}} w_{ij} \right\rceil. \tag{13}$$

With *local adaptive* planners k_i is proportional instead to the cumulative redundancy for that agent which may be large compared to other agents but involves less global knowledge

$$k_i = \left\lceil \frac{1}{2\gamma} \sum_{j\in\mathcal{A}\setminus\{i\}} w_{ij} \right\rceil. \tag{14}$$

Both local and global planners respect the following bound.

Theorem 3 (Suboptimality for random partitions): Given a budget $\gamma > 0$ for per-agent suboptimality and a planner defined according to (6) which partitions agents according to (12) by drawing partition indices d_i uniformly from $\{1, \ldots, k_i\}$ using (13) or (14) suboptimality is bounded in expectation as

$$f(X^\star) \leq 2\mathbb{E}[f(X^d)] + n\gamma. \tag{15}$$

Proof: The expectation of the cumulative weight of deleted edges for either planner is bounded as

$$\begin{aligned} \mathbb{E}\left[\sum_{(i,j)\in\tilde{\mathcal{E}}} w_{ij} \right] &= \frac{1}{2} \sum_{i=1}^{n} \mathbb{E}\left[\sum_{j\in D_{d_i}\setminus\{i\}} w_{ij} \right] \\ &\leq \sum_{i=1}^{n} \frac{1}{2k_i} \sum_{j\in\mathcal{A}\setminus\{i\}} w_{ij} \end{aligned} \tag{16}$$

where the inequality accounts for when $d_j > k_i$. That is, when another agent j selects a partition index $d_j > k_i$ outside of the set considered by i, the corresponding edge cannot be deleted from the perspective of agent i. For global adaptive planners, $k_i = n_d$ for all i and (16) simplifies to $\frac{1}{n_d} \sum_{(i,j)\in\mathcal{E}} w_{ij}$ and holds with equality. Then (15) follows by applying (13) or (14) and substituting into the expectation of (10) over partitions of agents. ∎

Restating in terms of α-redundancy provides a stronger statement that is useful when varying the number of agents.

Corollary 3.1 (Constant factor suboptimality): Problems with fixed α-redundancy satisfy the constant-factor bound

$$\frac{1-\epsilon}{2} f(X^\star) \leq \mathbb{E}[f(X^d)] \tag{17}$$

for $\epsilon > 0$ and a budget of

$$\gamma = \frac{\epsilon}{\alpha n} \sum_{(i,j)\in\mathcal{E}} w_{ij} \tag{18}$$

by substituting (18) into (15), applying (9), and rearranging.

Corollary 3.2 (Fixed n_d for global planning): Given fixed α-redundancy, global adaptive planners (13) provide constant-factor suboptimality for $n_d = \lceil\frac{\alpha}{\epsilon}\rceil$ sequential planning steps which follows by rearranging (18) to match (13).

C. Near-optimality for varying numbers of agents

In this section, we present sufficient conditions to preserve these guarantees when increasing the number of agents. These conditions correspond intuitively to scenarios where agents have access to local actions and where the environment volume and rewards scale with the number of agents.

We say problems (3) with n agents exhibit *β-linear scaling* for $\beta > 0$ if

$$f(X^\star) \geq \beta n \tag{19}$$

which expresses the condition that rewards scale with the number of agents.

In order to express the relationship between inter-agent distances—or the distribution of agents—and inter-agent redundancy, define a function of inter-agent distance $r : \mathbb{R}_+ \to \mathbb{R}_+$ so that

$$w_{ij} \leq r(||\mathbf{p}_i - \mathbf{p}_j||) \tag{20}$$

where $||\cdot||$ is some norm and $\mathbf{p}_i, \mathbf{p}_j \in \mathbb{R}^d$ are appropriately defined agent positions associated with the blocks of the partition matroid. The following theorem identifies sufficient conditions for problems to have finite α on average and in turn to satisfy Theorem 3 and corollaries which implies a constant expected number of sequential steps (n_d) for the global planner design (13) and any number of agents.

