Detecting and Mapping Hazardous Plumes with Aerial and Surface Robots

Yoonchang Sung, Spencer Buebel and Pratap Tokekar

I. Motivation

This paper addresses a problem of detecting and tracking flows of hazardous agents in aquatic environments using autonomous aerial and aquatic robots. Our overall vision is to develop systems and algorithms for enabling a team of robots to assist emergency responders in disaster scenarios such as dispersal of oil aerosols or radioactive particulates in the environment.

Previous work has shown the value of using Unmanned Surface Vehicles (USVs) for monitoring and sampling spatiotemporal plumes in aquatic environments [1], [2]. However, USVs can only provide a narrow (local) view of the plumes. Detecting whether hazardous agents are present in the environment may require a USV to cover a large portion of the aquatic system, which may take a considerable amount of time. Furthermore, even after detecting the threat, teams of USVs may not be able to keep up with the rapidly spreading plume. Thus, emergency responders may not be informed of the full extent of the hazards, making the response challenging and potentially sub-optimal. This motivates the use of Unmanned Aerial Vehicles (UAVs) which can provide a wider (regional) picture. Teams of UAVs can collectively track the plumes and act as scouts to direct the USVs to regions of interest. A heterogeneous team of UAVs and USVs can provide emergency responders with local and global pictures leading to better responses. In order to exploit this heterogeneity in sensing, we need efficient coordination algorithms and robust systems [3]. We are addressing the algorithmic and systems challenges in tracking spatiotemporal plumes with a heterogeneous team of UAVs and USVs.

We report our progress on detecting and mapping a static plume using a single UAV. We describe our planning algorithm, the overall system design, and preliminary experiments. So far, we have used a Unmanned Ground Vehicle (UGV) as a surrogate USV for experiments in the winter over land. We are in the process of transitioning the algorithms to the USV and conducting larger scale experiments in lakes during Spring.

II. System Design

A. Planning

Consider a grid map-based environment. The grid map contains a set of cells representing a plume region which the location of cells is initially unknown. We assume a plume region to be a single polygon that do not necessarily have to be simple or convex. The objective of planning is to detect and map the plume cells that are unknown *a priori* using a single UAV in minimum time.

A few literatures on exploring a polygonal region have proposed competitive-ratio results for the lawn mowing and milling problems, where the former allows the robot to move outside the boundary of a polygon whereas the latter does not. Icking and Kamphans [4] proposed a strategy of generating a tour of length S, such that $S \leq C + \frac{1}{2}E + H - 3$, where C, E and H denote the number of cells, that of edges and that of obstacles, respectively, for the online milling problem. The algorithms presented by Arkin *et al.* [5] have $(3+\epsilon)$–approximation for offline lawn mowing and 2.5–approximation for offline milling. Kolenderska *et al.* [6] showed a strategy of exploring a simple grid polygon that has a competitive ratio of $\frac{5}{4}$. All these works assumed that a robot can observe up to adjacent neighboring cells. Our problem, however, considers a limited Field-of-View (FoV) sensor that can only observe the current cell which the UAV is located in.

Our proposed algorithm primarily consists of two parts: lawn mowing initially followed by a variant of depth-first search. The UAV initiates a lawn-mower path that covers the entire environment to detect unknown plume cells. As soon as the plume is detected by the sensor, the UAV starts mapping the entire plume area by visiting all plume cells. The mapping strategy follows the nature of depth-first search; the UAV keeps *extreme cells* on a stack that have to be revisited later to search in the opposite direction on either x- (*critical cells*) or y-axis (*split cells*). The algorithm terminates if the stack becomes empty.

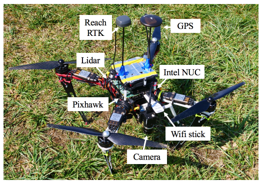
B. Platforms

We use DJI F450 as the UAV platform and Clearpath Husky (*i.e.*, UGV) as a mock USV, as shown in Figure 1. The UAV is set up with Pixhawk ArduPilot which a 3D accelerometer, gyroscope, magnetometer and barometer are internally integrated with. We mounted Intel NUC (NUC7i7BNH) as the onboard computer for UAV which runs Ubuntu 16.04 and ROS Kinetic [7]. The onboard software communicates with Pixhawk, detects the plume and controls the UAV. To estimate the global position (*i.e.*, UTM coordinates) of UAV, two GPSs (Ublox Neo-M8N GPS with compass and Reach RTK) are fused through Pixhawk. With

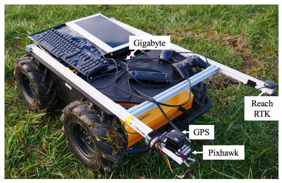
The authors are with the Department of Electrical and Computer Engineering, Virginia Tech, USA. {yooncs8, stbuebel, tokekar}@vt.edu.

This material is based upon work supported by the National Science Foundation under Grant No. 1637915.

fully charged batteries (*i.e.*, $24V$ (6 cells) Li-Ion battery pack), the total flight time is approximately 9 minutes. A single downward-facing camera (Flea3) is equipped with UAV to detect the plume via the vision algorithm.



