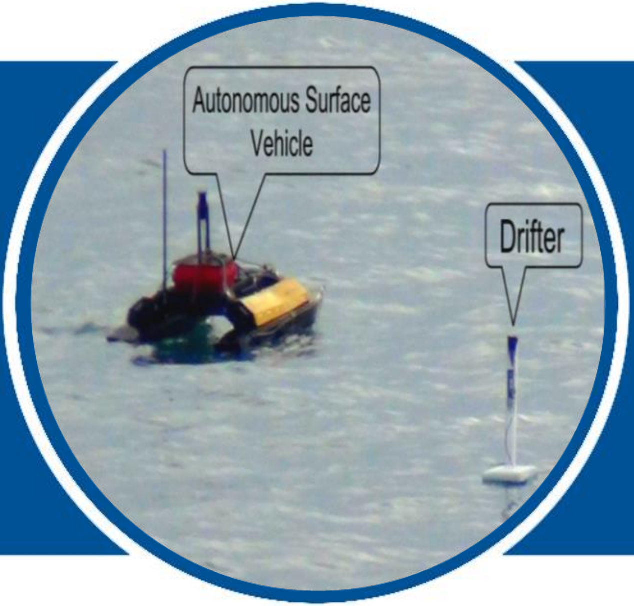


[IROS 2016]

Multi-Target Rendezvous Search

Malika Meghjani, Sandeep Manjanna and Gregory Dudek

2017. 05. 29

Presented By Suzi Kim

Background

Rendezvous Problem

How two players randomly placed in a known search region X can move at speed one to find each other in least expected time?

The rendezvous search problem, Alpern S., 1995

Background

Applications of Rendezvous Problem

- Search and rescue
- Environmental assessment
- Threat detection

Problem Description

Goal

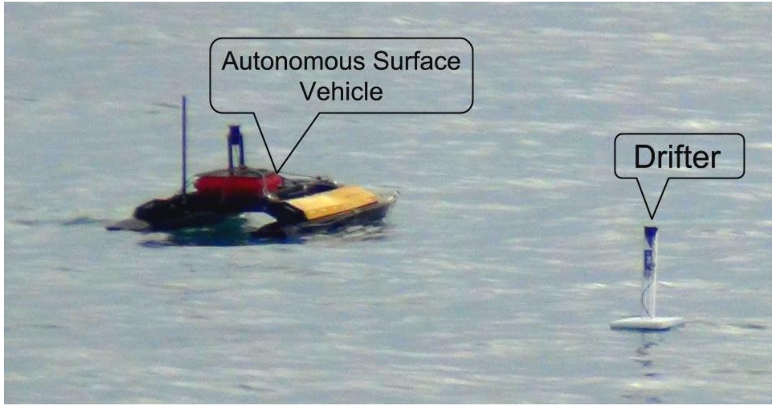
Searching for one or more targets for which we either have an ***initial probability distribution*** describing their suspected initial location or sparse information.

- Minimizing the time to detect targets
- Maximizing the likelihood of detecting targets

Problem Description

Problem Setting

- Marine environments
- Finding a drifting target with a mobile searcher, a robot boat
- Constraints on the communication range



