Sensor Planning for a Symbiotic UAV and UGV System for Precision Agriculture

Pratap Tokekar, Joshua Vander Hook, David Mulla and Volkan Isler

***Abstract*—We study two new informative path planning problems motivated by the use of aerial and ground robots in precision agriculture. The first problem, termed Sampling Traveling Salesperson Problem with Neighborhoods (SAMPLINGTSPN), is motivated by scenarios where Unmanned Ground Vehicles (UGVs) are used to obtain time-consuming soil measurements. The input in SAMPLINGTSPN is a set of possibly overlapping disks. The objective is to choose a sampling location in each disk, and a tour to visit the set of sampling locations so as to minimize the sum of the travel and measurement times. The second problem concerns obtaining the maximum number of aerial measurements using an Unmanned Aerial Vehicle (UAV) with limited energy. We study the scenario where the two types of robots form a symbiotic system—the UAV lands on the UGV, and the UGV transports the UAV between deployment locations.**

This paper makes the following contributions: First, we present an $\mathcal{O}\left(\frac{r_{\max}}{r_{\min}}\right)$ approximation algorithm for SAMPLINGTSPN, where $r_{\min}$ and $r_{\max}$ are the minimum and maximum radii of input disks. Second, we show how to model the UAV planning problem using a metric graph and formulate an orienteering instance to which a known approximation algorithm can be applied. Third, we apply the two algorithms to the problem of obtaining ground and aerial measurements in order to accurately estimate a nitrogen map of a plot. Along with theoretical results, we present results from simulations conducted using real soil data and preliminary field experiments with the UAV.

I. INTRODUCTION

Informative path planning is a fundamental problem for robotic sensing systems. The goal of informative path planning is to compute paths for robots acting as mobile sensors in order to accurately estimate some underlying phenomenon, typically a spatio-temporal field [2], [3]. In many cases, nearby points in the field are correlated [4] which enables efficient informative path planning [5]–[7]. In this paper, we introduce and study new informative path planning problems that are motivated by the use of robots in precision agriculture.

Precision agriculture is a data-driven technique to estimate and predict the health of crops in a farm, and use this information to design targeted fertilizer treatment plans [8]. Precision agriculture can improve crop productivity and farm profits through better management of farm inputs, leading to higher environmental quality [9]. By measuring soil nitrogen levels across a farm and applying the right level of nitrogen at the right time and place, it is possible to reduce fertilizer usage by 25% without affecting corn yield [10].

A key component of precision agriculture is data collection. We are building a robotic data collection system with small, low-cost Unmanned Aerial Vehicles (UAVs) and Unmanned Ground Vehicles (UGVs) working together. Ground robots are capable of traveling long distances, carrying large loads and measuring soil data but cannot obtain aerial imagery. Small aerial vehicles can take images from a low altitude but have limited battery life and cannot obtain soil measurements. Our system will combine the strengths of ground and aerial robots to provide on-demand sensing capabilities.

Without any prior information, data collection reduces to covering the entire field with, for example, a boustrophedon path [11]. If some prior information is available, then the robot can choose key, informative locations to obtain the measurements from. Furthermore, if nearby points are correlated the robot need not visit a location exactly, but instead visit some point in its neighborhood.

Obtaining soil measurements with the UGV is likely to be time-consuming. We can reduce the total time by combining measurement locations of nearby points with overlapping neighborhoods. This leads to a novel variant of the classical Euclidean Traveling Salesperson Problem (TSP) which we term as the *Sampling Traveling Salesperson Problem with Neighborhoods* (SAMPLINGTSPN). The input to SAMPLINGTSPN is given by a set of disks in the plane, not all of the same radius. The output must be a tour for the robot that obtains a measurement in each disk. The total time of the tour is given by the sum of the traveling time and the measurement time. SAMPLINGTSPN is NP-hard as it generalizes the NP-hard Euclidean TSP. Our main contribution is a $\mathcal{O}(\frac{r_{\max}}{r_{\min}})$ approximation algorithm for SAMPLINGTSPN, where $r_{\min}$ and $r_{\max}$ are the smallest and largest radii of the input disks.

We also study the corresponding planning problem for the UAV. Unlike soil measurements, aerial measurements, i.e., multi-spectral aerial images, can be obtained instantaneously. However, small UAVs have a limited battery life. Visiting all input points in a large plot may not be feasible. Hence, we study the problem of maximizing the number of points visited subject to the maximum battery life. The general problem of visiting the most number of points subject to a budget is called orienteering [12]. Instead of using the UAV alone, we consider the scenario where the UAV can land on the UGV, and use the UGV to travel to the next take-off locations. We show how to plan the motion of this symbiotic UAV+UGV system on a metric graph, which allows us to apply orienteering algorithms.

Both planning problems we study are of independent in-

P. Tokekar is with the Department of Electrical & Computer Engineering, Virginia Tech, U.S.A. tokekar at vt.edu

J. Vander Hook is with the NASA Jet Propulsion Laboratory in Pasadena, California, U.S.A. vanderhook at jpl.nasa.gov

V. Isler is with the Department of Computer Science & Engineering, University of Minnesota, U.S.A. isler at cs.umn.edu

D. Mulla is with the Deparment of Soil, Water and Climate, University of Minnesota, U.S.A. mulla003 at umn.edu

P. Tokekar is the corresponding author.

A preliminary version of this paper was first presented in [1] with a SAMPLINGTSPN algorithm restricted to the special case of equal radius disks, and without the experiments reported in Section VII.

terest. We also verify their utility in our motivating precision agriculture application. Specifically, we show how the SAMPLINGTSPN and symbiotic UAV+UGV planning problems can be used to accurately estimate nitrogen maps represented as a Gaussian Process. We perform simulations using real data collected from an agriculture plot and empirically evaluate the two algorithms. We also present results from preliminary field experiments for the UAV.

The rest of the paper is organized as follows. We begin by presenting the related work in Section II. We describe the motivating application of precision agriculture and formulate the two path planning problems in Section III. In Section IV we introduce the SAMPLINGTSPN problem for the UGV, and present the $\mathcal{O}(\frac{r_{\max}}{r_{\min}})$ approximation algorithm. In Section V, we show how to plan for the symbiotic UAV+UGV paths for obtaining aerial measurements. Simulation results based on field data are presented in Section VI, and preliminary field experiments are presented in Section VII. We finally conclude with a discussion of future work in Section VIII.

II. RELATED WORK

The work presented in this paper is related to existing literature in various areas. We discuss each in turn below.

A. Informative Path Planning

The problem of designing sensor trajectories and the related problem of selecting sensor locations has recently received much attention. Low et al. [13] presented a control law to minimize the probability of misclassification in a Gaussian Process map. The authors enforce measurements to be taken continuously, and sensors to only move along a 4-connected grid. Zhang and Sukhatme [14] presented an adaptive search algorithm for finding the optimal sensor path to estimate a scalar field. Song et al. [15] presented an algorithm to localize multiple radio sources using a mobile robot. They presented upper bounds on the time required to localize the sources up to a desired probability. Otte et al. [16] studied the problem of navigating to a global goal while foraging for interesting locations along the way. They analyzed two greedy strategies to bound the total distance traveled and verified the bounds through simulations.

In these works, sensing is considered equivalent to visiting some neighborhood of a location. That is, measurement time is zero and the total time of a trajectory is assumed to be proportional to the path length. Time spent for sensing becomes important when considering persistent monitoring of spatio-temporal fields. For example, Smith et al. [17] presented an algorithm for continuously optimizing the speeds of the robots traveling along given paths in order to frequently observe a spatio-temporal field. Yu et al. [18] studied the discrete version where the robots must observe transient events that arrive at discrete locations with unknown arrival times (but known statistics). The goal is to choose the time spent at a given location. They presented an algorithm that computes the optimal (in expection) wait times, if the order to visit the locations is given. They also presented an $(1+\epsilon)$–approximation algorithm for computing the optimal order. In SAMPLINGTSPN, we assume that the time for each measurement is fixed. However, our formulation allows the robot to combine measurements of nearby locations without any additional cost of measurement.

Krause et al. proposed mutual information as a measure of uncertainty [4]. An algorithm to place sensing locations was given which can closely approximate the optimal increase in mutual information. The work was extended to mobile sensor routing in [2], and multiple robots in [19]. This work considers classification uncertainty directly and not mutual information, and as such these results are not directly applicable.

B. TSPN and Data Mules

The SAMPLINGTSPN problem generalizes TSP and its variant, Traveling Salesperson with Neighborhoods (TSPN). In TSPN, we are given a set of geometric neighborhoods, and the objective is to find the shortest tour that visits at least one point in each neighborhood. Dumitrescu and Mitchell [20] presented an 11.15-approximation algorithm for TSPN when the neighborhoods are possibly-overlapping unit disks centered at each site. The main difference in SAMPLINGTSPN and TSPN is that our cost is not just the traveling time of the tour, but also the total time taken for obtaining soil measurements. Finding a minimum length/time path does not necessarily ensure that the robot takes fewer soil measurements, and the cost for the UGV tour is not necessarily minimized.

Recently, there has been some work on TSPN with stochastic neighborhoods. Kamousi and Suri [21] presented offline and online approximation algorithms when the centers of the disks are known, but the radii are *i.i.d.* random variables. In contrast, we show how to compute the disk radii when the underlying field is a spatial map represented as a Gaussian Process. This allows us to formulate SAMPLINGTSPN with known and given disk radii.

The most closely related application of TSPN in robotics is the data muling problem. Bhadauria et al. [22] studied the problem of computing a minimum time data collection tour for k robots tasked with wirelessly collecting data from deployed sensors by visiting a point in the sensor's communication range. In their model, robots spend time for both traveling and downloading data from robots. Tekdas et al. [23] extended this model to the case where the communication range consists of two disks centered at the sensor and the inner ring requires less download time than the outer. In these problems, the robot has to separately query each sensor, and thus the total measurement time always equals the number of sensors. On the other hand, in SAMPLINGTSPN the robot can combine soil measurements for multiple points and reduce measurement time by sampling the intersection of their neighborhoods. Hollinger et al. [24] do allow the robot to communicate with multiple sensors from the same location without paying a separate measurement cost. However, their main result is an exponential-time optimal solution and polynomial time heuristics for a more general formulation.

