Adaptive Planning for Deployment of Micro-Aerial Sensor Swarms

Aveek Purohit, Stefano Carpin, and Pei Zhang *

ABSTRACT

Micro-aerial vehicle (MAV) swarms are a new class of mobile sensor networks with many applications including search and rescue, urban surveillance, radiation monitoring, etc. These sensing applications require autonomously deploying a high number of low-cost, low-complexity MAV sensor nodes at suitable locations in hazardous environments. We propose a collaborative algorithm for resource-constrained MAV nodes to quickly and efficiently deploy at preassigned locations in multi-room scenarios. Through large-scale simulations we show that the proposed technique provides significant benefit over existing autonomous deployment strategies.

1. INTRODUCTION

In sensing applications with hostile, dangerous, or otherwise inaccessible environments (such as urban search and rescue, environmental monitoring, surveillance, etc.), in-situ sensor data is very valuable but manual deployment of sensors is often not feasible.

Micro-aerial vehicle (MAV) swarms are an emerging class of networked mobile systems with widespread applications in such domains. These swarms consist of miniature aerial sensor nodes with limited individual sensing, computing and communication capabilities [13, 7]. Initial work in the operation of MAVs has focused on outdoor or highly instrumented environments that rely on external sensors to control individual devices [2]. However, such centralized sensing approaches are hampered in indoor environments by obstructions (walls, floors, furniture, etc.). At the same time, reliance on infrastructure implies the requirement for a large deployment of support sensors covering all the locations that a MAV may visit [11]. Thus these approaches are only applicable in pre-surveyed locations.

This paper presents an algorithm for the cooperative deployment of swarms of Micro-Aerial sensors in environments not formerly preconditioned for their operation. The key focus behind this networked MAV swarm research is to rely on collaboration to overcome the limitations of individual nodes and efficiently achieve system-wide sensing objectives.

In the proposed approach, the MAV swarm self-establishes a temporary infrastructure of a few landed MAV's acting as radio beacons. Using radio signature or fingerprints from beacon nodes, the algorithm detects intersections in trajectories of exploring mobile MAV nodes. The algorithm combines noisy dead-reckoning measurements from multiple MAV's at the detected intersections to improve the accuracy of the MAVs' location estimates. Most importantly, the algorithm adaptively plans trajectories of MAV nodes according to the certainty of their location estimates – directing movement to improve location estimates when certainty is low, and directing them to follow a map bias when certainty of location estimates is high.

The main contributions of this paper is a combined location estimation and planning algorithm that determines the certainty of location estimates and uses it to adaptively plan node motion.

*A. Purohit and P. Zhang are with Carnegie Mellon University, Silicon Valley Campus, Moffett Field, CA, USA.
S. Carpin is with the School of Engineering, University of California, Merced, CA, USA.

2. OVERVIEW

Potential MAV swarm sensing applications will require mobile sensors to autonomously deploy in multi-room operating environments with no localization infrastructure. In this paper, we address the problem of how a network of mobile sensors can be deployed to pre-determined deployment positions under time and accuracy constraints.

The system begins operation with a swarm of MAV's being introduced into a multi-room connected space through an opening. We make the assumption that a coarse map of the building is available and can be

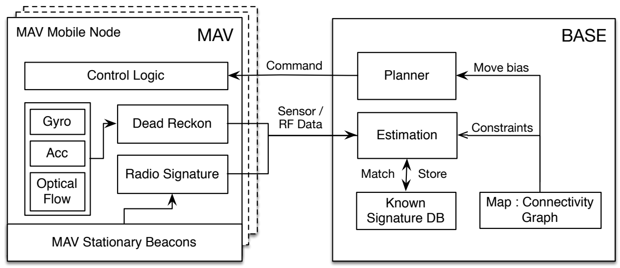


Figure 1: The figure shows architecture of our deployment system. The mobile MAV nodes send dead-reckoning sensor data and radio signatures to a base station. The base station runs our estimation and planning algorithm and issues movement commands to individual MAV nodes.

utilized by domain experts to pre-determine suitable placement of sensors. The system uses the rough map to extract a connectivity graph of the various spaces (rooms) in the deployment environment and determine deployment locations.