Main Approach

Search Strategies

(1) Global Maxima Search
(2) Heuristic Local Maxima Search
(3) Spiral Search

Main Approach

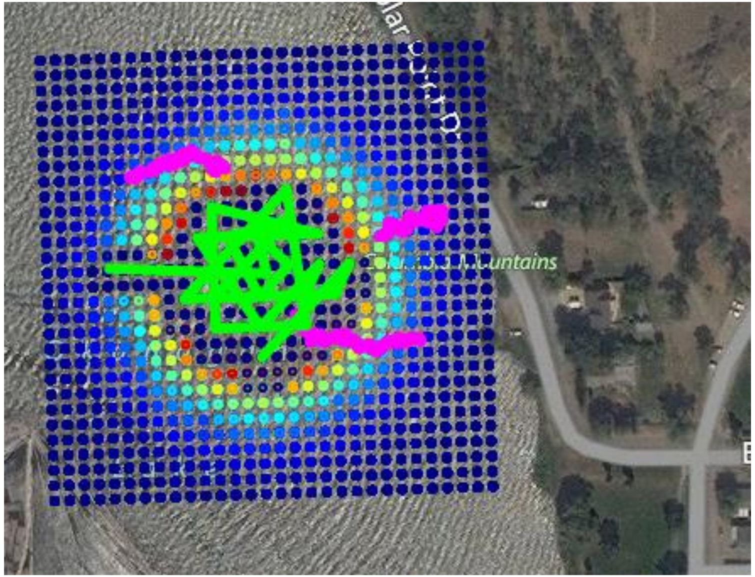
Search Strategies: (1) Global Maxima Search

- Visiting search region with highest probability
- **Method**:
  - Discretize search region into grids
  - Assign with a value equal to the integral of the probability under that grid-cell.
  - Visit the grid-cell with highest value until the target is found or the search region is covered.

Main Approach

Search Strategies: (1) Global Maxima Search

- **Drawback:** Multiple overlapping segments



Main Approach

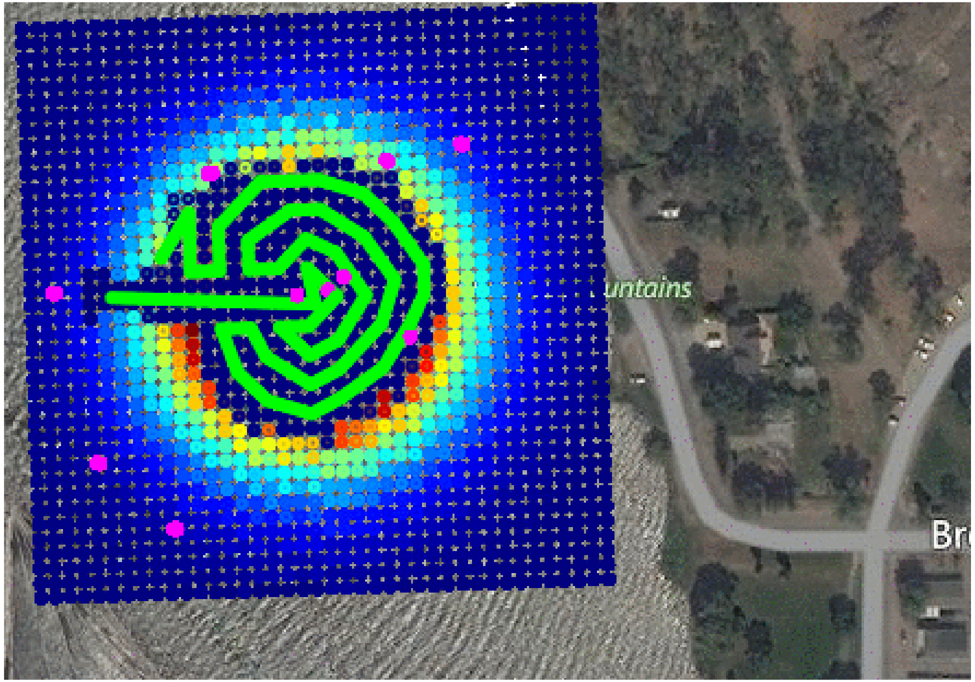
Search Strategies: (2) Heuristic Local Maxima Search

- Visiting search region with highest probability ***within a local maxima-search radius***
- To avoid getting stuck in local maxima and increase the success rate, heuristic method is added.

  - When stuck in local maxima, ***iteratively increase the maxima-search radius*** until the searcher recovers from the local maxima or the radius becomes equal to the radius of the entire search region.

