Proofs and Experiments in Scalable, Near-Optimal Search by Multiple Robots

Geoffrey Hollinger and Sanjiv Singh
Robotics Institute
Carnegie Mellon University, Pittsburgh, PA 15213
Email: gholling@ri.cmu.edu, ssingh@ri.cmu.edu

Abstract— In this paper, we examine the problem of locating a non-adversarial target using multiple robotic searchers. This problem is relevant to many applications in robotics including emergency response and aerial surveillance. Assuming a known environment, this problem becomes one of choosing searcher paths that are most likely to intersect with the path taken by the target. We refer to this as the Multi-robot Efficient Search Path Planning (MESPP) problem. Such path planning problems are NP-hard, and optimal solutions typically scale exponentially in the number of searchers. We present a finite-horizon path enumeration algorithm for solving the MESPP problem that utilizes sequential allocation to achieve linear scalability in the number of searchers. We show that solving the MESPP problem requires the maximization of a nondecreasing, submodular objective function, which directly leads to theoretical guarantees on paths generated by sequential allocation. We also demonstrate how our algorithm can run online to incorporate noisy measurements of the target's position during search. We verify the performance of our algorithm both in simulation and in experiments with a novel radio sensor capable of providing range through walls. Our results show that our linearly scalable MESPP algorithm generates searcher paths competitive with those generated by exponential algorithms.

I. INTRODUCTION

The problem of gaining line-of-sight to a non-adversarial target is one that is relevant to many applications in robotics. Emergency response teams may need to locate a lost first responder or survivor in a disaster scenario. Military operations in urban environments may seek to locate a friendly or neutral target during rendezvous or evacuation. The growing availability of emergency response robots and mechanized infantry necessitates the development of algorithms that can solve these problems autonomously. In such scenarios, noisy information about the target's position may be available. In indoor environments, non-line-of-sight sensors like ultra-wideband ranging radios can provide range to a target through walls. A similar scenario arises during outdoor operations if an air vehicle provides surveillance while ground vehicles search for a target. The major application that has motivated our work is that of using a team of autonomous searchers to locate a lost first responder in a disaster scenario while receiving noisy measurements to his or her location [1].

In the scenarios mentioned above, the target is non-adversarial (i.e. not actively avoiding the searchers). In these cases, we can model the target's motion and plan searcher paths that maximize the probability of finding the target over a time interval. We refer to this problem as the Multi-robot

Efficient Search Path Planning (MESPP) problem because robots must plan paths that *efficiently* find a non-adversarial target. This is in contrast with methods that seek to guarantee capture of an adversarial target. The MESPP problem is related to the Multi-robot Informative Path Planning (MIPP) problem in which robots must plan paths that best observe the environment [2]. These path planning problems maximize an objective (or reward) function that relates to how much information is gained by the robots' paths. Planning paths for multiple robots is an NP-hard problem in general because the number of searchable paths grows exponentially with the number of robots [3]. This motivates the development of approximation algorithms in these domains.

Solving the MESPP problem optimally requires planning in the joint space of searcher paths, which grows exponentially in the number of searchers. This is an example of *explicit co-ordination* during which the searchers explicitly plan for their teammates. Alternatively, if each searcher plans individually and shares information about its path for other searchers to consider, the search space no longer grows. We refer to this scenario as *implicit coordination*. Our algorithm uses sequential allocation, which is an instance of implicit coordination. During sequential allocation, searchers plan in an iterative fashion sharing their paths after planning and modifying their objective functions based on other searchers' shared paths. This algorithm scales linearly in the number of searchers. We present an algorithm using finite-horizon planning and sequential allocation that generates paths competitive with those generated by explicit coordination while using far fewer computational resources. Our algorithm also operates online allowing for the incorporation of measurements.

In this paper, we provide both theoretical and empirical results demonstrating the performance of finite-horizon planning with sequential allocation in the MESPP domain. We prove that MESPP requires the optimization of a nondecreasing, submodular objective function. An objective function is submodular if it follows an intuitive property of diminishing returns. The more areas (nodes) in the environment that the searchers have visited, the less incremental reward is gained. An objective function is nondecreasing if observing additional nodes can only increase the expected reward. The nondecreasing submodularity of the MESPP objective function directly leads to bounds on the performance of sequential allocation. We complement these theoretical results with empirical results

both in simulation and using data from ultra-wideband ranging radio sensors.

The rest of this paper is organized as follows. We first put MESPP into context by discussing prior work in pursuit-evasion and coordinated search (Section II). We then formally define the MESPP problem and show that it requires the optimization of a nondecreasing, submodular objective function (Section III). We present a linearly scalable algorithm for solving MESPP using sequential allocation and finite-horizon path enumeration, and we derive theoretical bounds on the performance of this algorithm (Section IV). We further demonstrate the performance of our algorithm both in simulation and using data from novel ranging sensors (Section V). Finally, we summarize our results and discuss avenues for future work in efficient search and other domains (Section VI).

II. RELATED WORK

Much research in coordinated search makes the adversarial assumption on the target's motion (i.e., assumes that the target is actively avoiding capture). This strong assumption on the target's motion necessitates algorithms that guarantee capture for worst-case target behavior. Algorithms for solving this problem both on graphs [4] and in polygonal environments [5] have been proposed. These methods do not directly estimate the target's position and consequently cannot properly handle uncertainty. They provide no mechanism for incorporating measurements or motion models during search. Thus, these methods provide worst-case solutions even if there is a high likelihood of a best-case scenario.