A key distinction in data muling is that the robot does not need to stop while downloading data from the sensors. For example, Sugihara and Gupta [25] show how to optimize the speed profile so as to guarantee that the data mule spends sufficient time to download the data in each sensor's neighborhood without stopping. Thus, the sum of travel time and measurement time as the objective may not be appropriate in data muling, unlike in SAMPLINGTSPN.

In [26], Alt et al. studied the problem of covering a given set of points with k radio antennas with circular ranges, where the algorithm has to choose the center and radius r_i for each circle. They consider a cost function which is a weighted sum of the length of the tour and the sum of r_i^α for each disk (α models the transmission power for the antennas). The main difference between this problem formulation and ours is that we do not require the number of samples, i.e., k, to be fixed. Instead our formulation penalizes higher k in the cost function.

C. Orienteering

The problem of maximizing the number of points visited by the UAV subject to a battery lifetime constraint is modeled as an orienteering problem. Blum et al. [12] presented a 4-approximation to the orienteering problem for complete graphs with metric edges. We show in Section V how the problem of selecting the most input points can be solved as an orienteering problem by constructing a complete graph with metric edges.

D. Heterogeneous Systems

Recently, there has been a significant interest in developing cooperative aerial and ground/surface/underwater robot systems. Grocholsky et al. [27] described a system with coordinating aerial and ground vehicles for the application of detecting and locating targets. Sujit and Saripalli [28] studied the problem of exploring an area to detect targets using an UAV and inspecting the targets with Autonomous Underwater Vehicles (AUV). The authors compared in simulations three strategies to address the trade-off between quickly exploring the environment for all targets, and minimizing the latency between detection with UAVs and inspection with AUVs. Tanner [29] presented control laws for the UGVs to form a grid of sensors and UAVs to fly in a formation over the grid, such that a target moving on the ground can be detected if it moves from one grid cell to the other. The main difference between existing literature and our work is that we explicitly consider that the UAV can be carried between takeoff locations by the UGV in the sensor planning phase. The resulting plan found by our algorithm may consist of multiple deployments for the UAV, which increases its coverage with limited battery.

E. Robotics in Agriculture

UGVs and UAVs are increasingly being used in agriculture [30]. Typical applications include automated harvesting [31], spraying [32], and yield estimation [33], [34]. A major emphasis in these works is to improve the sensing capabilities and autonomous navigation within farms [35]. These works are complementary to our work, where the focus is on active, informative path planning.

III. MOTIVATION AND PROBLEM FORMULATION

The motivating problem in this paper is that of estimating a Nitrogen (N) level map, a proxy for crop health. The goal is to label each point based on the N levels (e.g., "high-N", "medium-N", "low-N"). Depending on the prior information, some points have a high probability of being misclassified. The problem we study is that of planning informative tours for the robots to obtain measurements near points that are potentially mislabeled.

In this section, we show how to identify points whose probability of being mislabeled, based on a prior nitrogen map, is above a threshold. We term these as Potentially Mislabeled (PML) points. The set of PML points thus identified will be the input to the planning algorithms.

Our approach can be summarized as shown in Algorithm 1.

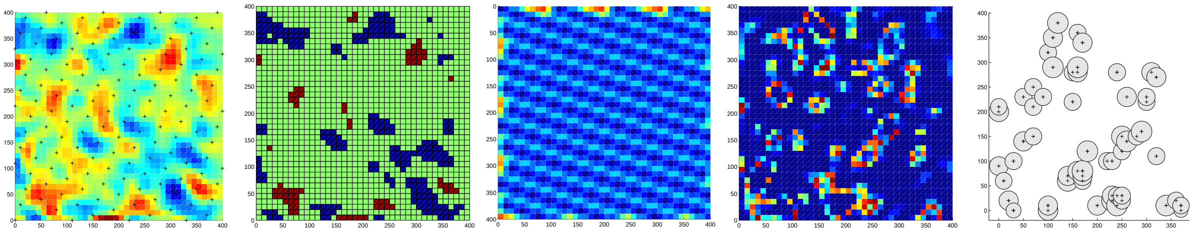
```
Algorithm 1: Symbiotic Path Planning
input : A Map of Soil N Level Labels and Desired
        Label Uncertainty
output: A UGV+UAV Tour of Measurement Locations
1 Identify the set of PML points, $\mathcal{X}_{pml}$, from a given prior
  nitrogen map (Section III-A).
2 Compute a disk centered at each such point, such that an
  expected measurement within this disk is sufficient to
  reduce the mislabeling probability below the user-defined
  threshold (Section III-A).
3 Find (an approximation to) the largest subset of the PML
  points, $\mathcal{X}_s \subseteq \mathcal{X}_{pml}$, that can be visited by the UAV using
  the symbiotic UAV+UGV system, subject to its
  maximum battery lifetime constraint (Section V).
4 Compute the UGV tour to obtain ground measurements
  for each PML point in $\mathcal{X}_s$ (Section IV).
```

A. Finding Potentially Mislabeled Points

Our operating environment is a farm plot discretized into a set of points $\mathcal{X} = \{x_1, x_2, \cdots, x_n\}$. We want to estimate the level of Nitrogen (N) at each point in $\mathcal{X}$ by combining ground and aerial measurements. We use Gaussian Process regression to estimate the N levels using the two types of measurements [36].

Previously obtained measurements are used to build a prior N level map. For each point we associate a most likely estimate as $N(x_i)$, with variance of the estimate given by $\sigma^2(x_i)$ (Figure 1). In general, we are given a set of labels, and each label L_i is specified by a minimum and maximum N level, L_i^-, L_i^+ respectively. For clarity, let the task be to classify each point into three labels: "low-N", "medium-N", and "high-N".

Since we do not have access to the true N levels and instead have a distribution $N(x_i)$, we associate with each label a probability of being correct. We define $\mathrm{P}_{lj}(x_i)$ as the probability that the label j for point x_i is correct $\mathrm{P}_{lj}(x_i) = \mathrm{P}(L_j^- \leq N(x_i) < L_j^+)$. Labels can then be assigned to points based on which is most likely to be correct, given the estimates of N levels at each point. We use the shorter notation $\mathrm{P}_l(x_i)$ to denote the probability of the most likely label.



(a) (b) (c) (d) (e)

Fig. 1. A generated random field using the GP parameters learned from the soil dataset (described in Section VI-A). In all figures, blue indicates low values, red indicates high values, and green is intermediate values. (a) The ground-truth samples of a soil health map obtained at locations marked by a cross, along with the GP regression. (b) The data partitioned into three labels. (c) The variance of the sampling. The variance has a regular pattern, since the samples were obtained along a grid. (d) The mislabel probability. Note mislabel probability is high in many places, even though the variance is roughly uniform and low since the mislabel probability also depends on the value $N(x)$. (e) The points at which the labeling certainty is below P_d, and the corresponding ranges described in Equation 9.

We define PML points as all points in $\mathcal{X}$ for which the probability of the most-likely label being incorrect is below a user-desired value $\mathrm{P}_d \in (0, 1)$.

$$\mathcal{X}_{pml} = \{x_i \in \mathcal{X} : p_{\text{mislabeled}}(x_i) \leq \mathrm{P}_d\}. \tag{1}$$

Our goal is to increase the probability of the label being correct by taking soil and aerial measurements near the PML points.

The previous equation expresses an upper bound on the probability that $N(x_i)$ is below the minimum value of the current label, $L^-(x_i)$, or above the maximum, $L^+(x_i)$. Let $\Phi(a)$ denote the Gaussian cumulative distribution function. Then we have,

$$p_{\text{mislabeled}}(x_i) \leq \mathrm{P}_d \tag{2}$$

$$\therefore\ \Phi\left(\frac{L^-(x_i) - N(x_i)}{\sigma(x_i)}\right) + 1 - \Phi\left(\frac{L^+(x_i) - N(x_i)}{\sigma(x_i)}\right) \leq \mathrm{P}_d \tag{3}$$

Taking measurements near x_i will reduce $\sigma(x_i)$ due to the spatial correlation of the N values. For any value of $N(x_i)$, there exists a corresponding $\sigma(x_i)$ such that Equation 3 is satisfied.

First, we define the constant $\Delta(x_i)$ for each PML point,

$$\Delta(x_i) = \min\left(|L^+(x_i) - N(x_i)|, |L^-(x_i) - N(x_i)|\right) \tag{4}$$

Now Equation 3 can be expressed more conveniently as,

$$2 \cdot \Phi\left(\frac{-\Delta(x_i)}{\sigma(x_i)}\right) \leq p_{\text{mislabeled}}(x_i) \leq \mathrm{P}_d \tag{5}$$

Rearranging the previous equation yields the desired value for $\sigma(x_i)$ as,

$$\frac{-\Delta x}{\Phi^{-1}\left(\frac{\mathrm{P}_d}{2}\right)} \geq \sigma(x_i) \tag{6}$$

We will use the shorthand σ_d for the left hand side of Equation 6 since σ_d can be calculated from prior data and can be treated as a constant. For each point, there will be a different σ_d depending on the exact value of $N(x_i)$, and the current most likely label $L(x_i)$.

Let the measurement location be denoted z, and the sensor noise of the measurement be σ_s. The correlation between the N levels at z and x_i is modeled by the Gaussian Process equations [36]. Thus the *new* variance at x_i, conditioned on the measurement at point z, satisfies,

$$\begin{aligned}\sigma^2(x_i|z) = &\sigma^2(x_i) \\ &- K(x_i, z)[K(z, z) + \sigma_s^2]^{-1} K^T(x_i, z)\end{aligned} \tag{7}$$

The function $K(\cdot, \cdot)$ is the *covariance* or *kernel* function of the Gaussian Process [36]. We fix $K(\cdot, \cdot)$ to be the *squared exponential* function, which is commonly used in precision agriculture [37].