The proposed system has 3 major operational phases, setup, estimation and planning (the latter two proceed in conjunction) –

- **Setup:** The system autonomously establishes a transient infrastructure of stationary MAV nodes acting as wireless beacons. These nodes land on being introduced into the area and remain stationary during the deployment process. These nodes use a simple dispersion algorithm [6] that lets them spread out in the environment *without any estimation of their location*.
- **Estimation:** The system then estimates the locations of nodes and guides them to their deployment locations. To realize this, the system first uses dead reckoning sensors such as an optical flow velocity sensor and gyroscope (in our test MAV platform) to get a rough estimate of the motion path of mobile nodes. Second, the system uses radio fingerprints, collected by mobile nodes from the self-established wireless beacons, to determine *rendezvous points*, i.e. points where nodes visit locations already visited by other nodes or by themselves.

  The radio fingerprints are collected in an *online* fashion, i.e., the nodes discover fingerprints as they explore. These fingerprints are sent to the Base and matched with a database of previously discovered signatures. If the signature matches an existing signature in the database (decided by a distance metric), the point is classified as a rendezvous, and a correction can be applied. If the signature does not match any existing signature, it is added to the database as a new entry.

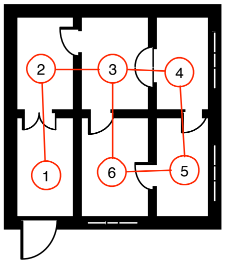
  Finally, the system uses the rendezvous points to combine location estimates from multiple nodes and collaboratively improve location estimates of the entire swarm. Combining estimates is a chicken and egg problem that requires a rendezvous point to estimate and update its own location from visiting mobile nodes, and subsequently, use the updated location to correct the estimates of the visiting mobile nodes. To achieve this, we employ a particle filter based approach. A particle filter [12] is a Bayesian estimation method used to estimate system state based on multiple noisy sensor measurements. We use a particle filter to track the position and orientation of each mobile node. Similarly, we use a particle filter to track the position of each rendezvous point as it is discovered and visited by the MAV nodes. Every visit to a rendezvous point by a mobile node, results in the the mobile node *correcting* the estimates of the particles of the rendezvous point, which in turn *corrects* the estimates of the particles of the mobile node.
- **Planning:** Having estimated locations, the system commands the nodes to follow a path to subsequent deployment positions. However, the quality of the planned path depends greatly on the accuracy of the initial location estimate of nodes. The novel aspect of our system is that it considers the quality of location estimates in planning node paths. The path planner commands node movement such that they increase rendezvous points and potentially improve location estimates when the quality of their estimates is likely to be low. On the other hand, when the location estimates are likely to be more accurate, the planner uses the map to direct them to their designated deployment locations.

Figure 1 shows the architecture of the system. The system deploys **Stationary MAV Nodes** through dispersion that act as wireless beacons. **Mobile MAV Nodes** explore, obtaining dead-reckoning measurements from their on-board sensors and radio RF-signatures from the stationary beacons. The mobile nodes relay this to a **Base**. The Base stores a database of known radio signatures (**Signature DB**) that is used to determine rendezvous in node paths and apply corrections to their dead-reckoning estimates. The corrected location estimates are used by the Base in conjunction with a **Connectivity Graph** (extracted from the coarse map) of the environment to command the subsequent movements of MAV nodes.

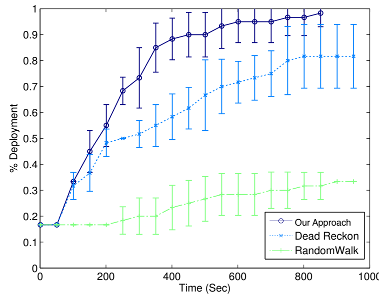
2.1 Adaptive Path Planning

We described how a rendezvous between the paths of nodes can be utilized to improve their location estimates.

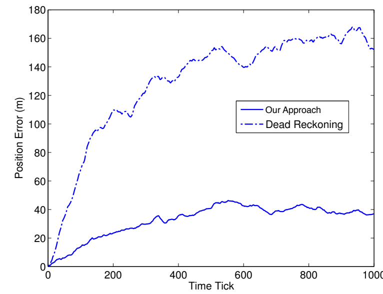
In order to reach the deployment regions, we use the floor plan to produce a graph of connected regions of the environment. Each room or space is a node in the graph and the edges represent the connecting openings between them. The graph enables us to bias the direction of movement of nodes towards predetermined



(2a) The figure shows the $50m \times 50m$ 6-room house floor-plan used in the simulation experiments with the connectivity graph.



(2b) The figure shows the % deployment completed over time (6-room map) for 10 nodes using 3 different strategies - Our approach, dead-reckoning based biased walk, and random walk.