To better model uncertainty and target motion, an alternative is to use a purely probabilistic formulation of coordinated search problems. Partially Observable Markov Decision Processes (POMDPs) can be used to model both non-adversarial and adversarial coordinated search problems. Solving the POMDP formulation requires planning in the joint space of searcher paths. Roy et al. demonstrated how belief compression can be used to directly solve the POMDP formulation of the adversarial coordinated search problem [6]. Smith developed Heuristic Search Value Iteration (HSVI), a near-optimal POMDP solver capable of solving POMDPs with thousands of states [7]. While these solvers are capable of near-optimally solving small instances of MESPP, the exponential increase in the size of the joint planning space yields several million states for large instances of the MESPP problem. These instances are well outside the scope of even state-of-the-art POMDP solvers.

The algorithms described above plan in the joint space of searcher paths to provide explicit coordination between the searchers. This space grows exponentially with increasing searchers. One popular approach to mitigating the computational demands of joint planning is to hold synthetic auctions between agents. Auction-based methods are designed to inject joint planning into the problem where it is most needed. In other words, robots hold auctions to explicitly coordinate when it is most beneficial. Kalra proposed an auction-based method for sharing plans in domains requiring tight coordination [8]. In her method, robots auction a limited number of joint-space plans for consideration by other robots. Gerkey et al. also developed a parallel stochastic hill-climbing technique that dynamically forms teams, which is closely related to auction-based methods [9]. These techniques reduce the complexity of planning for problems such as MESPP. However, they rely on the overhead of auctions and/or team formation, which can consume considerable communication bandwidth and computation.

The poor scalability of joint planning methods has spurred research in heuristic methods for solving coordinated search problems. Sarmiento et al. proposed a heuristic method for finding stationary targets with multiple robots [10]. Their work is purely heuristic and does not derive optimality bounds on performance. Our proofs using submodularity complement prior work in heuristic search by providing theoretical guarantees.

In this paper, we show that the MESPP problem requires the optimization of a submodular objective function, and this key insight provides optimality bounds on sequential allocation, an algorithm linearly scalable in the number of searchers. Submodularity has been utilized in related domains to provide theoretical guarantees on sequential allocation. Guestrin et al. used submodular set functions to develop algorithms for sensor placement problems in Gaussian Processes [11] and in more general domains [12]. They also extended their algorithms to robust observation selection against sensor failure [13]. These applications deal primarily with placing sensors to monitor information in an environment (e.g. monitoring algae blooms in lakes and temperature in a building). These algorithms do not incorporate moving nodes (searchers) and thus are not suitable for ESPP. Singh et al. developed algorithms for solving the Multi-robot Informative Path Planning (MIPP) problem, which does allow for moving nodes [2]. We extend their theoretical bounds to the MESPP problem. To the best of our knowledge, our work is the first approximation algorithm with theoretical guarantees in the MESPP domain.

III. PROBLEM SETUP

This section formally defines the problem of locating a mobile, non-adversarial target with multiple searchers (the MESPP problem). It also shows that the MESPP problem optimizes a submodular objective function.

To formulate the MESPP problem, we need to describe the environment in which the searchers and target are located. We first divide the environment into convex cells. The convexity of the cells guarantee that a searcher in a given cell will have line-of-sight to a target in the same cell. The searcher's goal is now to move into the same cell as the target. Gaining line-of-sight is relevant to most sensors that a mobile robot would carry including cameras and laser rangefinders. Our method for discretization takes advantage of the inherent characteristics of indoor environments. To discretize an indoor map by hand, label convex hallways and rooms as cells and arbitrarily collapse overlapping sections. Alternatively, a suitable discretization can be found automatically using a convex region finding algorithm (such as Quine-McClusky [14]). Taking into account

the cell adjacency in a discretized map yields an undirected graph that the searchers can traverse. For the rest of this paper we assume that the graph has N convex cells. Figure 1 shows three example floorplans used our experiments. We use the two large floorplans for simulated trials and the smaller floorplan for experiments with ranging radios. The museum floorplan is particularly challenging because it contains many cycles by which the target can avoid line-of-sight contact with the searchers.
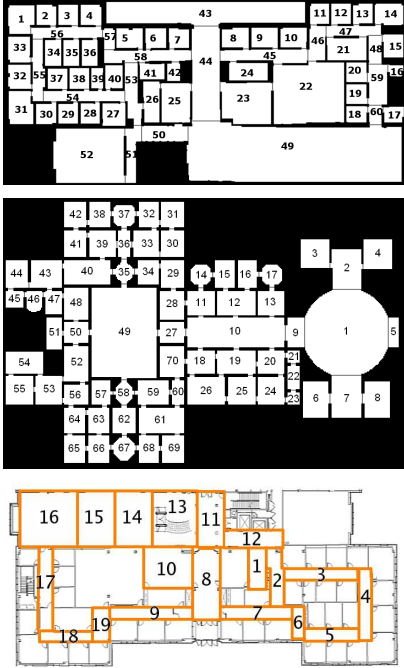

Fig. 1. Example floorplans of environments used for efficient search trials. The larger maps (top and middle) were used for simulated testing. The smaller map (bottom) was used for "hybrid" trials in which simulated searchers found a Pioneer robot with measurements from an experiment.