Recall from Equation 6, $\sigma^2(x_i|z)$ should be no greater than σ_d^2. Given the measurement location z, Equation 7 simplifies as follows,

$$\sigma_d^2 - \sigma^2(x_i) \geq -\sigma_f^4(\sigma_f^2 + \sigma_s^2)^{-1} \exp(-\frac{1}{2l^2}||x_i - z||^2) \tag{8}$$

σ_f and l are the *hyperparameters* of the covariance function, which are previously learned from the data.

After further rearrangement and taking the natural log of both sides,

$$||x_i - z||^2 \leq -2l^2 \log[(\sigma^2(x_i) - \sigma_d^2)(\sigma_f^2 + \sigma_s^2)\sigma_f^{-4}]. \tag{9}$$

Denote the right hand side of Equation 9 by r_i. Thus, for every PML point $x_i \in \mathcal{X}_{pml}$ (i.e., points where N estimates do not satisfy Equation 6), we can find a disk of radius r_i centered at x_i. A sample obtained inside this disk will yield sufficiently small variance on $N(x_i)$ to determine the proper label with probability higher than P_d. An example of a field, the field labels, and the points with high mislabeling probability are shown in Figure 1.

In the next sections, we show how to plan for the ground and aerial measurements where the input is the set of PML points and their corresponding disks. We start with the case of ground measurements.

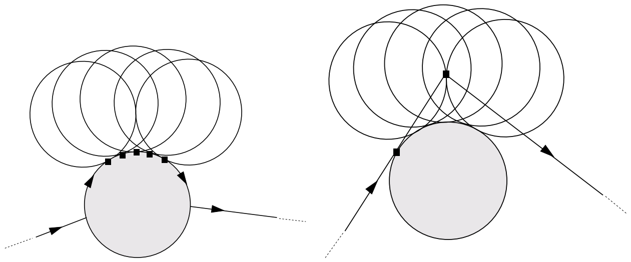
IV. UGV Planning: Sampling TSPN

In our motivating application, the UGV must obtain at least one soil measurement in the disk corresponding to each PML point (as described in Section III-A). Measurements for overlapping disks can be combined. Obtaining a soil measurement is likely to be time consuming. Therefore, instead of minimizing only the travel time, we seek to optimize the sum of the travel and measurement time. Motivated by this scenario, we formulate the following path planning problem, termed SAMPLINGTSPN.

The input to SAMPLINGTSPN is a set of disks specified as $X = \{(x_1, r_1), \ldots, (x_n, r_n)\}$ where x_i is the center of disk i and r_i is its radius. Let $r_{\max}$ and $r_{\min}$ be the maximum and minimum radii. Imagine a robot which can travel at unit speed and obtain a sample in C_g units of time. The objective in SAMPLINGTSPN is:

Given X, find a tour τ of N distinct sample locations to minimize the cost len$(\tau) + C_g \cdot N$ such that each disk contains a sample location.

SAMPLINGTSPN generalizes TSPN with disk neighborhoods. The objective in TSPN is to minimize only the length of the tour. A natural strategy for finding a SAMPLINGTSPN tour would be to first find a TSPN tour, and then choose sampling locations on this tour. However, this approach may lead to bad solutions (see Figure 2 and Section IV-C). Instead, we present an algorithm which finds a tour whose total time is at most $\mathcal{O}(\frac{r_{\max}}{r_{\min}})$ of the optimal time.


(a)

(b)

Fig. 2. Solving SAMPLINGTSPN by first finding a TSPN tour, and then choosing sampling locations on this tour can lead to bad results. (a) The TSPN tour presented in [20] visits all the disks by touring the circumference of each disk in the Maximal Independent Set (one shown shaded). This tour will be forced to take a separate measurement for each outer disk and thus have $O(n)$ measurement locations. (b) In general, we are not forced to move along the circumference and can visit a smaller number of locations where the disks overlap.

We proceed as follows: To introduce the underlying ideas and the necessary background, we first present a simpler algorithm whose approximation ratio is $\mathcal{O}(\frac{r_{\max}^2}{r_{\min}^2})$. We then present a more involved algorithm which improves the approximation factor to $\mathcal{O}(\frac{r_{\max}}{r_{\min}})$. The main challenge in the analysis of both algorithms is to compute lower-bounds on the performance of the optimal solution and to bound the cost of our algorithms with respect to these lower-bounds.

A. An $\mathcal{O}(\frac{r_{\max}^2}{r_{\min}^2})$-approximation

The first algorithm, ALG1, contains the following stages:

- Stage 0: Let $\bar{X}$ be the set of all disks in X with the radius of each disk set to $r_{\max}$.
- Stage 1: Compute $I = MIS(\bar{X})$
- Stage 2: For each disk $d \in I$, place a grid co-centered with d and with dimensions $6r_{\max} \times 6r_{\max}$ and with resolution $r_{\min}/\sqrt{2}$
- Stage 3: Output a TSP tour of all grid points produced in Stage 2.

In Stage 0, $\bar{X}$ is a modified version of the input where the radius of each disk in the input is set to $r_{\max}$. That is, $\bar{X} = \{(x_1, r_{\max}), \ldots, (x_n, r_{\max})\}$. The function MIS computes a set of maximally independent set of disks: the disks in I are mutually non-intersecting (independent) and any disk in $\bar{X} \setminus I$ intersects with some disks in $\bar{X}$ (maximal). The set I can be computed by a simple greedy procedure: choose an arbitrary disk d from $\bar{X}$, add it to I, remove all disks in $\bar{X}$ which intersect d, and repeat the procedure until no such d exists. Computing the TSP tour in Stage 3 is NP-complete. However, for any ϵ, a $(1+\epsilon)$-approximation can be computed in $\mathcal{O}(n(\log n)^{\mathcal{O}(\epsilon^{-1})})$ using [38].

It can be easily verified that ALG1 will take a sample from every disk in X: The center x_i of every disk in X must be within distance $3r_{\max}$ of the center of some disk in I (otherwise I would not be maximal). The grid resolution ensures that even the smallest disk contains a grid point.

We now bound the performance of ALG1. To do this, we establish lower-bounds on the cost of the optimal algorithm, OPT. First, we focus on the number of samples.

Claim 1: OPT must take at least $|I|$ measurements.

The claim is easy to see because the disks in I are disjoint and their radii are set to $r_{\max}$.

Next, we focus on T^*, the travel cost of OPT. We observe that OPT must visit each disk in I and hence,

Claim 2: $TSPN(I) \leq T^*$ where $TSPN(I)$ is the optimal TSPN tour.

We now bound the cost of ALG1 with respect to these lower-bounds.

For the sampling cost, for each disk $d \in I$, ALG takes $\mathcal{O}(\frac{r_{\max}^2}{r_{\min}^2})$ measurements given by the size of the grid. Together with Claim 1, the measurement cost is within $\mathcal{O}(\frac{r_{\max}^2}{r_{\min}^2})$.

To bound the travel cost, we proceed as follows: we show that there exists a tour T which visits all grid points whose cost is within a factor of $\mathcal{O}(\frac{r_{\max}}{r_{\min}})$ of $TSPN(I)$. Hence, the optimal TSP tour visiting the grid points in Stage 3, must also be within this factor. Combined with Claim 2, we obtain a bound on the tour cost.

To obtain tour T, we simply start with $TSPN(I)$. The new tour T follows $TSPN(I)$ to visit each disk. Whenever a disk $d \in I$ is visited, it adds a detour to visit all associated grid points. A simple way of doing this is to move to a corner of the grid from the center and to scan each column in order (Figure 5). The added cost of this detour per disk is less than $c \cdot (r_{\max} + 1) \times \frac{r_{\max}}{r_{\min}}$, where c is a constant.

So far we established that the tour cost of ALG1 is

$T \leq TSPN(I) + nc\frac{r_{\max}^2}{r_{\min}}$ where the second term on the right is the total cost of the detours.

In order bound the cost of the detour with respect to TSPN, we adapt Theorem 1 from [23] which, in our notation, states that the length of any tour that visits n non-overlapping disks of radius $r_{\max}$ is at least $\frac{n}{2}0.4786r_{\max}$. That is, $TSPN(I) \geq \frac{n}{2}0.4786r_{\max}$. This gives, $nr_{\max} \leq \frac{2}{0.47}TSPN(I)$.

Therefore, we have

$$
\begin{aligned}
T &\leq TSPN(I) + nc\frac{r_{\max}^2}{r_{\min}} \\
&\leq TSPN(I) + c\frac{2}{0.47}TSPN(I)\frac{r_{max}}{r_{\min}} \\
&\leq c'\frac{r_{max}}{r_{\min}}TSPN(I) \leq c'\frac{r_{max}}{r_{\min}}T^*
\end{aligned}
$$

where c' is a constant and T^* is the tour cost of OPT.

To summarize, the travel cost of ALG1 is $\mathcal{O}(\frac{r_{max}}{r_{\min}})$ of OPT, and the sampling cost is within $\mathcal{O}(\frac{r_{\max}^2}{r_{\min}^2})$. Therefore, the cost of ALG1 is within $\mathcal{O}(\frac{r_{\max}^2}{r_{\min}^2})$ of OPT.

In the next section, we present our main contribution which achieves an approximation ratio of $\mathcal{O}(\frac{r_{\max}}{r_{\min}})$. This is obtained by replacing the grid points of Stage 2 of ALG1 with a carefully chosen set of points.

B. An $\mathcal{O}(\frac{r_{\max}}{r_{\min}})$-approximation

Stage 2 of ALG1 consists of $\mathcal{O}(\frac{r_{\max}^2}{r_{\min}^2})$ grid points *per disk in the MIS*. On the other hand, there could possibly be a single location within each MIS disk, where an optimal algorithm can obtain a sample. Our main algorithm, GRIDSAMPLE, chooses a much smaller set of points as sampling locations, instead of all grid points, as follows:

- Stage 1: Create a set system, $(P, \mathcal{R})$, from the arrangement of all disks in X. Compute C = HITTINGSET$(P, \mathcal{R})$.
- Stage 2: Output a TSP tour of C.

In Stage 1, we solve a *hitting set* problem. A hitting set problem is defined for a set system $(P, \mathcal{R})$, where P is a set of points and $\mathcal{R}$ is a collection of subsets of P. The hitting set solution $C \subseteq P$ is the smallest subset of P, such that $C \cap \mathcal{R}_i \neq \emptyset$ for all $\mathcal{R}_i \in \mathcal{R}$.