(2c) The figure shows the location error as a function of time using our estimation approach and dead reckoning alone, using 10 nodes in the 6-room map scenario.

deployment regions, if the current location of the node in the map can be reasonably determined. However, due to noisy sensors, the location of individual nodes cannot always be estimated correctly making it difficult to consistently plan correct paths. The system attempts to solve this by operating in two modes –

- **Exploration:** In this mode, the MAV node attempts to seek rendezvous points that can potentially improve the location estimates of the MAV node. This is executed when the quality of location estimates (determined by the entropy of the tracking particle filter distribution) is low.
- **Navigation:** In this mode, the MAV node attempts to follow the direction of the bias from the deployment graph using the estimated location from the DrunkWalk algorithm. This is executed when the quality of location estimates is high.

It is easy to see that the performance of the navigation step depends on the outcome of exploration step. However, the exploration step requires extra use of resources that increase the time of deployment. Therefore, the proposed algorithm seeks to optimize this trade-off by adaptively switching between these two modes.

3. EVALUATION

In this section we evaluate the performance of our system in deploying in multi-room operating environments through simulations using the SensorFly MAV [9] simulator. The evaluation focuses on characterizing the performance of the system in terms of time to complete deployment and average accuracy of deployment in comparison to existing deployment approaches.

3.1 Simulation Environment

We extend a MAV simulation environment [9] for the SensorFly MAV indoor sensor swarm to evaluate our deployment algorithms at scale in a realistic scenario. The simulator supports inclusion of realistic physical arenas, sensors with configurable sensor noise models, MAV nodes with mobility models, indoor radio propagation models, and environment sensing. The simulator allows users to program the logic for actuation of MAV's and implement control and planning algorithms.

For our evaluations we configure the simulator as follows. We assume a multi-room indoor scenario, where nodes are required to autonomously deploy over all rooms. Figure 2a represents a typical indoor 6-room apartment scenario where such systems may be deployed in search and rescue applications. The sensor nodes in the simulation are modeled after the SensorFly [7] MAV platform. Each node has a 802.15.4 radio and dead-reckoning sensors – gyroscope, an optical flow velocity sensor, an ultrasonic altitude measurement sensor. The MAV nodes can turn by a commanded angle and move for a commanded time and velocity. We set the velocity to 0.25 m/s. The velocity of course varies in accordance with the noisiness of the optical flow sensor, that provides feedback to each MAV's control algorithm. The simulation time-step is chosen as $1sec$. The simulation supports estimating received signal strength (RSS). The RSS is computed using shadowing with a path loss exponent of 3, which is an estimate for an indoor single-floor scenario [10].

3.2 Results

We evaluate our system by comparing the percentage of deployment completed in the 6-room scenario as a function of time. We compare our approach to two other deployment strategies that do not require any location infrastructure. They are – (1) **Random Walk:** This is a popular strategy for simple robots with few sensors and has been extensively researched for use in scenarios where no location infrastructure exists [3]. We use this as a baseline for comparison. (2) **Dead-Reckoning with Map Bias:** Dead-reckoning is another infrastructure-free technique used to estimate a node's location in unknown environments [1]. The method uses measurements from motion sensors to estimate the change in position of the node. Having an estimate of location we use the map to bias the direction of the node's movement.

All experiments were performed 25 times and the error bars show the standard deviation of the measured values. Figure 2b shows the percentage of deployment completed as a function of time using our approach, Dead-Reckoning with map bias, and Random Walk. The simulation uses 10 nodes for each strategy in the 6-room map (Figure 2a). The nodes are introduced into room 1 and the objective is to deploy at least one node in each of the 6-rooms of the map. We run the simulation for a time period of 1000 seconds ($\tilde{1}5$ minutes) corresponding to the typical battery life of current generation MAV nodes. The dead-reckoning sensor noise models are set $\sigma = 20\%$ and radio fingerprint accuracy is $1m$.

Our approach is significantly faster at deployment and also manages to complete the deployment before other strategies. Dead-reckoning achieved 80% deployment at the end of the node lifetime, while Random Walk managed 30%. The poor performance of Random Walk is expected in multi-room scenarios with small openings between rooms since the probability of robots making it to subsequent rooms is low. The better planning and location accuracy of our approach enables it to perform better than dead-reckoning, especially later in the deployment when dead-reckoning error becomes extremely large.

Figure 2c shows the location error as a function of time using our proposed estimation approach and dead reckoning alone. Our approach reduces the location by more than $3\times$.