Theorem 4 (Finite average redundancy): Consider a class of problems (3) with a distribution of agents in $\mathbb{R}^d$ with finite density at most ρ that satisfies linear scaling (19) and has redundancy bounded in terms of inter-agent distance (20) for fixed β and r. If $\int_0^\infty r(s)s^{d-1}\,\mathrm{d}s$ is finite, the average value of α, interpreted as a random variable, is also finite.

Proof: Let ρ upper bound the marginal density of agents in $\mathbb{R}^d$, and let A_d be the surface area of the unit sphere under the chosen norm. Because the distribution of agents has fixed maximum density ρ, an arbitrarily large numbers of agent is equivalent to a distribution of agents that covers the entire Euclidean space in the limit. By applying (20) and integrating over spheres centered on $\mathbf{p}_i$ for an arbitrarily large environment, the expected redundancy for a given agent is at most $\hat{r} = \rho A_d \int_0^\infty r(s)s^{d-1}\,\mathrm{d}s \geq \mathbb{E}\left[\sum_{j\in\mathcal{A}\setminus\{i\}} w_{ij}\right]$ which is proportional to the integral in Theorem 4. Treating α as a random variable so that (9) is tight and applying (19) we get $\mathbb{E}[\alpha] \leq \mathbb{E}\left[\frac{\sum_{(i,j)\in\mathcal{E}} w_{ij}}{f(X^\star)}\right] \leq \frac{\hat{r}}{\beta}$ which is finite if $\hat{r}$ is also finite given that $\beta > 0$. ∎

D. Limited communication range

Similar analysis can be used to analyze or design limits on communication range. Let $r_{\max}$ be the maximum sensor range so that the set of agents that are in range is $\mathcal{B}_i = \{j \mid r_{\max} > ||\mathbf{p}_i - \mathbf{p}_j||\}$. Then, any existing communications graph can be readily modified by intersecting the set of in-neighbors with the set of agents that are in range to obtain a new set of neighbors $\hat{\mathcal{N}}_i = \mathcal{B}_i \cap \mathcal{N}_i$. By applying (20) to Theorem 2, each agent then incurs at most

$$\sum_{j\in\mathcal{A}\setminus(\{i\}\cup\mathcal{B}_i)} \frac{w_{ij}}{2} \leq \sum_{j\in\mathcal{A}\setminus(\{i\}\cup\mathcal{B}_i)} \frac{r(||x_i - x_j||)}{2} \tag{21}$$

additional suboptimality i.e. increase to γ in (15). Then, as in Sec. VI-C, the expectation over the distribution of agents is upper-bounded by $\rho A_d \int_{r_{\max}}^\infty r(s)s^{d-1}\,\mathrm{d}s$. This reduces the problem of limited communication range to a question of whether the additional suboptimality is acceptable and whether a techniques such as multi-hop communication are necessary to extend the communication range. Note that even though the communication range can be designed to incur arbitrarily little additional suboptimality, limiting the communication range is not sufficient to guarantee a constant number of sequential planning steps as agents that are within range may depend on agents that are out of range and so on, indefinitely.

VII. Probabilistic Coverage Objectives

Although submodular set functions have been studied extensively, set functions with higher-order monotonicity properties such as what we refer to as supermodularity of conditioning (2) have received relatively little interest [16–18]. Before moving on to present simulation results, let us examine one such objective which satisfies the conditions presented in Sec. II-A. The two scenarios that we will study in simulation involve special-cases of this following objective which is a mild extension of weighted set cover.