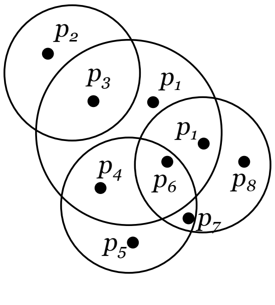


Fig. 3. A geometric hitting set instance. We define a collection of sets $\mathcal{R} = \{\mathcal{R}_i\}$ where each $\mathcal{R}_i$ is the set of disks that intersect with a point p_i. X is the set of all disks. The hitting set solution finds the minimum subset of $\mathcal{R}$ such that its union is X. The minimum hitting set for this example has size 2 (e.g., $\{p_2, p_6\}$).

We formulate a geometric hitting set problem by creating a set system as follows: Consider the arrangement of disks in X in the plane. This arrangement consists of set of faces, as shown in Figure 3. We create a set of points P by placing a point in each face of the arrangement. For each point in $p_i \in P$, let $\mathcal{R}_i$ be the set of disks containing p. Let $\mathcal{R} = \cup \mathcal{R}_i$ be the set of such sets for all points in P. The solution to the geometric hitting set problem finds the minimum number of points in P such that each disk in X has at least one such point in its interior. It is easy to see that the number of samples in an optimal SAMPLINGTSPN solution cannot be less than the size of the optimal hitting set solution.

Finding the optimal hitting set solution is NP-complete in general. However, there exist efficient approximation algorithms, e.g., the $(1+\epsilon)$–approximation in [39], that we can use. Let C be the result from the hitting set algorithm. Hence, measurement cost of the above algorithm is no more $1+\epsilon$ times the optimal SAMPLINGTSPN solution.

Bounding the travel cost of the above algorithm (i.e., bounding the length of the TSP tour of C) is not straightforward. In the case of ALG1 we were able to bound the length of the tour by first finding a TSPN tour and then adding local detours to the grid locations. For analysis purposes, we follow a similar strategy to construct a tour that combines ALG1 and GRIDSAMPLE:

- Stage 0: Let $\bar{X}$ be the set of all disks in X with the radius of each disk set to $r_{\max}$.
- Stage 1: Compute $I = MIS(\bar{X})$. Let T_C be the TSP tour of the centers of all disks in I.
- Stage 2: Create a set system, $(P, \mathcal{R})$, from the arrangement of all disks in X. Compute C = HITTINGSET$(P, \mathcal{R})$.
- Stage 3: For each disk $d \in I$, place a grid co-centered with d and with dimensions $6r_{\max} \times 6r_{\max}$ and with resolution $r_{\min}/\sqrt{2}$.
- Stage 4: Output a TSP tour of all grid points within $2r_{\min}$ of a sampling location in C.

This algorithm, termed ALG2, is a restricted version of GRIDSAMPLE and therefore its cost upper bounds the cost of GRIDSAMPLE. The key component of the analysis is to show that no more than $25|C|$ grid locations will be visited in Stage 4. We further show that visiting these $25|C|$ grid locations is sufficient to guarantee that all the disks are sampled. Consequently, the measurement cost of the tour will be no more than $25(1+\epsilon)$ times the optimal SAMPLINGTSPN algorithm.

The tour produced in Stage 4 visits a subset of the grid points covered by the tour in ALG1. Consequently, the length of the tour will be shorter than the tour in ALG1 which we showed to be no more than $\mathcal{O}(\frac{r_{\max}}{r_{\min}})$ times the optimal SAMPLINGTSPN cost. This yields an overall approximation factor of $\mathcal{O}(\frac{r_{\max}}{r_{\min}})$ for our algorithm.

Let S be the set of grid locations computed in Stage 4. In Lemma 1 we show the correctness of the algorithm by proving that S has at least one sampling location in each input disk (not just the larger disk). In Lemma 2 we upper bound the number of candidate sampling locations, and in Lemma 3 we bound the total distance traveled. Finally, these results are combined to prove the approximation ratio of our algorithm in Theorem 1.

Lemma 1: Let S be the set of all candidate sampling locations in ALG2. Then, for each disk in X, there exists a point in S lying in its interior.

Proof: The set S of sampling locations is computed based on the solution C to the hitting set problem. For any disk in X centered at x, there exists a point $p \in C$ lying in its

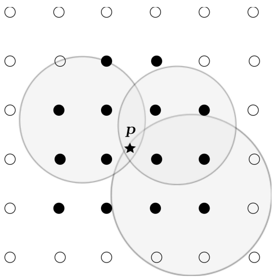


Fig. 4. The point p marked by a star is the output from solving a hitting set problem. We compute grid locations $G(p)$ (filled circles) at a distance of at most $2r_{\min}$ from p. Lemma 1 guarantees that any disk containing p having radius greater than $r_{\min}$, also contains at least one point from $G(p)$.

interior. Since we choose sampling locations from the grid, our algorithm may not be able to choose p. However, we show that by including at most 25 points for each point in the hitting set, we can hit all disks.

$G(p)$ is the set of grid points within $2r_{\min}$ of p. Instead of sampling at p, we sample at some grid point in $G(p)$ (Figure 4). Let D be any disk in X that contains p. We will show that at least one grid point, say $p' \in G(p)$, is also contained in D. Draw a disk D_2 centered at p, with radius $2r_{\min}$. Any disk of radius $r_{\min}$ contained completely within D_2 must also contain at least one point of $G(p)$. Replace D by a smaller disk, say D_1, such that D_1 has a radius $r_{\min}$, D_1 is contained completely within D, and D_1 contains p. D_1 is completely contained within D_2. Hence, D_1 contains some point of $G(p)$.

Next, we will show that this point p' is also included in S. We have one of two cases, either the larger disk centered at x lies in the MIS or not. If it lies in the MIS, then p' is within $3r_{\max}$ of x and we are done. If not, then the larger disk of radius $r_{\max}$ intersects some other larger disk, centered at say x', lying in the MIS. Hence, the distance between x and x' is at most $2r_{\max}$, which implies that p' is at most $3r_{\max}$ away from x'. Hence, in both cases, p' will be in S. ■

Lemma 2: If N^* is the number of samples by an optimal algorithm for the general SAMPLINGTSPN problem and S is the set of grid locations computed in ALG2, then $|S| \leq 25(1+\epsilon)N^*$.

Proof: N^* is the minimum number of points, such that there exists at least one point per disk in X. The set C can be found using any constant-factor approximation for this hitting set problem. For example, using the algorithm in [39], we have $|C| \leq (1+\epsilon)N^*$. For each point in C, we add at most 25 points in S. Hence, $|S| \leq 25|C| \leq 25(1+\epsilon)N^*$. ■

Lemma 3: Let T_{ALG} be the tour constructed by ALG2, and T^* be the tour for the optimal SAMPLINGTSPN algorithm. Then $\text{len}(T_{ALG}) \leq \mathcal{O}\left(\frac{r_{\max}}{r_{\min}}\right) T^*$.

Proof: For ease of notation, in this proof we refer to both a tour and its length by T, and T^* refers to an optimal tour.

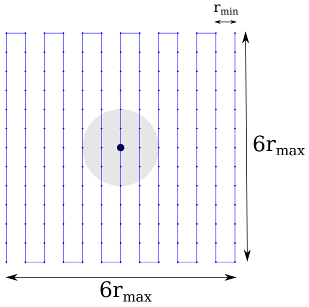


Fig. 5. Bounding the distance traveled by a tour that visits a $6\frac{r_{\max}}{r_{\min}} \times 6\frac{r_{\max}}{r_{\min}}$ grid centered around a disk (shown shaded) in the MIS.

Denote by T_I and T_C the TSPN tour of the MIS and TSP tour of the center of the MIS respectively. Let n be the total number of disks in the MIS. Now

$$T_C^* \leq T_I^* + 2nr_{\max} \tag{10}$$
$$\leq T^* + 2nr_{\max}. \tag{11}$$

The first inequality follows from the fact that a tour of the centers can be constructed by taking a detour of at most $2r_{\max}$ for each disk from the tour of the disks. The second inequality comes from the fact that the optimal tour is also a tour of the disks in the MIS.

T_{ALG} consists of a TSP tour of the centers of the disks in MIS and a tour of the grid locations within $3r_{\max}$ of the center. Using the $(1+\epsilon)$-approximation for the TSP tour [40] we get,

$$T_{ALG}$$
$$\leq (1+\epsilon)T_C^* + n\left(\left(6\frac{r_{\max}}{r_{\min}}\right)(6r_{\max}) + 6r_{\max} + 6\sqrt{2}r_{\max}\right) \tag{12}$$
$$= (1+\epsilon)T_C^* + 36n\frac{r_{\max}}{r_{\min}}r_{\max} + 6n(1+\sqrt{2})r_{\max}$$

In Equation 12, the second term gives the distance traveled along $6\frac{r_{\max}}{r_{\min}}$ vertical grid columns (see Figure 5). The length of each column is $6r_{\max}$. The third term accounts for the total distance traveled horizontally to move from one column to next. The fourth term gives the distance traveled to go from the center of the disk to one corner of the grid and back to the center at the end.

Using Theorem 1 from [23], we know that the length of any tour that visits n non-overlapping disks of radius $r_{\max}$ is at least $\frac{n}{2}0.4786r_{\max}$. That is, $T^* \geq \frac{n}{2}0.4786r_{\max}$. This gives, $nr_{\max} \leq \frac{2}{0.47}T^*$.

Therefore,

$$T_{ALG} \leq (1+\epsilon)\left(T^* + 2nr_{\max}\right) \tag{13}$$
$$+ \left(36\frac{r_{\max}}{r_{\min}} + 6 + 6\sqrt{2}\right) nr_{\max} \tag{14}$$
$$\leq (1+\epsilon)T^* + \left(36\frac{r_{\max}}{r_{\min}} + \mathcal{O}(1)\right) nr_{\max} \tag{15}$$
$$\leq \mathcal{O}\left(\frac{r_{\max}}{r_{\min}}\right) T^* \tag{16}$$

■

Theorem 1: GRIDSAMPLE gives a valid Sampling TSPN tour of cost $\mathcal{O}\left(\frac{r_{\max}}{r_{\min}}\right)$ times that of the optimal tour, where

$r_{\max}$ and $r_{\min}$ are the radii of the largest and the smallest of the input disks, respectively.