Main Approach

Search Strategies: (2) Heuristic Local Maxima Search



Main Approach

Search Strategies: (3) Spiral Search

- Does not require the discretization of the search region.
- Spiral equation:

$$\begin{pmatrix} x_r(t) \\ y_r(t) \end{pmatrix} = b\theta \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$$

b: a parameter to determine the distance between two consecutive spiral rounds

- Two variants: ***inward*** and ***outward*** spirals
  - Inward spiral search: minimize the escape of the targets
  - Outward spiral search: minimize the search time for a greedy search

Main Approach

Search Strategies: (3) Spiral Search

Experiments

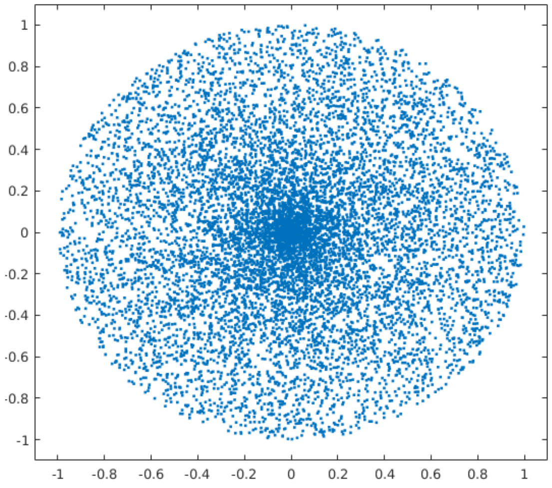
Experiments Setting

- Assuming the target to be a point object, and the searcher to be a disk or a point with a communication radius R_{comm}.
- Radius of search region: *100 meters*
- Maximum speed of the ASV: *1.2m/s*
- Target speed: *0.2m/s*
- Maximum communication range of the robot: *5 meters*
- 1,000 trials for each search strategy.

Experiments

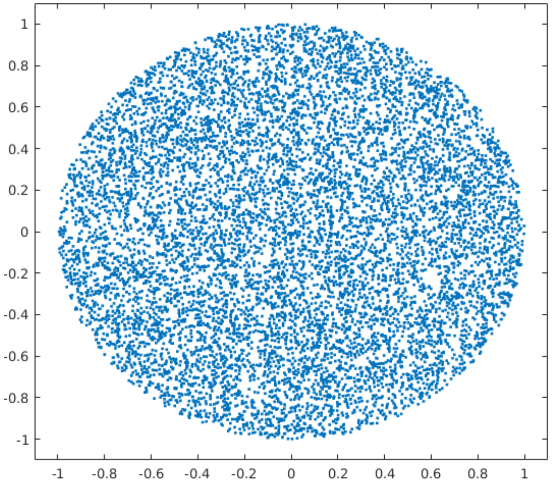
Probability Distribution of Search Region

Triangular



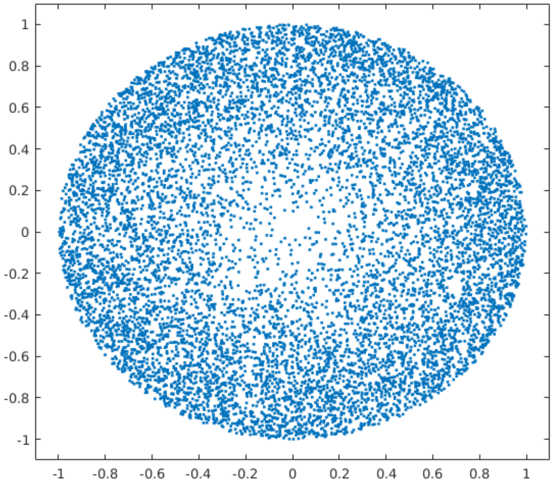
$$(x_d, y_d) = (r cos\theta, r sin\theta)$$

