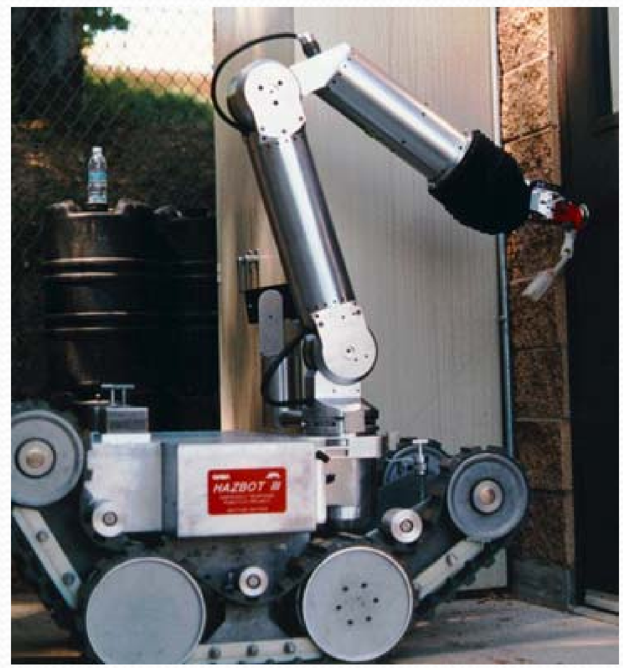
Proofs and Experiments in Scalable, Near-Optimal Search by Multiple Robots

Authors: Geoffrey Hollinger and Sanjiv Singh
Presenter: Matt Wu

Introduction

- Locating a non-adversarial target using multiple robotic searchers



Introduction

- Multi-robot Efficient Search Path Planning (MESPP)
  - Target non-adversarial
  - Try to maximize the probability of finding target over a time interval
  - The environment is known to the searcher

Difficulties

- Path planning for multiple robot is NP-hard
- Problem grow exponentially
- Single robot also difficult in large environments

Solutions

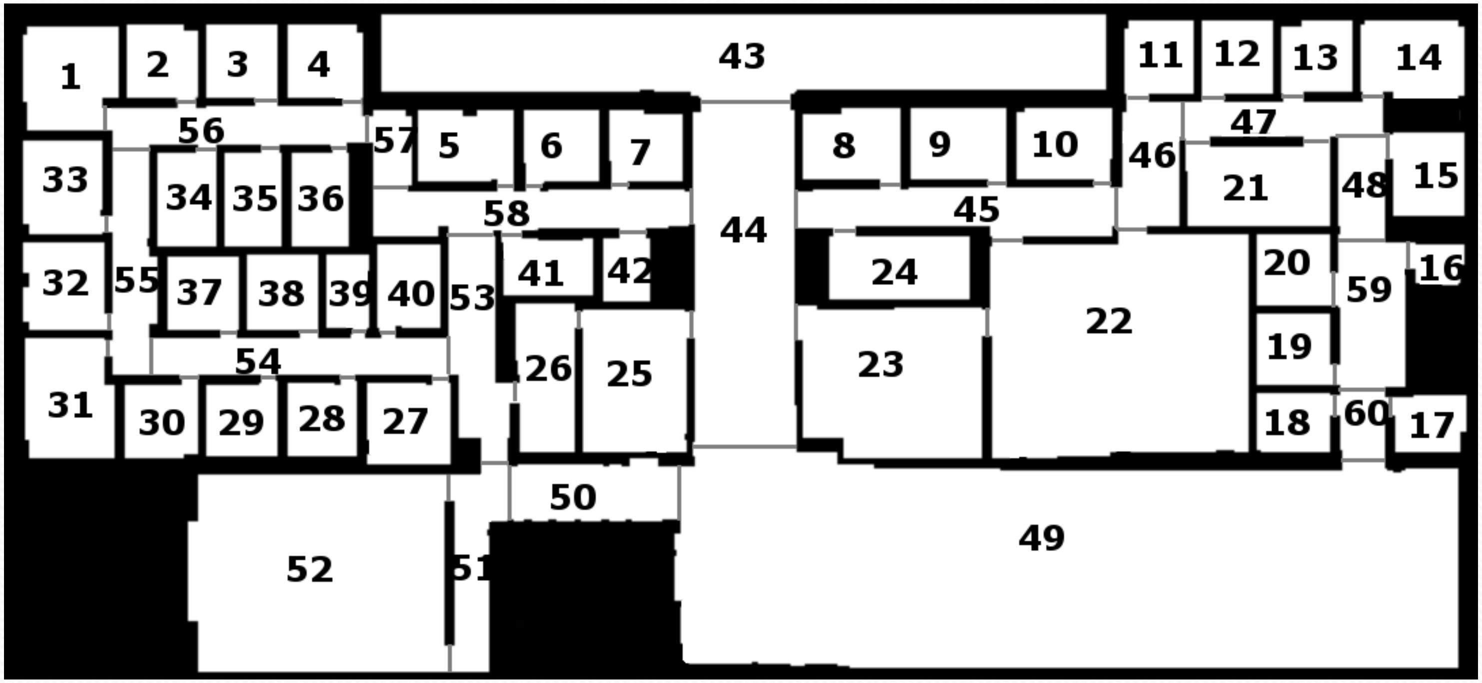
- Approximate algorithm
- Implicit coordination – sequential allocation
- Running time linear
- Good bounds because the problem is submodular