Consider a general event detection or identification problem with independent, probabilistic failures. We define a set of events $\mathcal{E}$ and let each event $e \in \mathcal{E}$ have value $v_e \geq 0$. Each event $e \in \mathcal{E}$ and element of the ground set $x \in \Omega$ is associated with an independent failure probability $0 \leq p_x^e \leq 1$. The expected value of identified events given a set of sensing actions $X \subseteq \Omega$ is then

$$f(X) = \sum_{e\in\mathcal{E}} \left(\left(1 - \prod_{x\in X} p_x^e \right) v_e \right) \tag{22}$$

TABLE I: Agent and sensor radii

	Area Coverage	Probabilistic Sensing
r_a	0.226	0.247
r_s	0.113	$6.18 \cdot 10^2$

and is equivalent to the well-known weighted set cover objective in the deterministic case ($p^e_x \in \{0, 1\}$). This objective is trivially normalized and non-decreasing by inspecting the product in (22) and is also submodular as for $A \subseteq B \subseteq \Omega$ and $C \subseteq \Omega \setminus B$ then

$$f(C|B) = \sum_{e\in\mathcal{E}} \left(\left(1 - \prod_{c\in C} p^e_c\right) \prod_{b\in B} p^e_b v_e \right) \leq \sum_{e\in\mathcal{E}} \left(\left(1 - \prod_{c\in C} p^e_c\right) \prod_{a\in A} p^e_a v_e \right) = f(C|A). \tag{23}$$

A. Supermodularity of conditioning for probabilistic sensor coverage

Supermodularity of conditioning and even conditions on higher order differences [16] follow by demonstrating that the differences are similar in form to original function.

Theorem 5: Coverage with sensor failure (22) and, by extension, weighted set cover satisfy supermodularity of conditioning.

Proof: The conditioning function is $g_X(Y) = f(X|Y) = \sum_{e\in\mathcal{E}} \left(\left(1 - \prod_{x\in X} p^e_x\right) \prod_{y\in Y\setminus X} p^e_y v_e \right)$. Consider a new set of values $\hat{v}_e = \left(1 - \prod_{x\in X} p^e_x\right) v_e$ and the set function $\hat{f}(Y) = \sum_{e\in\mathcal{E}} \left(\left(1 - \prod_{y\in Y\setminus X} p^e_y\right) \hat{v}_e \right)$ which has the same form as f (setting failure probability to one in $Y \cap X$). The conditioning function can then be rewritten as $g_X(Y) = \sum_{e\in\mathcal{E}} \hat{v}_e - \hat{f}(Y)$. As submodular functions are closed over addition of a constant, g_X is supermodular and f satisfies supermodularity of conditioning. ■

VIII. Results and Discussion

The proposed distributed planning approach is evaluated through two sets of simulation experiments, each using a variant of the objective function analyzed in Sec. VII. The first evaluates the performance of distributed planners that use various numbers of sequential planning steps (agent partition size n_d) in an area coverage task. The second set of experiments evaluates adaptive planning and limits on communication range in a more complex problem with spatially varying rewards and probabilistic sensing.

A. Common parameters of experiment designs

Several aspects of experiment design are kept constant in each scenario. Each scenario is evaluated in 50 random trials[4] and features a large number of agents (50) so that the proposed distributed planners utilize many times fewer sequential planning steps than a standard sequential planner

[4]Unless otherwise specified, each trial features both random planners and scenarios according to their respective designs.