(a) UAV (DJI F450).



(b) UGV (Clearpath Husky).

Fig. 1. Platforms used for detecting and mapping the plume.

On the UGV side, we employ Gigabyte (GB-BNi7HG4-950) for onboard softwares. To obtain a better maneuverability, we also use Pixhawk ArduPilot empirically because it not only has a robust estimator coming from various sensors but also gives a global position of the UGV. The wireless communication we chose includes Ubiquiti Bullet M5 that supports a $5GHz$ link with an omnidirectional antenna (AirLive WAE-5AG). The UAV and UGV communicate with each other using a pair of master (UAV) and slave (UGV) via ROS.

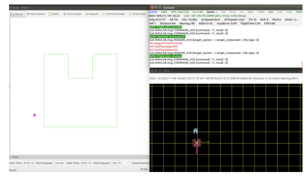
C. Plume Detection

In the preliminary field experiment, we substitute a plume region by a circular tarp. We adopt the blob detection library from OpenCV [8] to detect the tarp from camera images. Among parameters of the blob detection library, we set the area to lie between thresholds and the inertia ratio to ignore non-circular blobs. This makes the detection robust to false-positive measurements. In order to deal with the change in lighting condition due to the amount of sunlight, we strove to find a reasonable exposure value that works in different cloudy conditions. Note also that it should be avoided to have a high exposure time because the high frequency of the wobbling UAV can take enough time to make blurry images while the lens is open.

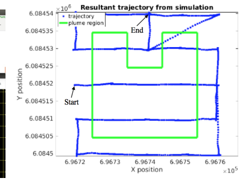
III. Preliminary Experiments

We present simulation results that verify the completeness of the proposed planning strategy. We utilized a software-in-the-loop simulator [9] that allows to fly the UAV without hardware. Figure 2 (a) presents the snapshot of simulation showing the shape of polygon which represents the plume. Figure 2 (b) shows the resultant trajectory of the UAV after the termination of the algorithm. The size of each grid cell is set to $10m \times 10m$. In this simulation the UAV gathers thirty measurements at each cell to identify the plume cell.

Next, we demonstrate the result of the preliminary experiment carried out in Kentland Farm (Figure 3). We placed the tarp in the middle of waypoints in a bounded environment. The resultant path in Figure 3 demonstrates the preliminary result of detecting and mapping the tarp using a pair of UAV and UGV.

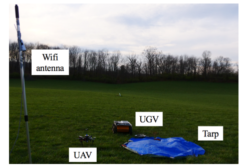


(a) Snapshot of the simulation.

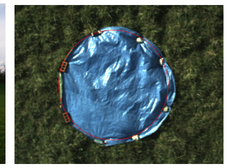


(b) Resultant trajectory of the UAV.

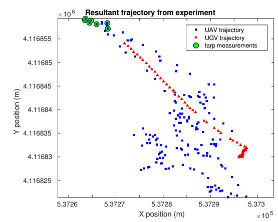
Fig. 2. Result of the software-in-the-loop simulation.



(a) Experimental setup.



(b) Result of the tarp detection (red circle).



(c) Resultant trajectory of the UAV and UGV.

Fig. 3. Result of preliminary experiments (the video for both the simulation and experiment is available at `https://youtu.be/wiOWjr8h8iY`).

IV. Ongoing Work

The immediate future plan for this work is to carry out larger-scale experiments at the actual lake (Claytor Lake, Virginia, USA) in April. We plan to use a dye and USV in place of the tarp and UGV.

References

[1] G. W. Podnar, J. M. Dolan, A. Elfes, S. Stancliff, E. Lin, J. C. Hosler, T. J. Ames, J. Moisan, T. A. Moisan, J. Higinbotham *et al.*, "Operation of robotic science boats using the telesupervised adaptive ocean sensor fleet system," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, 2008, pp. 1061–1068.

[2] M. Dunbabin and L. Marques, "Robots for environmental monitoring: Significant advancements and applications," *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 24–39, 2012.

[3] A. Prorok, M. A. Hsieh, and V. Kumar, "Fast redistribution of a swarm of heterogeneous robots," in *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2016, pp. 249–255.

[4] C. Icking, T. Kamphans, R. Klein, and E. Langetepe, "Exploring an unknown cellular environment." in *EuroCG*, 2000, pp. 140–143.

[5] E. M. Arkin, S. P. Fekete, and J. S. Mitchell, "Approximation algorithms for lawn mowing and milling," *Computational Geometry*, vol. 17, no. 1-2, pp. 25–50, 2000.

[6] A. Kolenderska, A. Kosowski, M. Małafiejski, and P. Żyliński, "An improved strategy for exploring a grid polygon," in *International Colloquium on Structural Information and Communication Complexity*. Springer, 2009, pp. 222–236.

[7] "ROS Wiki," http://wiki.ros.org/, accessed: 2018-03-28.

[8] "OpenCV," https://opencv.org/, accessed: 2018-03-28.

[9] "ArduPilot," http://ardupilot.org/ardupilot/index.html, accessed: 2018-03-28.