Let $G(N, E)$ be the undirected environment graph with vertices N and edges E. At any time t, a searcher exists on vertex $s(t) \in N$. The searcher's movement is deterministically controlled, and it may travel to vertex $s(t + 1)$ if there exists an edge between $s(t)$ and $s(t+1)$. A target also exists on this graph on vertex $e(t) \in N$. The target moves probabilistically between vertexes. The searcher receives reward by moving onto the same vertex as the target, $s(t) = e(t)$, and no reward is gained after this occurs. The reward is discounted by γ^t so that the searcher receives more reward for finding the target at a lower t. This discount factor corresponds to the probability that the search will end at a given time. For instance, the target may leave the search area or expire. This necessitates locating the target in a short time.

The target's movement model is known to the searcher, and it is independent of the searcher's position on the graph. In this paper, we assume that the target's motion model is Markovian (i.e., it depends solely on its current cell). This assumption allows for a rich space of motion models including those followed by randomly moving and stationary targets. The

searcher knows its own position and it has knowledge of the target's position at a time t in the form a belief distribution over all vertices, $b_N(t)$. Since $b_N(t)$ can be an arbitrary distribution, this formulation allows multi-modal estimates of the target's position. Call the problem so far the Efficient Search Path Planning (ESPP) problem. To extend to MESPP, place more than one searcher on vertices $s_k(t) \in N$. The searchers now gain reward if any of them are on the same vertex as the target. Incorporating additional searchers forces both the state and action space to grow exponentially. Both ESPP and MESPP can be formulated as a Partially Observable Markov Decision Process (POMDP) with deterministic actions, but we do not give the full formulation here due to space constraints.

This formulation shows that the MESPP requires the optimization of the objective function in Equation 1. Searchers choose a feasible set of paths that maximize the expected probability of intersecting the target's path at the earliest possible time (before reward is heavily discounted). To see why, expand the graph G into a time augmented graph G'. Each node N' now represents a cell in the environment at a discrete time point. The reward function can now be seen as the expected intersection between the searchers' paths and the target's path. Let $A \subset N'$ be a feasible set of searcher paths in G', Ψ be the space of all possible target paths, $P(Y)$ be the probability of the target taking path Y, and $F_Y(A)$ be the discounted reward received by path A if the target chooses path Y. Theorem 1 shows that the resulting objective function is nondecreasing and submodular.

Theorem 1: Equation 1 is a nondecreasing, submodular set function. This is the objective function optimized by the MESPP.

$$F(A) = \sum_{Y \in \Psi} P(Y) F_Y(A) \qquad (1)$$

The proof of Theorem 1 along with a formal definition of submodularity is given in the Appendix. This result is used in the next section to prove optimality bounds on sequential allocation in this domain. Since the Markov assumption is made on the target's motion model, calculating the expectation in Equation 1 can be done using diffusion matrices rather than enumerating paths. This greatly simplifies the computation of $F(A)$ over the space of searcher paths. This is explained in more detail below.

IV. ALGORITHM DESCRIPTION

In this section, we describe our algorithm for non-adversarial coordinated search with multiple robots utilizing sequential allocation and finite-horizon planning. We show theoretical bounds by taking advantage of the nondecreasing submodularity of the MESPP objective function. This somewhat surprising result shows that sequential allocation, an algorithm linearly scalable in the number of searchers, generates near-optimal paths in the MESPP domain.

A. Sequential Allocation

As noted above, the joint space of searcher paths grows exponentially in the number of searchers. Explicitly coordinating

by planning in this space quickly leads to intractably large planning spaces with multiple searchers. As an alternative, we propose that searchers choose their paths sequentially. Algorithm 1 gives pseudocode for our sequential allocation algorithm in the MESPP domain. The algorithm maintains a list of nodes $V \subset N'$ that have been visited by the searchers. Note that N' is the time-augmented version of the nodes N in the environment, which allows for revisiting areas in the environment more than once. The searchers choose paths A_k that maximize the objective function $F(V \cup A_k)$ and then add the nodes they have visited to V. Effectively, subsequent searchers treat the paths of previous searchers as "given", and they are not allowed to change them. Sharing nodes to update V is an instance of implicit coordination as described above. Since the search space does not grow with the number of searchers, the complexity of sequential allocation is linear in the number of searchers. It is important to note that while path planning occurs sequentially, the execution of paths is simultaneous.

Algorithm 1 Sequential allocation MESPP algorithm

Input: Multi-agent efficient search problem
% $V \subset N'$ is the set of nodes visited by searchers
$V \leftarrow \emptyset$
for all searchers k **do**
    % $A_k \subset N'$ is a feasible path for searcher k
    % Finding this $\arg\max$ solves the ESPP for searcher k
    $A_k \leftarrow \arg\max_{A_k} F(V \cup A_k)$
    $V \leftarrow V \cup A_k$
end for
Return A_k for all searchers k

Algorithm 1 requires maximizing the objective function for the ESPP problem as a subroutine. Any algorithm for solving the ESPP problem can be inserted here. However, if the ESPP solver is bounded, the nondecreasing submodularity of the objective function leads to bounds on the performance of sequential allocation. Theorem 2 from previous work shows that sequential allocation leads to theoretical guarantees in the informative path planning domain.

Theorem 2: From Singh et al. [2]: Let κ be the approximation guarantee for the single path instance of the informative path planning problem for any nondecreasing, submodular function. Then sequential allocation achieves an approximation guarantee of $(1 + \kappa)$ for the multi-robot informative path planning problem.