Proof: Let C^* be the cost of the optimal algorithm for the general Sampling TSPN problem. Therefore, $C^* \geq T^* + N^* \cdot C_g$, where T^* is the optimal TSPN tour visiting all disks, and N^* is the minimum number of sample locations such that each disk has at least one sample location.

Consider the cost of ALG2,

$$C_{ALG2} = T_{ALG} + |S| \cdot C_g, \tag{17}$$

$$\leq \mathcal{O}\left(\frac{r_{\max}}{r_{\min}}\right) T^* + \mathcal{O}(1) N^* \cdot C_g, \tag{18}$$

$$\leq \mathcal{O}\left(\frac{r_{\max}}{r_{\min}}\right) C^*. \tag{19}$$

where the first inequality comes from Lemmas 2 and 3.

Let C_{ALG} by the cost of GRIDSAMPLE. Both the length of the tour and the number of samples in ALG2 are greater than that of GRIDSAMPLE. Consequently, we get $C_{ALG} = \mathcal{O}\left(\frac{r_{\max}}{r_{\min}}\right) C^*$. ■

The running time in Stage 1 is dominated by the TSP tour, T_C. For any ϵ, T_C can be computed in $\mathcal{O}(n(\log n)^{\mathcal{O}(\epsilon^{-1})})$ using [38]. The geometric hitting set, C, in Stage 2 can be computed in $\mathcal{O}(n^{\mathcal{O}(\epsilon^{-2})})$ time [39]. The total number of grid points considered in Stage 3 is $\mathcal{O}(|C|) = \mathcal{O}(n)$. The final operation is modifying T_C to visit the grid points, which takes at most $\mathcal{O}(n)$ time. Thus, the running time of GRIDSAMPLE is $\mathcal{O}(n(\log n)^{\mathcal{O}(\epsilon^{-1})} + n^{\mathcal{O}(\epsilon^{-2})})$.

C. Comparison of TSPN and GRIDSAMPLE

Figure 2(a) shows an instance where first finding a TSPN tour and then finding sampling locations along the tour can be arbitrarily worse than the optimal SAMPLINGTSPN solution. Consequently, using Theorem 1 we can say that GRIDSAMPLE outperforms TSPN in the worst-case. In order to further study the relative performance of the two algorithms, we carried out simulation studies. The results of the simulation are shown in Figure 6. The sampling cost was set to $t_m = 10$ for these simulations.

We compared the performance of three other algorithms with our GRIDSAMPLE. We compute the optimal hitting set by first imposing a fine grid over all disks, and then enumerating all possible subsets of grid points. The smallest subset of grid points that hit each disk is chosen as the output. We eliminate those grid points that hit the same set of disks. The TSP strategy is the naive baseline strategy of choosing a separate sampling location at the center of each input disk. The two TSPN strategies first compute the TSPN tour of all the input disks. The TSPNGreedy strategy chooses sampling locations along the tour greedily. That is, as soon as the tour encounters a disk not previously covered, we add a new sample at that location and mark the disk covered. The TSPNMin strategy, on the other hand, waits till the robot is about to leave a disk that hasn't previously been covered.

GRIDSAMPLE outperforms all three strategies on an average. GRIDSAMPLE can be further refined by moving each sampling location such that it hits the same set of disks but shortens the path.

In the next section, we show how to plan the corresponding tour for aerial measurements.

V. UAV PLANNING: ORIENTEERING

In this section, we study the problem of obtaining aerial measurements near PML points (Section III-A) using the UAV. The main limitation for the UAV is the limited on-board energy. The UAV may not be able to visit all input PML points. Consequently, we consider the problem of maximizing the number of PML points visited subject to the maximum battery lifetime. We reduce this to the orienteering problem. Let $G(V, E, \pi, w)$ be a graph with weights $w(u, v)$ on edges, and rewards $\pi(v)$ on the vertices. The objective in the orienteering problem is to find a tour of a subset of vertices collecting maximum reward, with the constraint that the sum of weights of edges on the tour is less than a given budget.

Instead of using the UAV alone, we consider the scenario where the UAV and UGV operate together, in order to increase the number of points visited. The UAV can land on the UGV, and the UGV can carry the UAV between deployment locations, thus saving energy. However, the UAV still spends some energy taking-off and landing on the UGV. We show how to model this trade-off for the symbiotic UAV+UGV system as an orienteering instance.

First consider the simpler case of finding the maximum subset of points in a UAV-only system. For simplicity, let the camera footprint be a single point for now. Let the vertices of the graph be the set of PML points and let each vertex have unit reward. We add an edge to G between every pair of points with weight equal to the Euclidean distance between the points. The budget for the UAV equals the battery lifetime minus $2C_a$ to account for the single takeoff and landing. The solution for the orienteering problem for this instance will be a path traversing a set of PML points (with a single landing and take-off location).

Since the edge weights are Euclidean distances, this graph is a complete metric graph. Blum et al. [12] presented a 4-approximation for orienteering problems on undirected metric graphs. Applying this algorithm to the graph we constructed above will yield a UAV tour visiting at least $1/4^{\text{th}}$ of the PML points visited by the optimal algorithm.

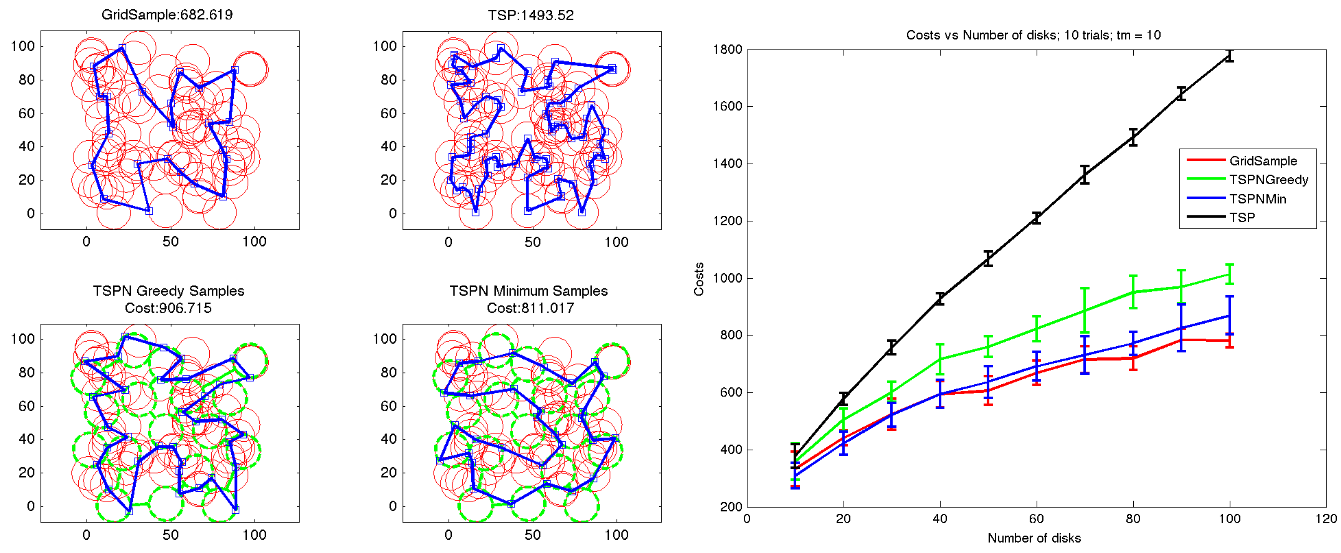
Now consider the case of a UAV+UGV system. The UGV can transport the UAV between two PML locations, without affecting the UAV's battery life. Furthermore, since the UAV carries a camera with a footprint of diameter C, it can sample a point without flying directly over it. Hence, we will also modify the set of vertices. The detailed construction of the input graph for the orienteering problem is given in Algorithm 2.

The following lemma shows that the resulting graph G is a metric graph.

Lemma 4: The graph G constructed in Algorithm 2 is a metric graph.

The proof is presented in the appendix. Since G is a metric graph, we apply the algorithm in [12] to obtain a 4-approximation for this problem.

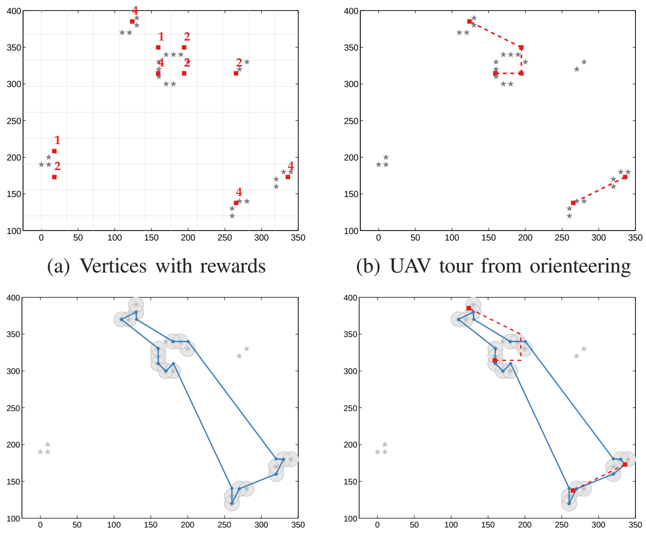
Next, we study the performance of the two planning algorithms through simulations based on field data.



(a)

(b)

Fig. 6. Comparing four solutions for SAMPLINGTSPN: (1) GRIDSAMPLE, the algorithm proposed in this paper; (2) find the TSP tour of the centers of all input disks (baseline); (3) find a TSPN tour using [20] and then choose sampling locations along the tour greedily; and (4) sample as (3) except the sampling locations are chosen optimally. In (1), (2), and (4), the final tour is produced by computing a TSP tour of only the sampling locations produced. The plot shows the average and standard deviation of the costs for 10 randomly generated instances with increasing number of randomly drawn input disks.