Uniform



$$(x_d, y_d) = (\sqrt{r} cos\theta, \sqrt{r} sin\theta)$$

V-shaped



$$(x_d, y_d) = (r^{1/3} cos\theta, r^{1/3} sin\theta)$$

Experiments

Cost Analysis

- Performance Factors
  (1) Mean Time to Find (MTTF)
  (2) Failure Rate

- Score Function

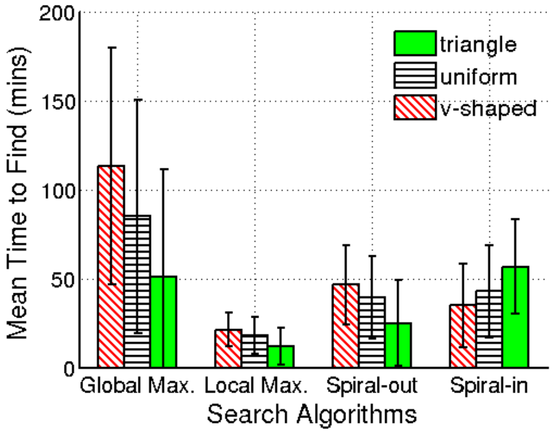
$$C = \sum_{i=1}^{n} (\mathbf{1}_{\mathbf{Found_i}} * t_i + \mathbf{1}_{\mathbf{Found_i^C}} * \beta)$$

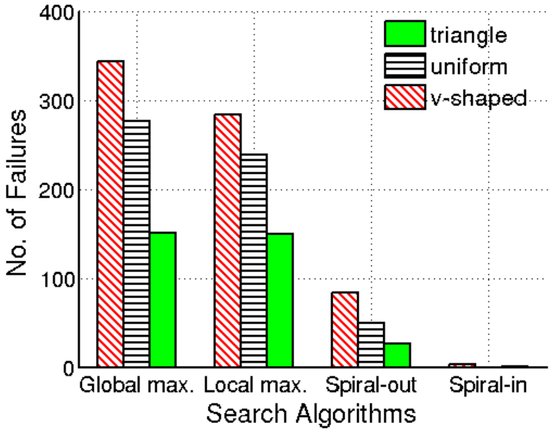
$$\mathbf{1}_{\mathbf{Found_i}} = \begin{cases} 1 & \text{if target } i \text{ is found} \\ 0 & \text{if target } i \text{ is not found,} \end{cases}$$

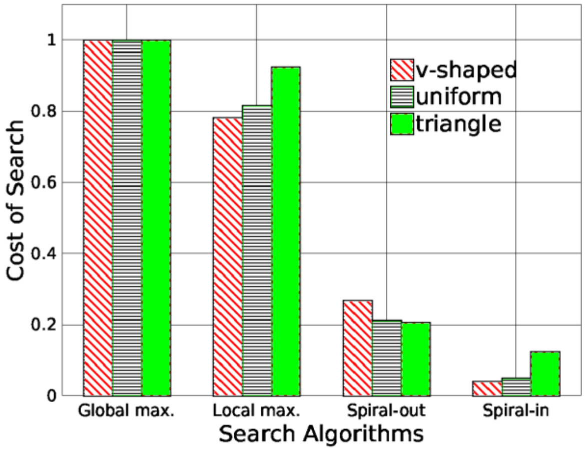
$$\mathbf{1}_{\mathbf{Found_i^C}} = \mathbf{1} - \mathbf{1}_{\mathbf{Found_i}}$$