The findings in Theorem 1 can be leveraged to extend these results to MESPP. Corollary 1 states that if an ESPP solver has an approximation guarantee of κ, then sequential allocation on the MESPP will yield an approximation guarantee of $(1 + \kappa)$.

Corollary 1: If a solver achieves an approximation guarantee of κ for the ESPP problem, sequential allocation yields an approximation guarantee of $(1 + \kappa)$ for the Multi-robot ESPP (MESPP) problem.

Proof: The proof of Corollary 1 is immediate from Theorem 2 and Theorem 1. Theorem 2 states that sequen-

tial allocation achieves this bound for any single-agent path planning problem optimizing a nondecreasing, submodular function. Theorem 1 shows that the ESPP problem requires the optimization of such an objective function. ∎

Here an approximation guarantee κ states that if the ESPP solver returns a path $A \subset N'$, then $F(A) \geq \frac{1}{\kappa} F(A^{OPT})$, where A^{OPT} is the set of nodes visited by an optimal path. Clearly, $\kappa \geq 1$ since $F(A)$ cannot be greater than the optimal reward. The case where $\kappa = 1$ corresponds to solving the ESPP problem optimally. In this case, sequential allocation can achieve no worse than half ($\kappa = 2$) the optimal reward. This theoretical result allows single-agent performance bounds to be extended to the multi-agent case using a linearly scalable algorithm, albeit with a loss in approximation quality. The next section presents a bounded algorithm for solving the ESPP problem using finite-horizon path enumeration.

B. Finite-Horizon Planning

In large environments, even the single-agent ESPP may be intractable to solve optimally (or even near-optimally) due to the computational overhead of considering many infinite-horizon paths. In these cases, one option is for the searchers to plan a finite number of cells ahead and choose the best path to that horizon. Any time while traversing this path, the searcher can plan again utilizing new information on a new horizon. This leads to an online solution to MESPP, and it allows for the incorporation of measurements of the target's position as they become available. Algorithm 2 gives pseudocode for solving the ESPP problem using finite-horizon path enumeration. Because the finite-horizon method relies on path enumeration to solve ESPP, it scales exponentially with the search depth: $O(b^d)$, where b is the maximum branching factor of the search graph, and d is the search depth in cells.

Algorithm 2 Finite-horizon path enumeration for ESPP

Input: Single-agent efficient search problem
for All paths A to horizon d **do**
    Calculate $F(A)$
end for
Return $A \leftarrow \arg\max_A F(A)$

Lemma 1 derives optimality bounds for finite-horizon path enumeration, and the result extends to the multi-robot case with sequential allocation as in Theorem 3.

Lemma 1: Finite-horizon path enumeration on the ESPP problem achieves a lower bound of:

$$F(A^{FH}) \geq F(A^{OPT}) - \epsilon, \quad (2)$$

where A^{FH} is the path returned by finite-horizon path enumeration, A^{OPT} is the optimal feasible path, and $\epsilon = R\gamma^{d+1}$.

Proof: Finite-horizon path enumeration achieves the optimal reward inside the horizon depth for ESPP because it checks all paths. The maximum reward that could be gained outside the horizon depth is given by $\epsilon = R\gamma^{d+1}$, where R is the reward received for locating the target, γ is the discount factor, and d is the search depth. The bound is immediate. ∎

Theorem 3: Finite-horizon path enumeration with sequential allocation on the K-robot MESPP problem achieves a lower bound of:

$$F(A_1^{FH}, \ldots, A_K^{FH}) \geq \frac{F(A_1^{OPT}, \ldots, A_K^{OPT}) - \epsilon}{2} \quad (3)$$

Proof:

The maximum reward outside the horizon remains the same as in Lemma 1 (i.e., ϵ is unchanged). Since the single robot case achieves the optimal reward within the finite-horizon (i.e., $\kappa = 1$), sequential allocation yields an approximation guarantee as in Corollary 1 as $\kappa + 1 = 2$. ∎

Intuitively, as search depth increases, the bound tightens. Additionally, decreasing the discount factor tightens the bound. This is because lesser discount factors more heavily weight reward gained earlier, which is more likely to be within the finite-horizon. It is important to note that the quality of the bound is independent of the number of searchers. This is a worst-case bound for arbitrary starting distributions and motion models. In practice, searchers can run finite-horizon path enumeration repeatedly on a receding horizon, which leads to performance that far exceeds this lower bound.

C. Measurement Incorporation

If non-line-of-sight measurements of the target's location are available during ESPP, searchers can utilize them to assist in search. The online capabilities of finite-horizon search allow for measurements to be easily incorporated. We use a Bayesian update method to calculate a new belief of the target's position given a measurement.

Denote the probability that the target is in cell i at time t as p_t^i. To incorporate measurements, we recursively estimate the posterior distribution $p_t^i = P(e_t = i | z_1 \ldots z_t)$ for all i, where $z_1 \ldots z_t$ are the measurements received thus far.[1] We are also given a known motion model, which provides $P(e_t = i | e_{t-1} = j)$ for all cells i and j. Once we calculate the posterior, we can renormalize and define a new belief distribution on the target's state. We use a method for modifying the target's belief distribution that calculates the likelihood of a measurement using a finely discretized grid. Using standard Bayesian recursion [15], the posterior can be rewritten as in Equation 4.

$$p_t^i = \eta P(z_t | e_t = i) \sum_j P(e_t = i | e_{t-1} = j) p_{t-1}^j, \quad (4)$$

where η is a normalizing constant.