(a) Vertices with rewards

(b) UAV tour from orienteering

(c) Final Sampling TSPN

(d) Final UGV+UAV Tours

Fig. 7. Path Planning Algorithm. (a) Square grid of resolution $C/\sqrt{2}$. The reward for visiting each grid point (red square) is the number of PML points (gray star) falling within the grid. (b) UAV tour found using orienteering on the graph of grid points. For this instance, UAV budget was 500 secs out of which 200 secs are spent traveling and 240 secs are spent for the 2 ascents/descents. (c) Sampling TSPN tour (Section IV) for the UGV. (d) Final UGV tour including UAV take-off locations (red squares).

VI. SIMULATIONS

In the previous sections, we showed theoretical bounds on the number of PML points selected and the distance traveled by our algorithm with respect to optimal. We expect the UAV+UGV system to sample more PML points as compared to a UAV-only system with the same battery limitations. We investigate this through simulations based on actual system parameters and real data collected from an agricultural plot.

Algorithm 2: Creating Input Graph G.

```
Create a square grid of resolution $C/\sqrt{2}$ over the plane.
  Each point in $\mathcal{X}_{pml}$ is associated with its nearest grid
  location (Figure 7(a)). Store the number (denoted by
  $\pi(v)$) of PML points associated with a grid location.
Let $V$ be the set of grid vertices with at least one PML
  point associated. For each $v \in V$, let $\pi(v)$ be the number
  of associated PML points (Figure 7(a)).
Build a complete undirected graph $G = \{V, E, \pi, w\}$. For
  each edge between $(u, v) \in V$, add a weight
  $w(u, v) = \min\{d(u, v), 2C_a\}$. This implies there are two
  types of edges between grid points: The UAV can either
  use the UGV to travel paying only for the ascent/descent
  ($2C_A$) or travel directly between points paying the
  distance cost ($d(u, v)$).
```

A. System Description

We present the details of the robotic system we are developing to motivate the choice of our simulation parameters. Our UGV is a Husky A200 by Clearpath Robotics [41]. The UGV has a typical battery life of two hours on a single charge. The operating lifetime can be extended to over six hours easily with additional batteries. The UGV will measure soil organic matter as a proxy for soil N supply to the crop using a Minolta SPAD-502 Chlorophyll meter [42].

Our UAV is a Hexa XL by MikroKopter [43]. This UAV can operate for a maximum of 25 mins (under ideal conditions). Deploying the UAV to approximately 100 meters height gives the camera a 50 meter diameter coverage with a single image. The UAV takes about 2 minutes to ascend/descend this height. The images include multi-spectral information, such as near-infrared reflectance, which is used to estimate the crop N status [44].

B. Modeling

To generate realistic data, we need a generative model of nitrogen levels and realistic values for the sampling noise for both systems. We will briefly discuss how we obtained these from an existing nitrogen remote sensing and soil sampling dataset [44]. The data from [44] consists of 1375 soil measurements taken manually in a 50m by 250m corn field, along with corresponding 1m spatial resolution remote sensing images in the green (G), red (R) and Near Infrared (NIR) portions of the spectrum. The samples were taken along a dense uniform coverage (see Figure 8) and provided the levels of soil Organic Matter (OM). R and NIR are known to be inversely related to crop N status [44].

We used OM as a proxy for the initial quantity of soil N supplied to the crop. We modeled the UGV as taking direct measurements of OM, corrupted by some sensor noise with variance σ_g^2, and the UAV as measuring the Normalized Difference Vegetation Index (NDVI), which is a combination of NIR and R levels [8]. We assume the NDVI levels are corrupted by sensor noise with variance σ_a^2. The noise variances were estimated to be $\sigma_a = 0.31$ and $\sigma_g = .05$ for our dataset, using the following procedure.

To model the spatial patterns of the OM levels, we used GP regression over the set of sample points and OM measurement values. This densely-sampled GP defined the hyperparameters which were used to generate new ground-truth N maps in our simulations. We used the GPML Toolbox [45] for performing the GP regression.

As part of the ground-truth GP regression, we can estimate the sample noise at each point from the data directly (σ_s in Equation 7). We used this value directly as σ_g, since we assumed the robot would have the same sensing capability as the human operators. To estimate σ_a, we calculated the sample covariance between NDVI (from the hand-measured R and NIR levels) and OM (measured directly), yielding the 2×2 matrix,

$$\begin{bmatrix} \sigma^2_{\text{OM}} & \sigma_{\text{OM,NDVI}} \\ \sigma_{\text{NDVI,OM}} & \sigma^2_{\text{NDVI}} \end{bmatrix} \tag{20}$$

From the above equation, we can find the variance in OM given a measurement of NDVI, and use this as the UAV sensor noise as,

$$\sigma_a^2 = \sigma^2_{\text{OM}|\text{NDVI}} = \sigma^2_{\text{NDVI}} - \frac{\sigma^2_{\text{OM,NDVI}}}{\sigma^2_{\text{OM}}} \tag{21}$$

In simulations, we formed a prior estimate of OM levels by down sampling each randomly-generated ground-truth N-map by a factor of 20 and fitting a new GP. We randomly generated 100 N-maps for a 600×400 m field. For each randomly generated prior GP, we found the PML point set as described in Section III using a desired labeling probability of 0.6. The number of PML points in any instance depends on the randomly generated map.

C. Results

We first compare the number of PML points covered by the UAV+UGV system versus an UAV-only system. We use the procedure described in Section V for finding the subset of PML points visited by the UAV-only and the UAV+UGV system, subject to the battery constraint of 25 mins. We used the implementation from the SFO Toolbox [46] for finding an orienteering tour, and the Concorde TSP solver [47] as a subroutine in the Sampling TSPN algorithm implementation.

Figure 9 shows a sample run from the simulations. We observe that the UAV-only tour is constrained to only one part of the field, whereas the UAV+UGV system can obtain measurements from farther away locations. This input instance consisted of 75 potentially mislabeled points, the UAV-only tour covers 38 points whereas the UAV+UGV tour covers 50 points. Figure 10 shows a histogram of the ratio of the points covered by the UAV+UGV and the UAV-only tours for 100 random instances. As expected, the ratio is always greater than 1 as the UAV+UGV system is at least as good as a UAV-only system in terms of the number of points visited. Table I shows the effect of varying the budget on the percentage of input PML points visited.

TABLE I
PERCENTAGE OF INPUT PML POINTS VISITED (AVG. OF 30 INSTANCES).

Budget (sec)	UAV-only	UAV+UGV
500	19	25
1000	36	49
1500	55	72

The UAV+UGV system can cover points that are spread across the field. Intuitively, if the measurements are distributed across the field, we expect the resulting map (after incorporating the measurements) to have fewer mislabeled points than if all measurements are nearby. After calculating the desired UAV/UGV tours, random measurements for the sensors were sampled directly from OM values given the dense (ground truth) GP. We added noise to the measurements using estimated variances $\sigma_a = 0.31$ and $\sigma_g = 0.5$ as described in Section VI-B. These values were then used to update the prior GP, which was then used to find the posterior PML points. We observe the posterior PML points in Figures 9(c) & 9(d). For a fair comparison, we add UGV measurements for each PML point visited by a UAV-only tour, in obtaining the updated N level map.

Figure 10(b) shows a histogram of the ratio of the posterior PML points with a UAV+UGV system and a UAV-only system. Since the number of PML points depend on both the variance, and the estimated $N(x)$ values, occasionally there are instances when the number of posterior PML points with UAV-only system are lesser than that of UAV+UGV system. However, as we can observe in Figure 10(b) the UAV+UGV system often outperforms the UAV only system in terms of number of posterior PML points.

VII. FIELD EXPERIMENTS

We conducted preliminary field experiments with the system we are developing. Our current system capabilities include

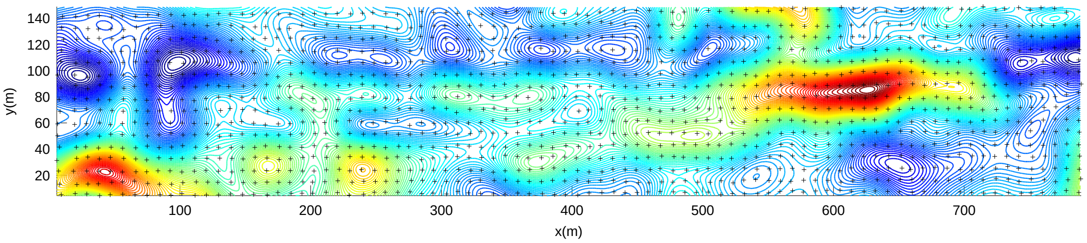
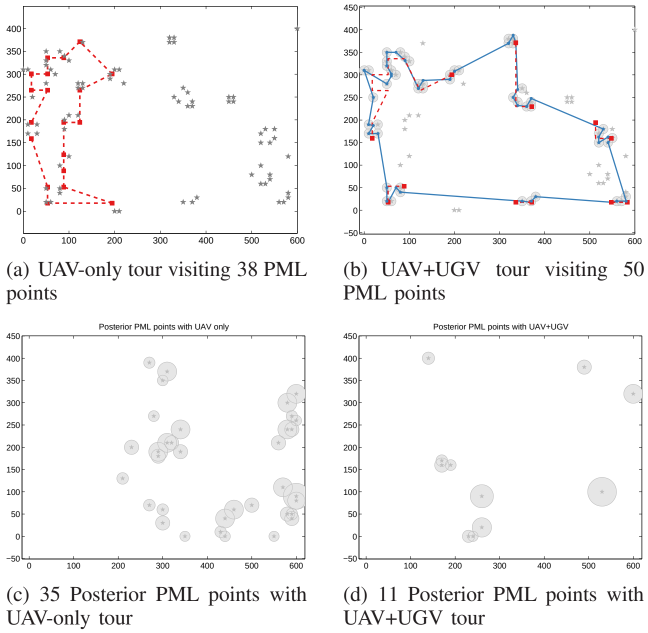


Fig. 8. Soil organic matter data set from [44]. Dense sampling was collected by hand (black crosses) and used to train a Gaussian Process. The resulting estimate of nitrogen levels is shown as the contour map. From this data set we learn the sensor noise values σ_a and σ_g, as well as model the underlying soil organic matter for larger simulations (Figure best viewed in color).