Experiments

Single Target Search

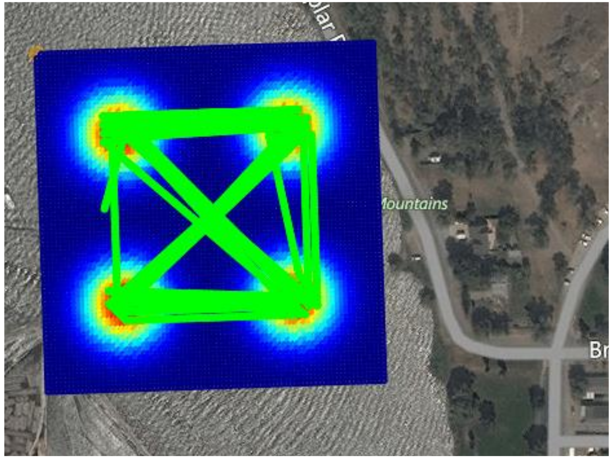






Experiments

Multi-Target Search



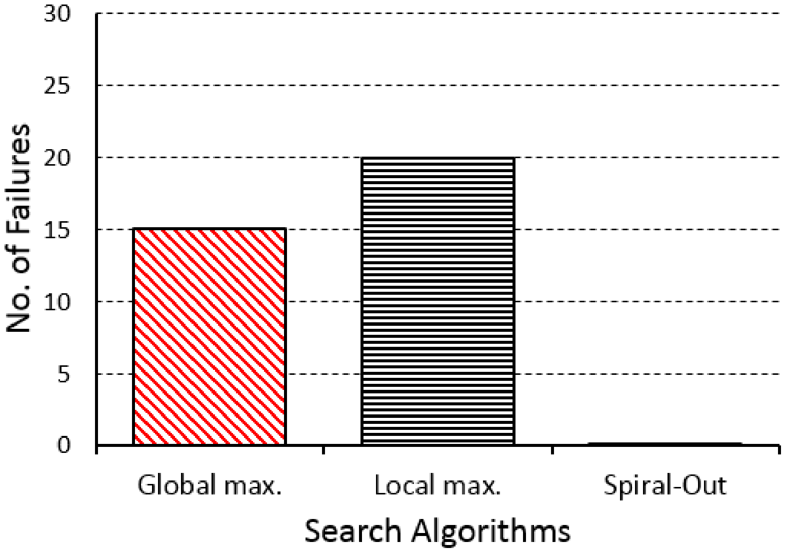
Global Maxima

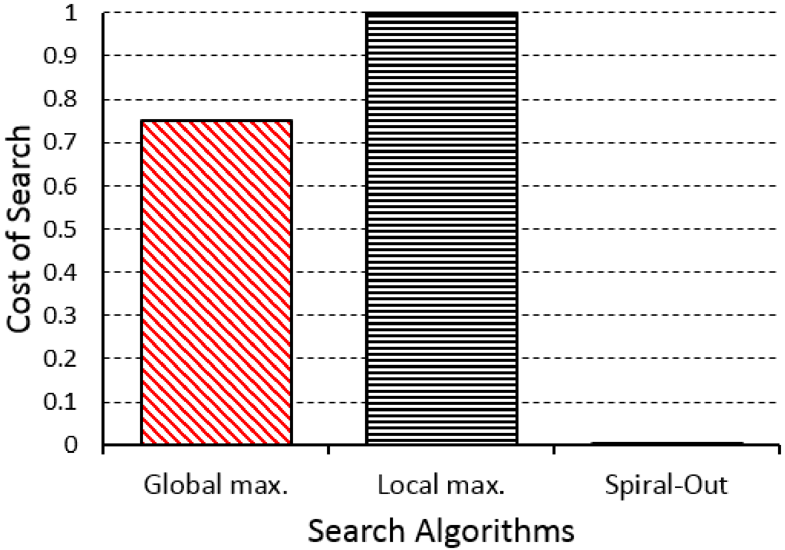
Heuristic Local Maxima

Spiral

Experiments

Multi-Target Search

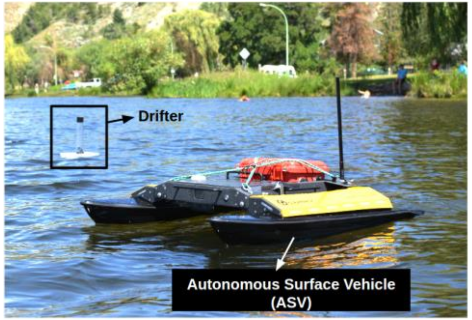




Experiments

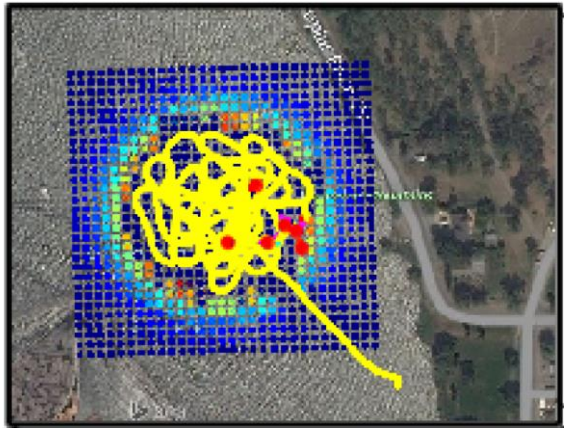
Field Trials

- Searcher Robot
  - Catamaran style Autonomous Surface Vehicle (ASV)
- Target Drifter
  - Equipped with a miniPC (Android MK-802), GPS receiver