Assuming a known motion model, this reduces the problem of calculating the posterior to that of calculating a likelihood term $P(z_t | e_t = i)$. Since each cell is represented as a continuous set of points in the map plane, this calculation is difficult. To reduce the complexity of the problem, further discretize each cell into small subcells and calculate a likelihood at the center of each subcell. We denote the M^i subcells of cell i as m^{ij} for all $j \in \{1, \ldots, M^i\}$. The calculation of $P(z_t | c_t = i)$

---

[1]Note: we denote the target's location $e(t)$ as e_t for this section to simplify notation.

---

is now one of calculating a likelihood at many points and then taking the sum of these likelihoods.

For range measurements, we calculate $P(z_t | e_t = i)$ by determining the expected range value for the center of each subcell. Let q^{ij} be the Euclidean distance from the ranging sensor to subcell m^{ij}, and let r_t be the received range measurement with assumed Gaussian noise variance σ^2. The likelihood is then calculated as in Equation 5.

$$P(z_t | e_t = i) = \sum_{j=1}^{M^i} N(r_t; q^{ij}, \sigma^2) \quad (5)$$

After receiving a new measurement, searchers can use sequential allocation and finite-horizon path enumeration to replan using the new belief distribution on the target's location. Combining sequential allocation, finite-horizon path enumeration, and Bayesian measurement updating yields a scalable and online algorithm for solving the MESPP problem. Since searchers replan after receiving measurments, online measurement incorporation heuristically improves path quality but does not affect theoretical guarantees.

V. EXPERIMENTAL RESULTS

A. Simulated Results

To test our MESPP algorithm, we ran simulated trials on a multi-agent coordinated search simulation in C++ on a 3.2 GHz Pentium 4 processor. Our simulation allows for multiple searchers and both stationary and moving targets. We assumed that the average speed of the target is $1 \ m/s$, and that it moves holonomically between cell boundaries. The searchers also move with a maximum speed of $1 \ m/s$, which would be a reasonable speed for state-of-the-art autonomous vehicles. The searchers start in the same location for all trials, and the location of the target is initialized at random on the map. We ran simulated experiments in the museum ($150 \ m \times 100 \ m$) and office ($100 \ m \times 50 \ m$) environments shown in Figure 1. In all tests, the performance metric is the average reward received over many trials. For a given trial, reward received is calculated as $R(t_c) = R\gamma^{t_c}$, where R is the reward for locating the target, γ is the discount factor, and t_c is the time at which the target was found. We arbitrarily set $R = 1$ and $\gamma = 0.95$ for all experimental trials.

Our results in Figure 2 compare finite-horizon path enumeration (horizon depth five) to the infinite horizon POMDP solution for a single searcher. With a single searcher, the POMDP formulation of ESPP is still solvable using Heuristic Search Value Iteration (HSVI2) [7]. Our results show that, for the single searcher case, finite-horizon path enumeration yields average rewards competitive with those generated by the HSVI POMDP solution. We attempted running HSVI with two searchers, but these trials were unsuccessful because the exponentially increased state-action space would not fit in memory. This demonstrates the poor scalability of the POMDP formulation of the MESPP problem.

Since solving the POMDP formulation is intractable for multiple searchers, we introduce for comparison a finite-horizon explicit coordination algorithm that is identical to Algorithm 2 except that it enumerates paths for all searchers in the joint space. This scales $O(b^{dK})$, where b is the branching factor, d is the search depth, and K is the number of searchers. Figure 3 gives a comparison of reward received by sequential allocation and this explicit coordination algorithm. Since explicit coordination grows intractable at large lookahead depths, a depth of two was used for comparison. Figure 3 also shows a lower bound for sequential allocation calculated from Corollary 1 using the explicit coordination results. This bound is the lowest reward that sequential allocation could achieve if explicit coordination yielded the optimal reward. On both maps, sequential allocation greatly outperforms its lower bound.

These simulated experiments demonstrate that implicit coordination with sequential allocation yields results nearly equivalent to those achieved through explicit coordination. In sharp contrast with explicit coordination's exponential scalability, sequential allocation is linearly scalable in the number of searchers. Figure 4 demonstrates the scalability of sequential allocation by showing reward received with up to five searchers in the museum and office.
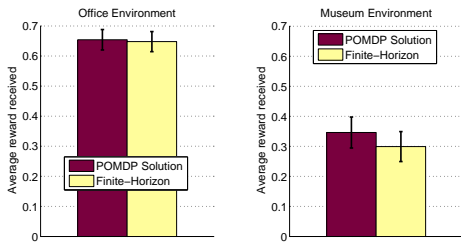

Fig. 2. Comparison of finite-horizon path enumeration (lookahead depth five) versus the POMDP solution for a single searcher in two complex simulated environments. Error bars are one standard error of the mean (SEM), and averages are over 200 trials. Target and searchers move at a maximum speed of $1\ m/s$.
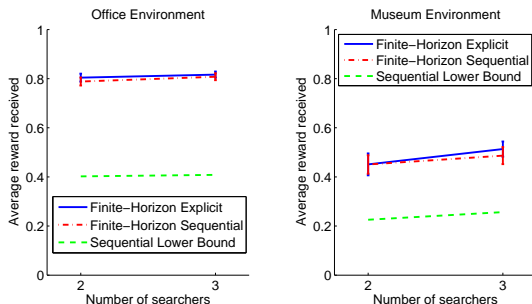