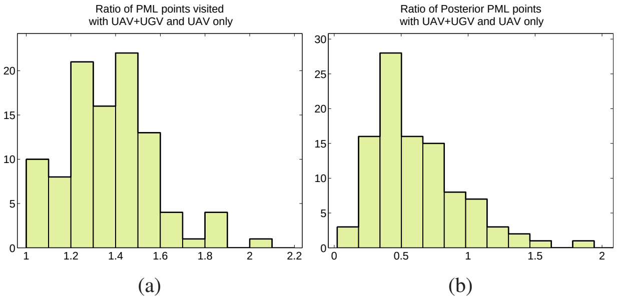
(a) UAV-only tour visiting 38 PML points

(b) UAV+UGV tour visiting 50 PML points

(c) 35 Posterior PML points with UAV-only tour

(d) 11 Posterior PML points with UAV+UGV tour

Fig. 9. Sample simulation instance. (a) & (b) shows the tours found using a UAV-only and UAV+UGV system. The input consists of 75 PML points. The UAV+UGV tour consists of 6 subtours. (c) & (d) shows the PML points found in the updated N level map after incorporating aerial and ground measurements. The UGV allows the UAV to transport to farther locations in the plot which is reflected in fewer posterior PML points.



(a)

(b)

Fig. 10. Histograms of the ratio of (a) number of PML points visited, and (b) number of posterior PML points generated after updating the N map with simulated measurements for UAV+UGV system and a UAV-only system, for 100 random instances. Both systems are given an equal budget of 25 minutes.

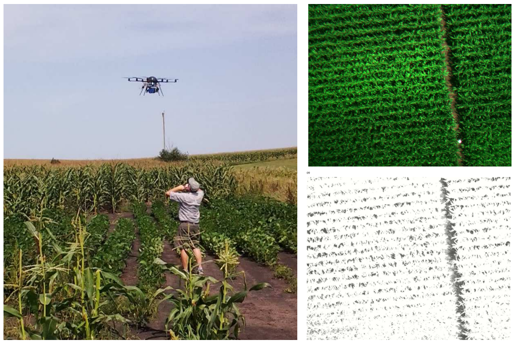
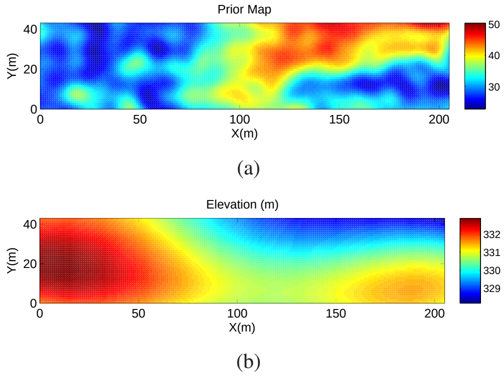


Fig. 11. Preliminary field experiment with a UAV carrying a multi-spectral camera in a corn plot in Janesville, MN, U.S.A. Visible and near-infrared images shown were obtained from an altitude of 30 m during the experiment.

data collection with an autonomous UAV (Figure 11). The experiments were conducted in a corn plot at Janesville, MN, U.S.A. The corn plot is a 122 m × 61 m site for studying the effect of fertilizer treatments on nitrogen stress. The UAV was fitted with a multi-spectral camera from Tetracam [48]. Figure 11 shows the visible and NIR images obtained from 30 m altitude. A camera footprint of diameter $C = 30\,\text{m}$ was determined empirically for a flying altitude of 30 m.

A prior estimate was built using dipole data as a proxy for the nitrogen level map. The dipole data consists of ground measurements of electrical conductivity of the soil. The conductivity, in turn, depends on the land elevation, soil moisture, and soil texture. Elevation changes lead to water run-off leading to changes in the nitrogen levels due to leaching. The prior estimate built using the dipole data is shown in Figure 12(a), and the elevation map of the plot is shown in Figure 12(b).

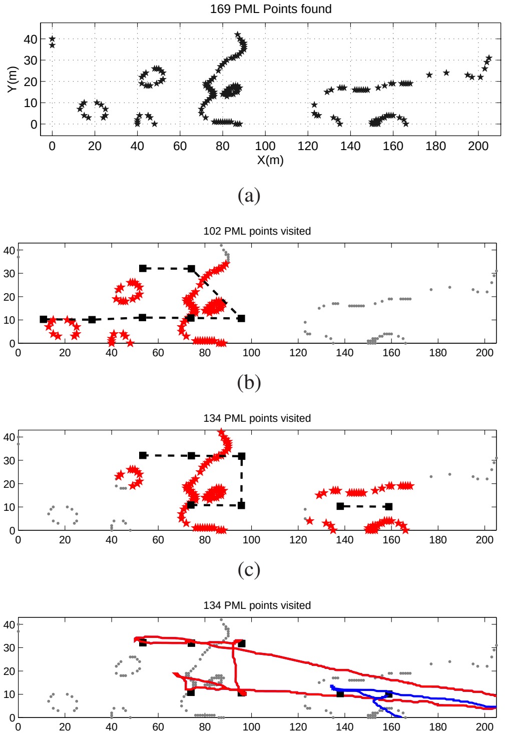


(a)

(b)

Fig. 12. (a) Prior map built using dipole measurements as proxy for nitrogen levels. Dipole measurements of the soil conductivity depend on the elevation and soil moisture and texture, which in turn affect nitrogen levels. (b) Elevation map of the test site. (Figures best viewed in color)

Each point in the prior map was labeled as either high or low, using the average prior map value as the threshold. The desired maximum probability of mislabeling was set to $P_d = 0.45$. 169 PML points were identified based on this (Figure 13(a)). These points were partitioned into a grid of resolution $C/\sqrt{2}$. The orienteering algorithm was run on the graph constructed using the grid. All the tours were computed considering a nominal UAV speed of 4 m/s. The waypoint following controller on-board the UAV was programmed to maintain this speed. The typical battery lifetime of the UAV at this speed was observed to be approximately 600 s. The UAV can potentially cover all the input PML points in 600 s. Since the goal of this experiment was to demonstrate the proof-of-concept implementation for a larger setup, we restricted the battery lifetime to 200 s.



(a)

(b)

(c)

(d)

Fig. 13. (a) 169 PML points were found after classifying the prior map into two labels. (b) The camera footprint along a UAV-only tour covered only 102 PML points (marked by larger star, in red) with a maximum budget of 200 s. (c) UAV+UGV tour consisted of two paths, and covered 134 of the PML points with the same budget. (d) GPS coordinates of the UAV for executing the tours in (c).

Figure 13(c) shows the UAV tour found for a UAV-only system. The number of points visited by the UAV increase from 102 with a UAV-only system to 134, using a UAV+UGV system. Figure 13(b) shows two deployments computed for a UAV+UGV system. Figure 13(d) shows the GPS coordinates of the UAV during actual execution of the two deployments given in Figure 13(b). We programmed the UAV to take-off and land from the same location at one of the corners of the plot where we were stationed (instead of the locations computed by the algorithm) for safety purposes. Our ongoing work is focused on implementing the UGV system.

VIII. Conclusion

In this paper, we studied two informative path planning problems motivated by the use of robots in precision agriculture. Precision agriculture uses data from ground and aerial sensors in order to estimate and predict the status of crops. Obtaining a soil measurement from a ground robot requires spending some time. With this as motivation, we introduced a new planning problem, termed as the SamplingTSPN problem, which penalizes the time spent in traveling and the time spent for obtaining measurements. We presented an $\mathcal{O}\left(\frac{r_{\max}}{r_{\min}}\right)$ approximation algorithm where $r_{\min}$ and $r_{\max}$ are the minimum and maximum radius of input disks. Aerial images can be obtained instantaneously; however, small UAVs have limited on-board energy. We studied the problem of maximizing the number of points visited by the UAV, subject to its maximum battery lifetime. Unlike traditional approaches, our algorithm takes into consideration the situation where the UAV can land on the UGV and thus be carried between points without expending energy.

In order to execute the algorithms presented in this paper, additional capabilities such as soil sampling and autonomous landing are necessary, which are part of our ongoing work. Future work includes studying the online version of the planning problems. In the online version, the N level map is updated after each measurement. New PML points are likely to appear during sampling tours. This presents an interesting trade-off between using the slower UGV measurements to handle the new PML points, or using part of the UAV budget to cover a large number of points simultaneously.

Acknowledgment

This work is supported by NSF Awards #1111638, #0917676, and #1566247 and a grant from MnDrive RSAM initiative. We thank Roman Ripp, Andrew Scoobie, and Daniel Kaiser for their help with the field experiments.

References

[1] P. Tokekar, J. Vander Hook, D. Mulla, and V. Isler, "Sensor Planning for a Symbiotic UAV and UGV System for Precision Agriculture," in *Proc. International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[2] A. Meliou, A. Krause, C. Guestrin, and J. M. Hellerstein, "Nonmyopic informative path planning in spatio-temporal models," in *AAAI*, vol. 10, no. 4, 2007, pp. 16–7.

[3] S. T. Jawaid and S. L. Smith, "Informative path planning as a maximum traveling salesman problem with submodular rewards," *Discrete Applied Mathematics*, vol. 186, pp. 112–127, 2015.

[4] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies," *The Journal of Machine Learning Research*, vol. 9, pp. 235–284, 2008.

[5] N. Cao, K. H. Low, and J. M. Dolan, "Multi-robot informative path planning for active sensing of environmental phenomena: A tale of two algorithms," in *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2013, pp. 7–14.

[6] A. Singh, A. Krause, C. Guestrin, and W. J. Kaiser, "Efficient informative sensing using multiple robots," vol. 34, no. 2, 2009, p. 707.

[7] J. Yu, M. Schwager, and D. Rus, "Correlated orienteering problem and its application to informative path planning for persistent monitoring tasks," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 342–349.

[8] D. J. Mulla, “Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps,” *Biosystems Engineering*, vol. 114, no. 4, pp. 358 – 371, 2013, special Issue: Sensing Technologies for Sustainable Agriculture.