Problem Setup

- Movement model of target: moves according to a Markov chain
- Target's movement model is known to searchers
- Searcher knows its own position and it has knowledge of the target's position at a time *t in the form of a belief distribution* over all possible locations

Example Environment

- Environment: discretized to connected convex regions – undirected graph



Reward Function

- $G(N,E)$, expanded to $G'(N',E)$ for all time t
- At any time t, a searcher exists on vertex $s(t)$ in N
- A target also exists on this graph on vertex $e(t)$ in N
- Reward R gained when $s(t) = e(t)$, discounted by γ^t
- Reward function F for search path A iterating through all possible target path Y:

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

Proving Submodularity

- Show that $F(A) = \sum_{Y \in \psi} P(Y) F_Y(A)$ is submodular
- Reminder of submodularity:

  $$A \subset B, s \notin B \rightarrow F(A+s) - F(A) \geq F(B+s) - F(B)$$
- Why is $F_Y(A)$ given Y submodular?
- Each $F_Y(A)$ given Y is submodular, so the weighted sum of submodular function is also submodular
- Since target moves in a Markov chain, computing the sum is efficient

Algorithm: Sequential Allocation

- Algorithm for MESPP using ESPP:

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: Finite Horizon Planning

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 $A \leftarrow \arg\max_A F(A)$

Approximation Guarantee

- Finite horizon bounds:

$$F(A^{FH}) \geq F(A^{OPT}) - \epsilon \qquad \epsilon = R\gamma^{d+1}$$

- Why?
- Approximation guarantee of MESPP is (1+κ) where κ is the approximation guarantee of the ESPP problem
- MESPP using finite horizon planning:

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

Measurement Incorporation

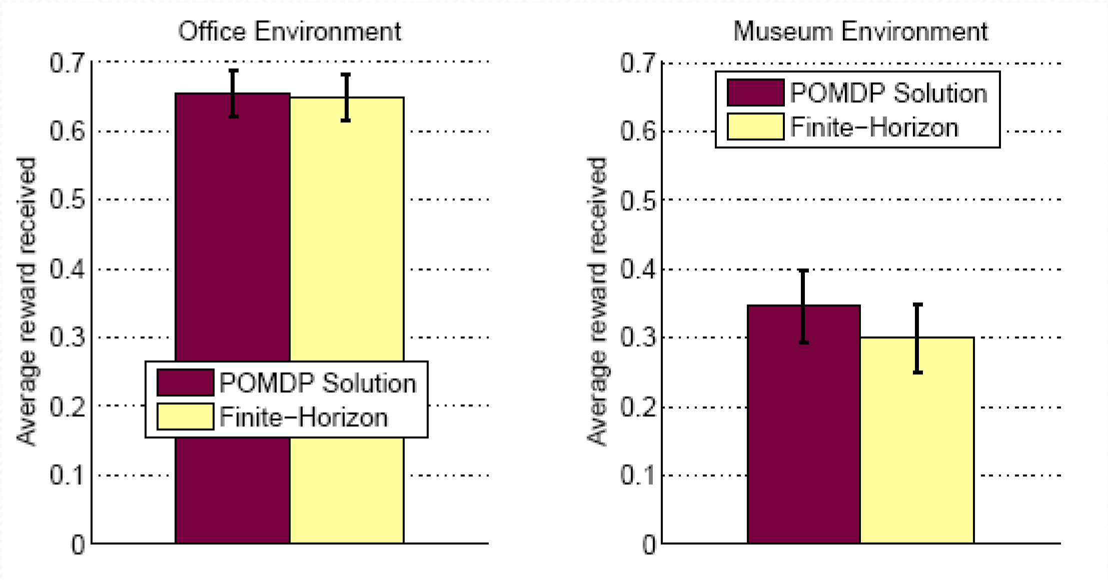
- MESPP does not predict measurement outcomes
- Instead, it uses past measurement outcomes to update belief distribution to plan paths