Fig. 3. Comparison of sequential allocation versus explicit coordination in two complex simulated environments. Finite-horizon path enumeration with lookahead depth two was used for both methods. Error bars are one standard error of the mean (SEM), and averages are over 200 trials. Target and searchers move at a maximum speed of $1\ m/s$. Sequential allocation greatly outperforms its lower bound.
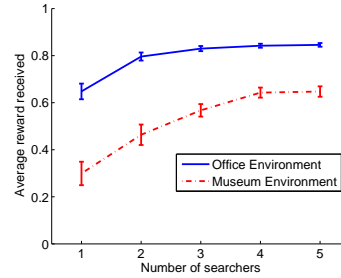

Fig. 4. Multiple searcher scalability trials for finite-horizon path enumeration (lookahead five) and sequential allocation in simulated environments. Error bars are one standard error of the mean (SEM), and averages are over 200 trials. Target and searchers move at a maximum speed of $1\ m/s$.

B. Ranging Radio Measurements

One major application of MESPP is that of finding lost first responders in disaster scenarios. To better model this scenario, we set up an urban response test environment using a Pioneer robot and five Multispectral ranging radio nodes [16]. These sensors use time-of-flight of ultra-wideband signals to provide inter-node ranging measurements through walls. They have an effective range of approximately $30\ m$ indoors and provide accuracy approximately within $1-2\ m$. In our experiments, the Pioneer robot acted as a lost first responder and was teleoperated around the environment carrying a ranging radio node. Four stationary nodes were placed in surveyed locations around the environment to provide range to the Pioneer. The Pioneer also carried a SICK laser rangefinder, and its location was found using laser AMCL-SLAM methods from the Carmen software package [15]. The Pioneer's laser localization was used for ground truth but was not used to assist in search. The Pioneer's maximum speed was set to $0.3\ m/s$, the maximum that provided consistent laser localization. Figure 5 shows a photograph of the Pioneer robot as well as the office environment used for testing.
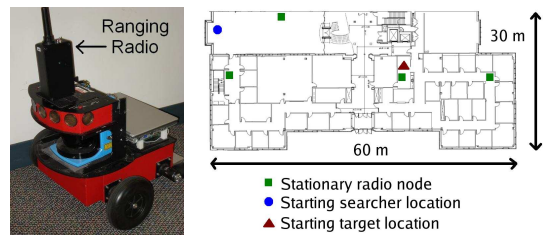

Fig. 5. Photograph of Multispectral ultra-wideband ranging radio mounted on Pioneer robot (left) and floorplan of testing environment (right). The robot was teleoperated around the environment to act as the moving target.

After gathering data from the ultra-wideband ranging sensors, simulated searchers were added to the environment. These searchers have access to the ranging measurements from the stationary nodes in the environment, which allows them to utilize real range data from the experiment to find the target in the simulated world. The searchers were given a maximum speed of $0.3\ m/s$ to match that of the Pioneer target. Figure 6 shows the results for a single searcher in these

"hybrid" trials. As in the purely simulated trials, the finite-horizon path enumeration method provides nearly equivalent reward as the POMDP solution. Figure 7 shows results with two searchers using sequential allocation. As above, sequential allocation is competitive with explicit coordination.

Since the experiments are run in playback, we can vary the number of sensors used by turning off some sensors' data streams. Figure 8 shows average rewards using an increasing number of ranging radio nodes. The zero node case corresponds to search without measurements. The results show that adding more searchers leads to decreasing capture times. Increasing the number of measurement beacons also leads to decreasing capture times. These results suggest that if a small number of searchers are available, this can be compensated with more measurement beacons, and vice versa.

Fig. 8. Multiple searcher scalability trials for finite-horizon path enumeration (lookahead five) and sequential allocation with ranging radio measurements. Error bars are one standard error of the mean (SEM). Target and searchers move at a maximum speed of $0.3 \ m/s$. The zero node case corresponds to the absence of measurements.
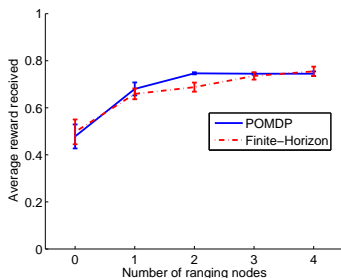

Fig. 6. Comparison of finite-horizon path enumeration (lookahead depth five) versus the POMDP solution for a single searcher using ultra-wideband ranging radio measurements from experimental trials. Error bars are one standard error of the mean (SEM), and the searcher moves at a maximum speed of $0.3 \ m/s$.
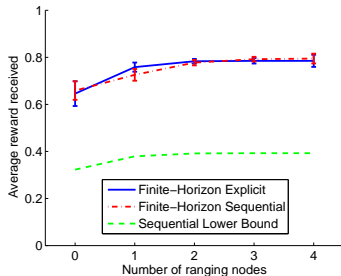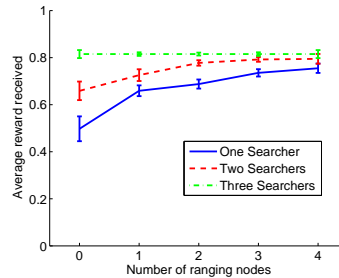