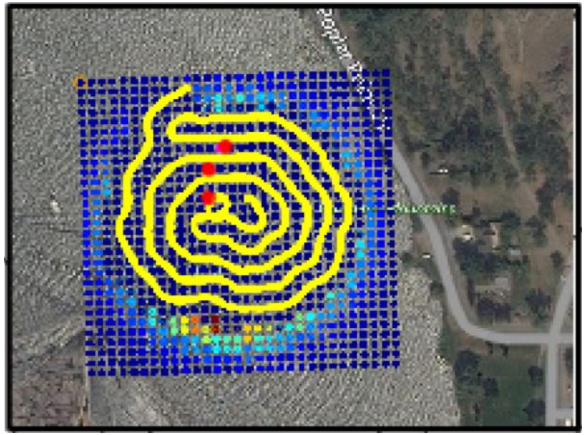


Experiments

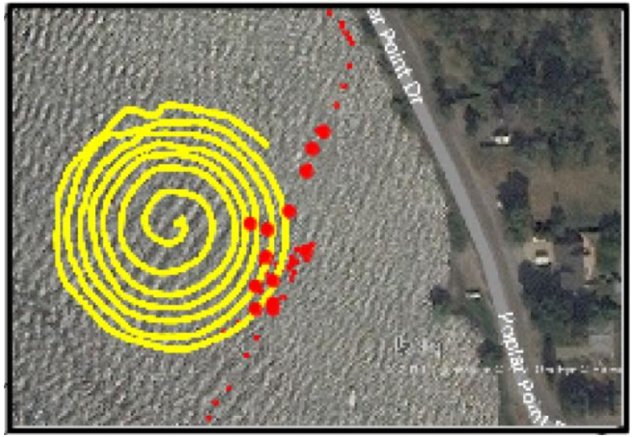
Field Trials



Global Maxima



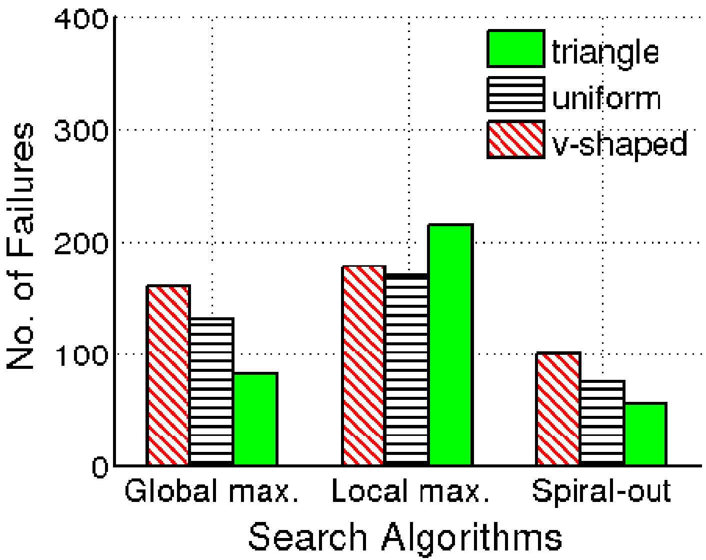
Heuristic Local Maxima

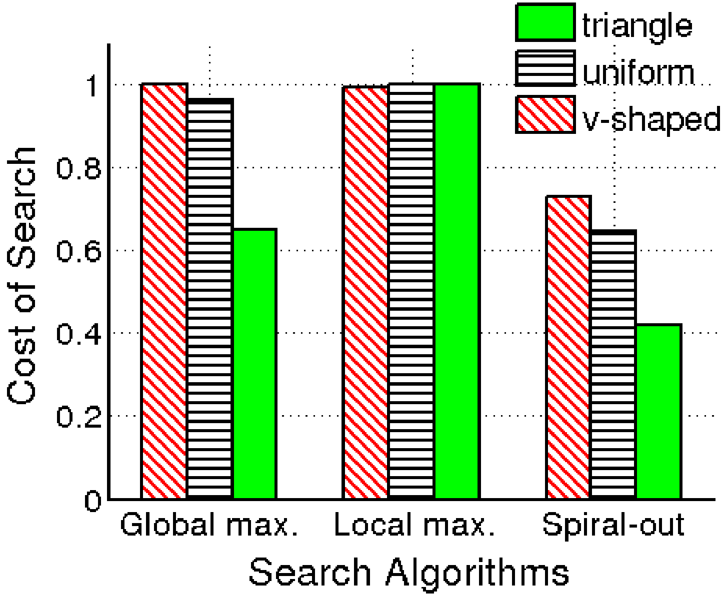


Spiral

Experiments

Field Trials





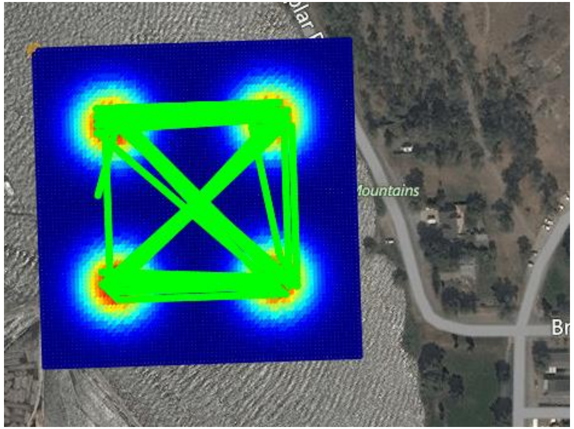
Conclusion

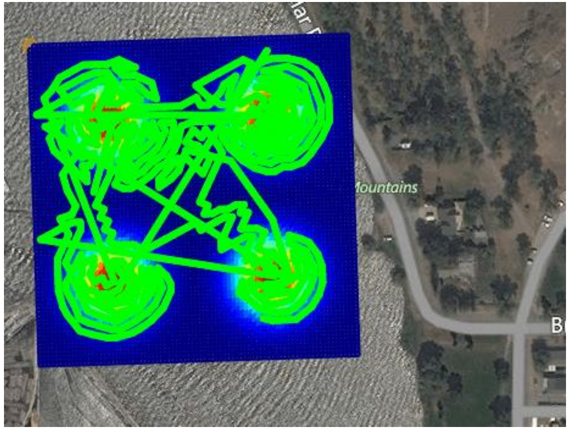
- Compare the performance of three search strategies: Global Maxima, Heuristic Local Maxima, Spiral Search
- Outward spiral search outperforms the other search strategies for both single-target and multi-target experiments.

Conclusion

Future Work

- In multi-target search, transition between targets should be well-optimized.