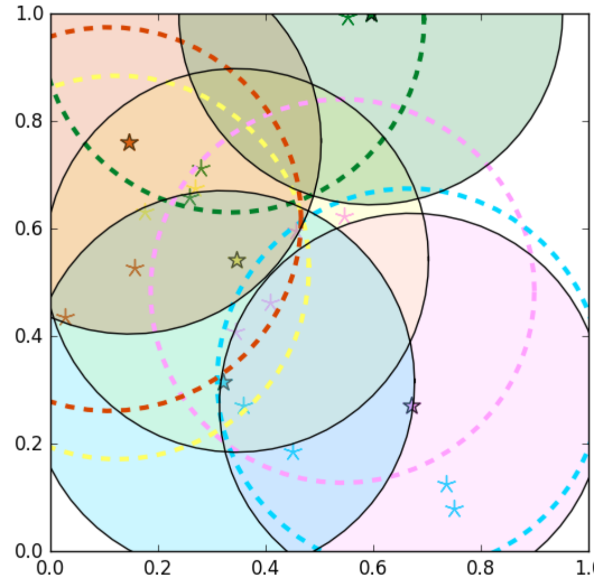


Fig. 3: This figure depicts a maximum area coverage problem and a sequential solution. Agents each have a unique color and are distributed uniformly throughout the environment. Sensing actions ($*$) are in turn distributed uniformly within agent radii (dashed lines). The solution consists of one selected action ($\star$) for each agent; these actions are centered on the translucent disks which make up the covered area.

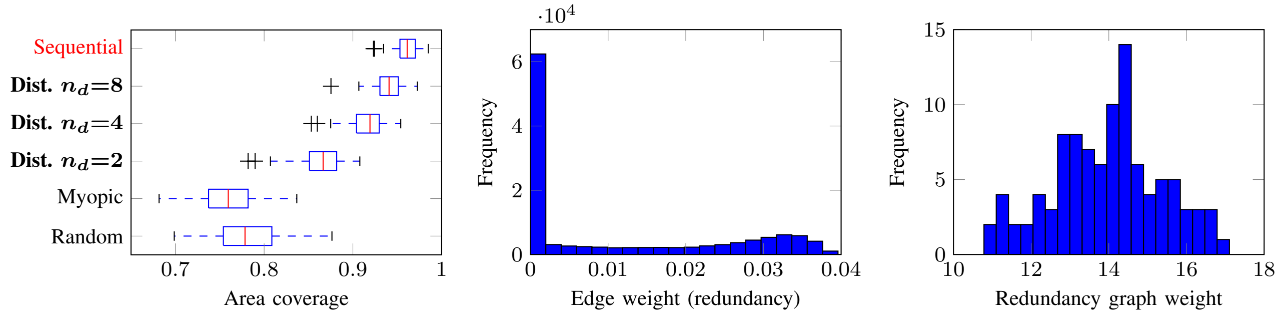
(4). Agent positions are distributed uniformly at random over the unit square, and each agent has a choice of 10 available sensing actions ($\mathcal{X}_i$) which are sampled from a uniform distribution over a disk with radius r_a centered on the agent position. Although the two sets of simulation experiments do not use the same sensor model, each is a function of sensor radius r_s. The sensor and agent radii used in each experiment[5] are listed in Tab. I. In each case, the objective is designed to take on values no greater than one.

B. Area coverage and evaluation of distributed planning

The reward for the area coverage task is the area of the union of discs, each with radius r_s, intersected with the unit square. In terms of the sensor model defined in Sec. VII, this is equivalent to having a failure probability of one outside the disk and zero inside. An example of one simulation trial (using parameters tuned for visualization purposes) is depicted in Fig. 3. The experiments compare distributed planners with fixed partition sizes, $k_i = n_d \in \{2, 4, 8\}$, to sequential planning (4) and to two naive planners: completely random action selection and myopic maximization of the objective over the local space of sensing actions (equivalent to $n_d = 1$). Figure 4 shows the results of these experiments.

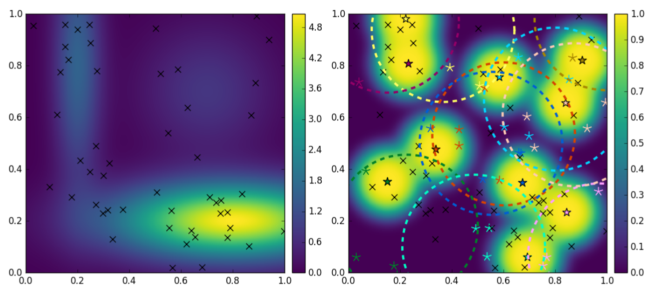
The proposed planner performs well although the performance bounds would no longer be applicable as the deleted edge weight would exceed the maximum possible possible objective value of one as evident in the cumulative weights of the inter-agent redundancy graph. However, the trend

[5]Agent and sensor radii are set according to a normalization over the number of agents and by using parameter search to minimize the ratio of the average performance of myopic and sequential planning to identify hard problem cases.