Experimental Results

- Simulated Result:
  - Target and searchers move at 1 m/s
  - Randomized starting location for target
  - Simulate with museum floor plan of size 150m by 100m and office of size 100m by 50m
- Ranging Radio Measurements
  - Uses range sensors up to 30 m with errors 1-2 m
  - Robot acts as lost first responder, moving at .3 m/s

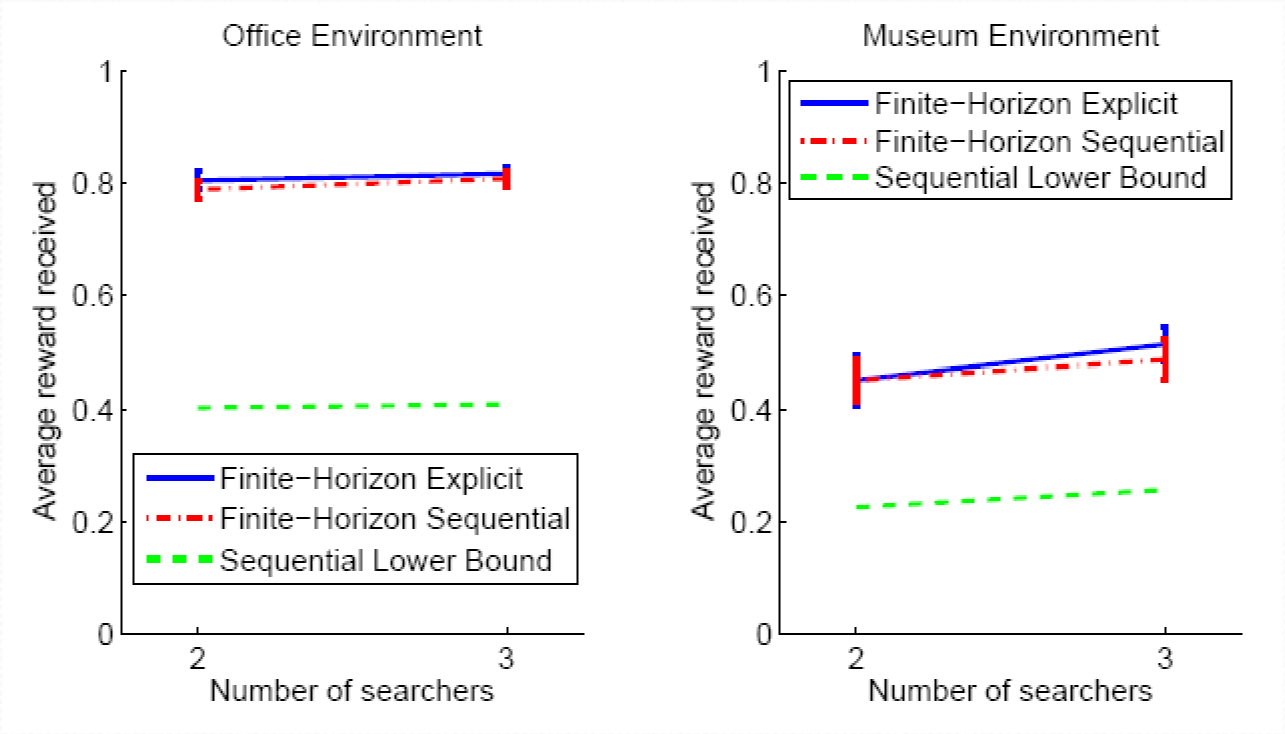
Simulated Results

- Single robot finite horizon compared to optimal



Simulated Results

- Sequential allocation compared to iteration through all search paths by all robots



Ranging Radio Measurements