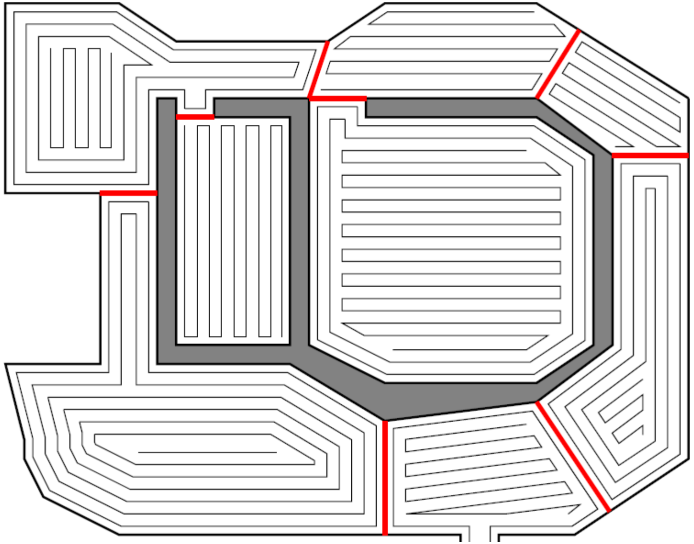


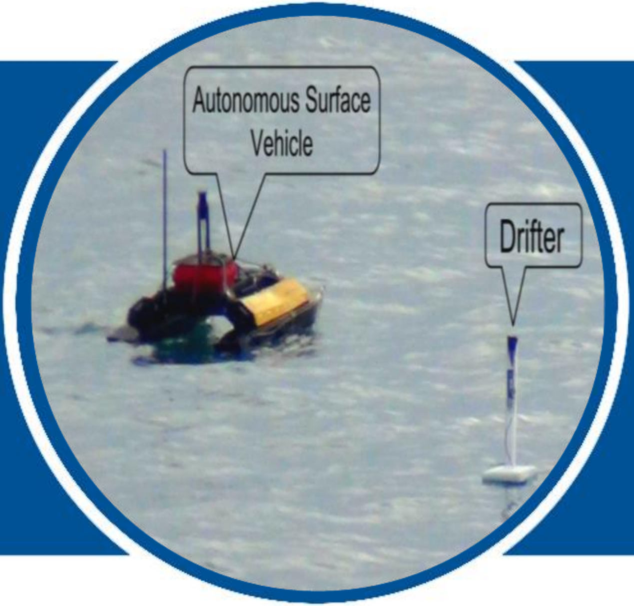


Conclusion

Future Work

- Combining with Coverage Path Planning (CPP), if it requires exhaustive search anyway.





Thank you!

Q&A

[Appendix] Performance Bounds

- The total number of circular rounds (n_s) that robot needs to complete for clearing the entire search region of radius r :

$$n_s = \left\lceil \frac{r}{b} \right\rceil$$

- The time taken to clear one circular round with radius r' :

$$\tau = \frac{2\pi r'}{s_r}$$

- Total time taken by the robot to clear the complete search area :

$$\tau_{tot} = \frac{2\pi n_s}{s_r} \sum_{i=0}^{(n_s-1)} (r - ib)$$

[Appendix] Performance Bounds

Guaranteed Capture

- Capture speed of the robot:

$$s_{cap} = \frac{2\pi r s_d}{b}$$

- The condition of robot speed for a guaranteed capture:

$$s_r > \frac{2\pi r s_d}{b}$$

[Appendix] Performance Bounds

Minimum Time Capture

- Minimized time to capture the target :

$$\tau_{min} = \frac{2\pi n_s}{s_r}(r - (n_s - 1)b)$$

$$\tau_{min} = \frac{2\pi b n_s}{s_r}$$

- The robot should start with an initial radius, $\tau_{min} = b$ and incrementally expand outwards by a factor b.