(a) Objective values **(b)** Edge weight **(c)** Total graph weight

Fig. 4: Results for the area coverage problem (Fig. 3): (a) Objective values for myopic and random planning (no coordination), the **proposed distributed planner** with n_d sequential planning steps, and fully sequential planning (intractable for large numbers of agents). The performance of the proposed distributed planner approaches sequential planning given many times fewer sequential planning steps. (b) Redundancies are computed for each each pair of agents. Sensing actions (disks) for distant agents cannot overlap resulting in many zero weighted edges, and remaining edges are distributed according to varying degrees of overlap in potential sensing actions. (c) The total weight of the redundancy graph is largely between 13 and 17. Even though the planners perform well, our suboptimality bounds would no longer be meaningful as deleted edge weights would exceed maximum objective values.



(a) Scenario **(b)** Planner result

Fig. 5: This figure shows an example of a probabilistic sensing scenario and a sequential solution. Parameters for the scenario are identical to the experimental trials, but the parameters for the agents have been tuned for purpose of visualization. The goal of this task is to maximize the expected number of successful detections or identifications. (a) For each trial, events (x) are sampled from a mixture of Gaussians and are identified correctly with some probability dependent on the sensing actions. (b) Agents, each shown in a different color, are distributed uniformly throughout the environment. Sensing actions ($*$) are distributed uniformly within the agent radius (dashed lines), and each agent selects a single sensing action ($\star$) and successfully identifies events according to a soft-coverage sensing model. The resulting identification probability given selected actions is shown in the background; identification probability is high (yellow) near selected sensing actions and, for good selections, near the events.

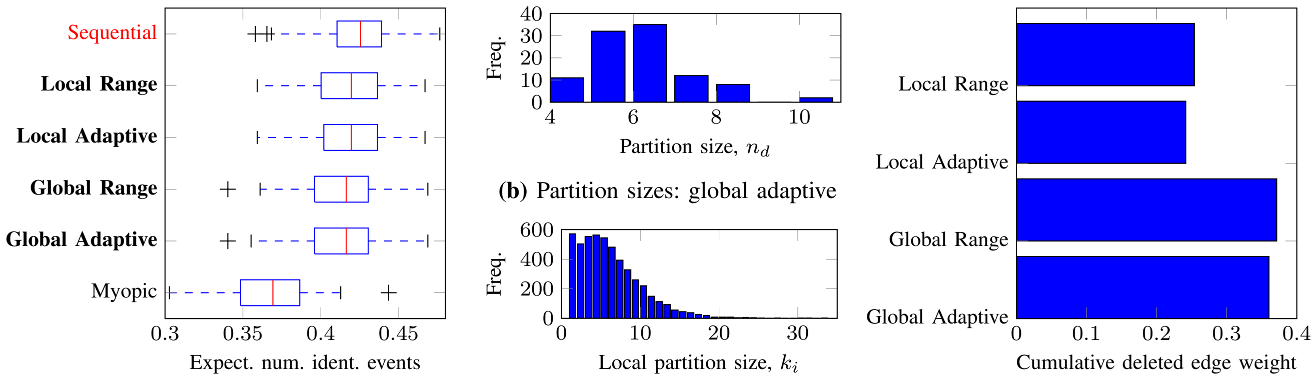
in performance is similar to what would be expected for increasing n_d as the objective values for the distributed planners approach the performance of sequential planning approximately as $1/n_d$. The difference in area coverage compared to sequential planning decreases by approximately half each time n_d is doubled and by 9.9 times from $n_d = 1$ to $n_d = 8$. Overall, the performance of the distributed planner design represents a significant improvement over sequential planning. Given the number of agents, even the greatest value of n_d provides a 6-times improvement in the number of sequential planning steps. Given the scalability analysis in Sec. VI-C, similar performance can be expected for larger problems given similar densities of agents.