Fig. 7. Comparison of sequential allocation versus explicit coordination with two searchers using ranging radio measurements. Finite-horizon path enumeration with lookahead depth four was used for both methods. Error bars are one standard error of the mean (SEM). Target and searchers move at a maximum speed of $0.3 \ m/s$. As in simulated trials, sequential allocation greatly outperforms its lower bound

VI. Conclusion and Future Work

This paper has presented a scalable algorithm for solving the Multi-robot Efficient Search Path Planning (MESPP) problem of locating a non-adversarial target using multiple robotics searchers. We have defined the MESPP problem and shown how it can be modeled using a Partially Observable Markov Decision Process (POMDP). We have also shown that current POMDP solvers are incapable of handling large instances of MESPP. Our proposed algorithm uses sequential allocation

and finite-horizon path enumeration to remain computationally tractable for multiple searchers in large environments. We have given a rigorous theoretical analysis that shows the near-optimality of sequential allocation in this domain by exploiting the nondecreasing submodularity of the MESPP objective function. Sequential allocation is an instance of *implicit coordination* during which multiple robots share information rather than planning in the joint path space. Implicit coordination is linearly scalable, and it remains tractable in large problem instances when exponential methods using explicit coordination are far beyond computational limits. Our simulated and experimental results using ultra-wideband ranging radios show the performance of our algorithm in complex environments.

One extension is to apply our algorithm to the case where actions of searchers are no longer fully deterministic. For instance, a searcher may have a fifty percent chance of failing to move because of rubble blocking the way. The POMDP formulation of MESPP can easily express this scenario. The solution to this POMDP would no longer be a deterministic searcher path, but it would instead be a distribution over paths. Even though submodular set analysis does not directly apply, the resulting objective function on distributions over paths may still show qualities related to submodularity leading to theoretical guarantees for sequential allocation.

Throughout this paper, we have made the assumption that the target's motion model is known to the searchers. If instead a set of candidate motion models are known, an algorithm like SATURATE [13] can be used to generate a multi-searcher policy that performs well against any of the models. This extension would allow for both stationary and moving target models to be considered simultaneously.

Finally, we have assumed that the target's motion is non-adversarial. If the target may be actively avoiding the searchers, the properties of submodularity no longer hold. In such scenarios, an algorithm with both average-case and worst-case guarantees would be desired. To the best of our knowledge, the development of such a search algorithm is still an open problem. However, methods utilizing implicit coordination show great promise in providing near-optimal solutions even in such challenging multi-robot domains.

REFERENCES

[1] V. Kumar, D. Rus, and S. Singh, "Robot and sensor networks for first responders," *Pervasive Computing*, pp. 24–33, 2004.

[2] A. Singh, A. Krause, C. Guestrin, W. Kaiser, and M. Batalin, "Efficient planning of informative paths for multiple robots," in *Proc. Int'l Joint Conf. on Artificial Intelligence*, 2007.

[3] A. Blum, S. Chawla, D. Karger, T. Lane, A. Meyerson, and M. Minkoff, "Approximation algorithms for orienteering and discounted-reward tsp," *SIAM Journal on Computing*, vol. 37, no. 2, pp. 653–670, 2007.

[4] T. Parsons, "Pursuit-evasion in a graph," in *Theory and Applications of Graphs*, Y. Alavi and D. Lick, Eds. Springer, 1976, pp. 426–441.

[5] L. Guibas, J. Latombe, S. LaValle, D. Lin, and R. Motwani, "Visibility-based pursuit-evasion in a polygonal environment," *Int'l Journal of Comp. Geometry and Applications*, vol. 9, no. 5, pp. 471–494, 1999.

[6] N. Roy, G. Gordon, and S. Thrun, "Finding approximate pomdp solutions through belief compression," vol. 23, pp. 1–40, 2005.

[7] T. Smith, "Probabilistic planning for robotic exploration," Ph.D. dissertation, Carnegie Mellon University, 2007.

[8] N. Kalra, "A market-based framework for tightly-coupled planned coordination in multirobot teams," Ph.D. dissertation, Carnegie Mellon University, 2006.

[9] B. Gerkey, S. Thrun, and G. Gordon, "Parallel stochastic hill-climbing with small teams," in *Proc. 3rd Int'l NRL Workshop on Multi-Robot Systems*, 2005.

[10] A. Sarmiento, R. Murrieta-Cid, and S. Hutchinson, "A multi-robot strategy for rapidly searching a polygonal environment," in *Proc. 9th Ibero-American Conference on Artificial Intelligence*, 2004.

[11] C. Guestrin, A. Krause, and A. Singh, "Near-optimal sensor placements in gaussian processes," in *Proc. Int'l Conf. on Machine Learning*, 2005.

[12] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *Proc. 22nd Conf. on AI*, 2007.

[13] A. Krause, B. McMahan, C. Guestrin, and A. Gupta, "Selecting observations against adversarial objectives," in *Proc. Neural Information Processing Systems*, 2007.

[14] J. S. Singh and M. D. Wagh, "Robot path planning using intersecting convex shapes: Analysis and simulation," *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 2, April 1987.

[15] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA: MIT Press, 2005.

[16] "Multispectral, company website: http://www.multispectral.com/," 2008.