4. RELATED WORK

Howard et al. [4] present techniques for mobile sensor network deployment in an unknown environment. Their approach constructs fields such that each node is repelled by both obstacles and by other nodes, enabling the network to spread itself throughout the environment. Similarly, Batalin et al. [5] present a deployment algorithm for robot teams without access to maps or location. The robots are assumed to be equipped with vision sensors and range finders and select a direction away from all their immediate sensed neighbors and move in that direction. The algorithm does not allow nodes to be deployed at designated locations. The domain experts have no control over the emergent deployment locations of the nodes.

The problem addressed in this paper can be seen as an instance of the Simultaneous Localization And Mapping (SLAM) problem that has been extensively studied in robotics [12]. These approaches, however, have been mostly applied to solve instance of the SLAM problem where mobile agents are equipped with sensors returning distances (e.g., laser range finders, or sonars) or cameras (either monocular or stereo). Purohit et al. [8] present a system for infrastructure-free single room *sweep* coverage with MAV sensor swarms. Their approach however does not support deploying nodes to pre-assigned destinations.

To the best of our knowledge, this paper presents the first attempt to solve a SLAM problem using a swarm of MAVs that combines location estimation and planning to improve the speed and accuracy of deployment.

5. CONCLUSION

This paper presents a system for collaborative deployment of resource-constrained MAV sensing swarms to quickly and efficiently deploy at preassigned locations. The system uses collaboration between nodes of the swarm to overcome the sensing and computational limitations of MAV nodes, and challenging operating environments. Simulations show that the proposed approach performs up to $3\times$ better than existing deployment strategies.

Acknowledgments

Stefano Carpin is partially supported by the Army Research Lab under contract SUPP-13-6-CNC. Pei Zhang are partially supported by National Science foundation under awards CNS-1135874 and CNS-1149611, and by DARPA under grant 1080247-D11AP00265-DOI-ZHANG. Any opinions, findings, and conclusions or recommendations expressed in these materials are those of the authors and should not be interpreted as representing the official policies, either expressly or implied, of the funding agencies of the U.S. Government

6. REFERENCES

[1] J. Borenstein, L. Feng, and H. R. Everett. *Navigating Mobile Robots: Systems and Techniques*. A. K. Peters, Ltd., Natick, MA, USA, 1996.

[2] K. Dantu, B. Kate, J. Waterman, P. Bailis, and M. Welsh. Programming micro-aerial vehicle swarms with karma. In *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, SenSys '11, pages 121–134, New York, NY, USA, 2011. ACM.

[3] D. W. Gage. Randomized Search Strategies With Imperfect Sensors. In *SPIE Mobile Robors VIII*, pages 270–279, Boston, 1993.

[4] A. Howard, M. J. Matarić, and G. S. Sukhatme. An Incremental Self-Deployment Algorithm for Mobile Sensor Networks. *Autonomous Robots*, 13(2):113–126, Sept. 2002.

[5] G. S. S. Maxim A. Batalin. Spreading Out: A Local Approach to Multi-robot Coverage. *Proceedings of the 6th International Symposium on Distributed Autonomous Robotics System*, 2002.

[6] R. Morlok and M. Gini. Dispersing robots in an unknown environment. In *7th International Symposium on Distributed Autonomous Robotic Systems (DARS)*, 2004.

[7] A. Purohit, Z. Sun, F. Mokaya, and P. Zhang. SensorFly: Controlled-mobile sensing platform for indoor emergency response applications. In *In Proceeding of the 10th International Conference on Information Processing in Sensor Networks (IPSN)*, pages 223–234, 2011.

[8] A. Purohit, Z. Sun, and P. Zhang. Sugarmap: Location-less coverage for micro-aerial sensing swarms. In *Proceedings of the 12th International Conference on Information Processing in Sensor Networks*, IPSN '13, pages 253–264, New York, NY, USA, 2013. ACM.

[9] A. Purohit and P. Zhang. Controlled-mobile sensing simulator for indoor fire monitoring. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International*, pages 1124–1129, 2011.

[10] D. Rutledge. Investigation of indoor radio channels from 2.4 GHz to 24 GHz. In *IEEE Antennas and Propagation Society International Symposium. Digest. Held in conjunction with: USNC/CNC/URSI North American Radio Sci. Meeting (Cat. No.03CH37450)*, volume 2, pages 134–137. IEEE.

[11] S. Shen, N. Michael, and V. Kumar. Vision-based autonomous navigation in complex environments with a quadrotor. In *iros*, Tokyo, Japan, nov 2013. Submitted.

[12] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, 2006.

[13] R. J. Wood. The First Takeoff of a Biologically Inspired At-Scale Robotic Insect. *IEEE transactions on robotics*, 24(2):341–347, 2008.