C. Adaptive planning with probabilistic sensing and non-uniform events

The goal of the probabilistic sensing task is to maximize the value of correctly identified events (e.g. correct classification of objects moving through the environment). For each trial $n_e = 50$ events, each with value $1/n_e$, are sampled from a fixed Gaussian mixture, rejecting samples outside the unit square. An example is shown in Fig. 5 although using agent parameters more appropriate for visualization purposes but the same Gaussian mixture and number of events. This results in a spatially varying distribution of reward and redundancy. The success probability of the sensor model is e^{-x^2/r_s^4}, where x is the distance from sensor to event location. This success probability effectively amounts to area coverage with soft edges.

This set of experiments evaluates adaptive planning and limited communication range. The budget for deleted edge weight per-agent for the local and global planners is set to $\gamma = 0.4/n = 8 \cdot 10^{-3}$. Communication limited planners are obtained by deleting edges from the respective instances of the distributed planners using a communication range of $r_{\max} = 2r_a$ which allows for a small amount of redundancy due to sensor range r_s. Random planning is not included in this set of experiments because it vastly under-performs myopic planning as the problem design ensures that a large fraction of sensing actions provide little value.

Fig. 6 shows the results of these experiments. The adaptive planners each perform almost identically in terms of distributions of objective values and with values slightly less than for sequential planning. As the objective and actions are highly local, enforcing limits on the communication range has little impact on the planner performance in terms of either objective value or cumulative weight of deleted edges. The global adaptive planner obtains consistent partition sizes by averaging over all agents. In contrast with the local planners agents sometimes select from as many as 33 partitions. Sec. VI-C provides some discussion of mitigation strategies. However, a mix of these strategies is desirable in order to



(a) Objective values

(b) Partition sizes: global adaptive

(c) Partition sizes: local adaptive

(d) Deleted edge weight

Fig. 6: Results for the probabilistic sensing problem (Fig. 5): (a) The proposed **local** and **global adaptive planners** along with their **range-limited counterparts** outperform myopic planning and approach the performance of the fully sequential planner (intractable for large numbers of agents) in terms of objective values. (b) The global adaptive planner uses 4 to 10 partitions in all trials while (c) the local adaptive planner occasionally uses local partitions sizes exceeding 20 for agents with high redundancy. (d) The cumulative weight of deleted edges is on the same order as the objective value and is below the desired limit of 0.4. Range-limited planners are obtained by deleting edges from the associated adaptive planner at the cost of relatively little additional deleted edge weight.

avoid computing averages over all agents.

IX. Conclusion

Efficiently solving submodular maximization problems on sensor networks is challenging due to the inherent sequential structure of common planning strategies. Whereas prior works [14, 15] have shown that worst-case performance degrades rapidly when reducing the number of sequential planning steps, we show that constant-factor performance approaching that of the standard sequential algorithm can be obtained for the proposed randomized planner when cumulative redundancy is at most proportional to the objective values. Toward this end, the inter-agent redundancy graph expresses the degree of coupling between agents in the submodular maximization problem, and functions having supermodularity of conditioning admit performance bounds in terms of this graph structure. The resulting bound is readily applied to design of planners that adapt the numbers of sequential planning steps or have limited range for communication.

An important area for future work is to apply these results to the design of online planners such as for multi-robot active sensing tasks [2]. This will involve increased attention to timing for planning steps such as reasoning about the impact of available planning time on anytime planning performance. Additionally, submodular objectives such as mutual information may not satisfy supermodularity of conditioning. Identifying when objectives functions satisfy supermodularity of conditioning exactly or approximately is central to broad application of the results in this paper. Finally, concepts such as supermodularity of conditioning may be useful in other problems involving submodular objectives, and such applications would be an interesting topic for further study.