APPENDIX: PROOF OF THEOREM 1

Proof: Consider a time-augmented version of graph G, $G'(N', E')$. Level 1 of G' represents the vertexes of G at time 1, and directed edges from level 1 connect to reachable vertexes at time 2. Extend this graph down to a max time τ. The graph now contains vertices $N' = N \times T$ where $T = \{1, \ldots, \tau\}$. This result extends to the infinite case by making τ arbitrarily large. $\mathbf{P}(N')$ denotes the powerset of N', i.e. the set of (time stamped) node subsets. A function $F : \mathbf{P}(N') \to \Re_0^+$ is called *nondecreasing* iff

$$A \subseteq B \Rightarrow F(A) \leq F(B).$$

It is called *submodular* iff

$$A \subseteq B \Rightarrow F(A \cup C) - F(A) \geq F(B \cup C) - F(B).$$

(In the above $A, B \in \mathbf{P}(N')$ and $C = \{(m, t)\} \subseteq N'$ – i.e. C is a singleton.)

In the following, for a given $Y \subseteq N'$ and any $A \subseteq N'$ we define $t_A = \min\{t : (m, t) \in A \cap Y\}$, $F_Y(A) = \gamma^{t_A}$, with the understanding that $\gamma \in (0, 1)$, $\min \emptyset = \infty$, and $\gamma^\infty = 0$.

We now show that for every $Y \subseteq N'$ the function $F_Y(A)$ is nondecreasing and submodular. Take an arbitrary Y and fix it for the proof. Take any $A, B \subseteq N'$ and any $C = \{(m_0, t_0)\} \subseteq N'$. We have:

$t_A = \min\{t : (m, t) \in A \cap Y\}$,
$t_B = \min\{t : (m, t) \in B \cap Y\}$,
$t_C = \min\{t : (m, t) \in C \cap Y\}$.
$A \subseteq B \Rightarrow \{t : (m, t) \in A \cap Y\} \subseteq \{t : (m, t) \in B \cap Y\} \Rightarrow$
$t_A \geq t_B \Rightarrow F_Y(A) = \gamma^{t_A} \leq \gamma^{t_B} = F_Y(B)$.

Hence $F_Y(\cdot)$ is nondecreasing.

Now, regarding submodularity, note that, since C is a singleton, we have two cases: either $C \cap Y \neq \emptyset$ and so $t_c = t_0 < \infty$; or $C \cap Y = \emptyset$ and so $t_c = \infty$. We examine the two cases separately.

Case I, $t_C < \infty$. In this case we have three subcases.

1) $t_B \leq t_A \leq t_C$. Then $F_Y(B \cup C) = \gamma^{t_B}$, $F_Y(B) = \gamma^{t_B}$, $F_Y(A \cup C) = \gamma^{t_A}$, $F_Y(A) = \gamma^{t_A}$ and $F_Y(A \cup C) - F_Y(A) = \gamma^{t_A} - \gamma^{t_A} = 0 = \gamma^{t_B} - \gamma^{t_B} = F_Y(B \cup C) - F_Y(B)$.

2) $t_B \leq t_C \leq t_A$. Then $F_Y(B \cup C) = \gamma^{t_B}$, $F_Y(B) = \gamma^{t_B}$, $F_Y(A \cup C) = \gamma^{t_C}$, $F_Y(A) = \gamma^{t_A}$ and $F_Y(A \cup C) - F_Y(A) = \gamma^{t_C} - \gamma^{t_A} > 0 = \gamma^{t_B} - \gamma^{t_B} = F_Y(B \cup C) - F_Y(B)$.

3) $t_C \leq t_B \leq t_A$. Then $F_Y(B \cup C) = \gamma^{t_C}$, $F_Y(B) = \gamma^{t_B}$, $F_Y(A \cup C) = \gamma^{t_C}$, $F_Y(A) = \gamma^{t_A}$ and $F_Y(A \cup C) - F_Y(A) = \gamma^{t_C} - \gamma^{t_A} \geq \gamma^{t_C} - \gamma^{t_B} = F_Y(B \cup C) - F_Y(B)$, since $t_A \geq t_B \Rightarrow \gamma^{t_A} \leq \gamma^{t_B} \Rightarrow -\gamma^{t_A} \geq -\gamma^{t_B}$.

Case II, $t_C = \infty$. Then we have a single subcase: $t_B \leq t_A \leq t_C$ from which follows $F_Y(A \cup C) - F_Y(A) = 0 = F_Y(B \cup C) - F_Y(B)$ as already seen.

In every case the submodularity inequality holds.

The one searcher reward function $F(A)$ (where A is the searcher's path) is defined by

$$F(A) = \sum_{Y \in \Psi} P(Y) F_Y(A),$$

where the summation is over all possible target paths Ψ, and $P(Y)$ is the probability of the target taking path Y. The K-searcher reward function $F(A_1, \ldots, A_K)$ (where A_k is the path of the k-th searcher) is defined by

$$F(A_1 \cup \ldots \cup A_K) = \sum_{Y \in \Psi} P(Y) F_Y(A_1 \cup \ldots \cup A_K).$$

We now show that for any $K = 1, 2, \ldots$ and every $Y \subseteq N'$ the function $F(A_1 \cup \ldots \cup A_K)$ is nondecreasing and submodular. Nondecreasing submodularity is closed under nonnegative linear combinations (and hence expectations). Here $F(A_1 \cup \ldots \cup A_K)$ is the expected value of $F_Y(A)$ where $A = A_1 \cup \ldots \cup A_K$, $F_Y(\cdot)$ is a nondecreasing, submodular function, and the expectation is taken over all possible target paths. ∎