References

[1] A. Singh, A. Krause, C. Guestrin, and W. J. Kaiser, "Efficient informative sensing using multiple robots," *J. Artif. Intell. Res.*, vol. 34, pp. 707–755, 2009.

[2] M. Corah and N. Michael, "Efficient online multi-robot exploration via distributed sequential greedy assignment," in *Proc. of Robot.: Sci. and Syst.*, Cambridge, MA, Jul. 2017.

[3] S. Jorgensen, R. H. Chen, M. B. Milam, and M. Pavone, "The matroid team surviving orienteers problem: Constrained routing of heterogeneous teams with risky traversal," in *Proc. of the IEEE/RSJ Intl. Conf. on Intell. Robots and Syst.*, Vancouver, Canada, Sep. 2017.

[4] M. Roberts, D. Dey, A. Truong, S. Sinha, S. Shah, A. Kapoor, P. Hanrahan, and N. Joshi, "Submodular trajectory optimization for aerial 3d scanning," *arXiv preprint arXiv:1705.00703*, 2017.

[5] N. A. Atanasov, J. Le Ny, K. Daniilidis, and G. J. Pappas, "Decentralized active information acquisition: Theory and application to multi-robot SLAM," in *Proc. of the IEEE Intl. Conf. on Robot. and Autom.*, Seattle, WA, May 2015.

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

[7] ——, "An analysis of approximations for maximizing submodular set functions-II," *Polyhedral Combinatorics*, vol. 8, pp. 73–87, 1978.

[8] 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.

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

[10] G. Calinescu, C. Chekuri, M. Pal, and J. Vondrák, "Maximizing a monotone submodular function subject to a matroid constraint," *SIAM J. Comput.*, vol. 40, no. 6, pp. 1740–1766, 2011.

[11] Y. Filmus and J. Ward, "A tight combinatorial algorithm for submodular maximization subject to a matroid constraint," in *Proc. of the IEEE Annu. Symp. Found. Comput. Sci.*, New Brunswick, NJ, Oct. 2012.

[12] B. Mirzasoleiman, R. Sarkar, and A. Krause, "Distributed submodular maximization: Identifying representative elements in massive data," in *Adv. in Neural Inf. Process. Syst.*, Stateline, Nevada, Dec. 2013.

[13] R. d. P. Barbosa, A. Ene, H. L. Nguyen, and J. Ward, "A new framework for distributed submodular maximization," in *Proc. of the IEEE Annu. Symp. Found. Comput. Sci.*, New Brunswick, NJ, Oct. 2016.

[14] B. Gharesifard and S. L. Smith, "Distributed submodular maximization with limited information," *IEEE Trans. Control Netw. Syst.*, 2017.

[15] D. Grimsman, M. S. Ali, P. Hespanha, and J. R. Marden, "Impact of information in greedy submodular maximization," in *Proc. of the IEEE Conf. on Decision and Control*, Melbourne, Australia, Dec. 2017.

[16] W. Chen, Q. Li, X. Shan, X. Sun, and J. Zhang, "Higher order monotonicity and submodularity of influence in social networks: from local to global," *arXiv preprint arXiv:1803.00666*, 2018.

[17] S. Foldes and P. L. Hammer, "Submodularity, supermodularity, and higher-order monotonicities of pseudo-boolean functions," *Mathematics of Operations Research*, vol. 30, no. 2, pp. 453–461, 2005.

[18] S. Ramalingam, C. Russell, P. H. Torr *et al.*, "Efficient minimization of higher order submodular functions using monotonic boolean functions," *Discrete Applied Mathematics*, vol. 220, pp. 1–19, 2017.

[19] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY: John Wiley & Sons, 2012.

[20] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of computer computations*. Springer, 1972, pp. 85–103.

[21] G. Andersson, "An approximation algorithm for Max p-Section," in *Annual Symposium on Theoretical Aspects of Computer Science*, Trier, Germany, 1999.