[9] D. Mulla, “Mapping and managing spatial patterns in soil fertility and crop yield,” in *Soil Specific Crop Management*. American Society of Agronomy, 1993, pp. 15–26.

[10] G. W. Randall and D. J. Mulla, “Nitrate nitrogen in surface waters as influenced by climatic conditions and agricultural practices,” *Journal of Environmental Quality*, vol. 30, no. 2, pp. 337–344, 2001.

[11] H. Choset, “Coverage of known spaces: The boustrophedon cellular decomposition,” *Autonomous Robots*, vol. 9, no. 3, pp. 247–253, 2000.

[12] A. Blum, S. Chawla, D. R. 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.

[13] K. H. Low, J. Chen, J. M. Dolan, S. Chien, and D. R. Thompson, “Decentralized active robotic exploration and mapping for probabilistic field classification in environmental sensing,” in *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, 2012, pp. 105–112.

[14] B. Zhang and G. S. Sukhatme, “Adaptive Sampling for Estimating a Scalar Field using a Robotic Boat and a Sensor Network,” *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 3673–3680, Apr. 2007.

[15] D. Song, C.-Y. Kim, and J. Yi, “Simultaneous localization of multiple unknown and transient radio sources using a mobile robot,” *Robotics, IEEE Transactions on*, vol. 28, no. 3, pp. 668–680, 2012.

[16] M. Otte, N. Correll, and E. Frazzoli, “Navigation with foraging,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 3150–3157.

[17] S. L. Smith, M. Schwager, and D. Rus, “Persistent robotic tasks: Monitoring and sweeping in changing environments,” *Robotics, IEEE Transactions on*, vol. 28, no. 2, pp. 410–426, 2012.

[18] J. Yu, S. Karaman, and D. Rus, “Persistent monitoring of events with stochastic arrivals at multiple stations,” *Robotics, IEEE Transactions on*, vol. 31, no. 3, pp. 521–535, June 2015.

[19] A. Singh, A. Krause, and W. J. Kaiser, “Nonmyopic Adaptive Informative Path Planning for Multiple Robots,” *International Joint Conference on Artificial Intelligence*, pp. 1843–1850, 2008.

[20] A. Dumitrescu and J. S. Mitchell, “Approximation algorithms for tsp with neighborhoods in the plane,” *Journal of Algorithms*, vol. 48, no. 1, pp. 135–159, 2003.

[21] P. Kamousi and S. Suri, “Euclidean traveling salesman tours through stochastic neighborhoods,” in *Algorithms and Computation*. Springer, 2013, pp. 644–654.

[22] D. Bhadauria, O. Tekdas, and V. Isler, “Robotic data mules for collecting data over sparse sensor fields,” *Journal of Field Robotics*, vol. 28, no. 3, pp. 388–404, 2011.

[23] O. Tekdas, D. Bhadauria, and V. Isler, “Efficient data collection from wireless nodes under the two-ring communication model,” *The International Journal of Robotics Research*, vol. 31, no. 6, pp. 774–784, 2012.

[24] G. Hollinger, S. Choudhary, P. Qarabaqi, C. Murphy, U. Mitra, G. S. Sukhatme, M. Stojanovic, H. Singh, F. Hover, *et al.*, “Underwater data collection using robotic sensor networks,” *Selected Areas in Communications, IEEE Journal on*, vol. 30, no. 5, pp. 899–911, 2012.

[25] R. Sugihara and R. K. Gupta, “Optimal speed control of mobile node for data collection in sensor networks,” *Mobile Computing, IEEE Transactions on*, vol. 9, no. 1, pp. 127–139, 2010.

[26] H. Alt, E. M. Arkin, H. Brönnimann, J. Erickson, S. P. Fekete, C. Knauer, J. Lenchner, J. S. Mitchell, and K. Whittlesey, “Minimum-cost coverage of point sets by disks,” in *Proceedings of the twenty-second annual symposium on Computational geometry*. ACM, 2006, pp. 449–458.

[27] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas, “Cooperative air and ground surveillance,” *Robotics & Automation Magazine, IEEE*, vol. 13, no. 3, pp. 16–25, 2006.

[28] P. Sujit and S. Saripalli, “An empirical evaluation of co-ordination strategies for an auv and uav,” *Journal of Intelligent & Robotic Systems*, vol. 70, pp. 373–384, 2013.

[29] H. G. Tanner, “Switched uav-ugv cooperation scheme for target detection,” in *Robotics and Automation, 2007 IEEE International Conference on*. IEEE, 2007, pp. 3457–3462.

[30] M. Bergerman, E. van Henten, J. Billingsley, J. Reid, and D. Mingcong, “Ieee robotics and automation society technical committee on agricultural robotics and automation [tc spotlight],” *Robotics Automation Magazine, IEEE*, vol. 20, no. 2, pp. 20–125, June 2013.

[31] C. W. Bac, E. J. van Henten, J. Hemming, and Y. Edan, “Harvesting robots for high-value crops: State-of-the-art review and challenges ahead,” *Journal of Field Robotics*, vol. 31, no. 6, pp. 888–911, 2014. [Online]. Available: http://dx.doi.org/10.1002/rob.21525

[32] D. Ball, P. Ross, A. English, T. Patten, B. Upcroft, R. Fitch, S. Sukkarieh, G. Wyeth, and P. Corke, “Robotics for sustainable broad-acre agriculture,” in *Field and Service Robotics*. Springer, 2015, pp. 439–453.

[33] S. Nuske, K. Wilshusen, S. Achar, L. Yoder, S. Narasimhan, and S. Singh, “Automated visual yield estimation in vineyards,” *Journal of Field Robotics*, vol. 31, no. 5, pp. 837–860, 2014.

[34] J. Das, G. Cross, C. Qu, A. Makineni, P. Tokekar, Y. Mulgaonkar, and V. Kumar, “Devices, systems, and methods for automated monitoring enabling precision agriculture,” in *Proceedings of IEEE Conference on Automation Science and Engineering*. IEEE, 2015, to appear.

[35] M. Bergerman, S. M. Maeta, J. Zhang, G. M. Freitas, B. Hamner, S. Singh, and G. Kantor, “Robot farmers: Autonomous orchard vehicles help tree fruit production,” *Robotics & Automation Magazine, IEEE*, vol. 22, no. 1, pp. 54–63, 2015.

[36] C. E. Rasmussen and C. K. Williams, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 1.

[37] D. E. Myers, “Estimation of linear combinations and co-kriging,” *Mathematical Geology*, vol. 15, no. 5, pp. 633–637, 1983.

[38] S. Arora, “Nearly linear time approximation schemes for euclidean tsp and other geometric problems,” in *Foundations of Computer Science, 1997. Proceedings., 38th Annual Symposium on*. IEEE, 1997, pp. 554–563.

[39] N. H. Mustafa and S. Ray, “Improved results on geometric hitting set problems,” *Discrete & Computational Geometry*, vol. 44, no. 4, pp. 883–895, 2010.

[40] J. S. Mitchell, “Guillotine subdivisions approximate polygonal subdivisions: A simple polynomial-time approximation scheme for geometric tsp, k-mst, and related problems,” *SIAM Journal on Computing*, vol. 28, no. 4, pp. 1298–1309, 1999.

[41] “Clearpath robotics,” 2013 (accessed Jan 2013). [Online]. Available: http://clearpathrobotics.com

[42] “Spectrum technologies inc.” 2013 (accessed Mar 2013). [Online]. Available: http://www.specmeters.com/

[43] “Mikrokopter,” 2013 (accessed Jan 2013). [Online]. Available: http://mikrokopter.de

[44] D. Mulla, M. Beatty, and A. Sekely, “Evaluation of remote sensing and targeted soil sampling for variable rate application of nitrogen,” in *Proceedings of 5th International Conference on Precision Agriculture*. American Society of Agronomy, Crop Science Society of America, and Soil Science Society of America, 2001.

[45] C. E. Rasmussen and H. Nickisch, “Gaussian processes for machine learning (gpml) toolbox,” *The Journal of Machine Learning Research*, vol. 11, pp. 3011–3015, 2010.

[46] A. Krause, “Sfo: A toolbox for submodular function optimization,” *The Journal of Machine Learning Research*, vol. 11, pp. 1141–1144, 2010.

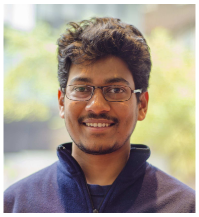
[47] D. Applegate, R. Bixby, V. Chvatal, and W. Cook, “Concorde tsp solver,” *URL http://www. tsp. gatech. edu/concorde*, 2006.

[48] “Tetracam,” 2014 (accessed Feb 2014). [Online]. Available: www.tetracam.com

APPENDIX

PROOF FOR LEMMA 4

Proof: We verify G is a metric graph. Consider a triple of vertices u, v, w. We know $w(u,v), w(v,w), w(w,u) \leq 2C_a$. It is easy to see the triangle inequality holds when two or three edges have weights equal to $2C_a$. Consider the case when only one edge has weight equal to $2C_a$, say $w(u,v) = 2C_a$. Now, $w(v,w)+w(w,u) = d(v,w)+d(w,u) \geq d(u,v)$. Since $w(u,v) = \min\{2C_a, d(u,v)\} = 2C_a$, we have $d(u,v) \geq 2C_a$. Hence, $w(v,w) + w(w,u) \geq w(u,v)$. And since $w(u,v) = 2C_a$ and $w(v,w), w(w,u) < 2C_a$, $w(u,v) + w(w,u) \geq w(v,w)$ and $w(u,v)+w(v,w) \geq w(w,u)$. For the case when all three edges have weights less than $2C_a$, the weights are equal to Euclidean distances. Hence, weights satisfy triangle inequality in addition to symmetry, identity and non-negativity. Hence, the graph constructed above is a complete metric graph. ■



Pratap Tokekar received the B.Tech. degree in electronics and telecommunication from College of Engineering Pune, Pune, India, in 2008 and the Ph.D. degree in computer science from University of Minnesota, Minneapolis, MN, USA, in 2014. He is currently an Assistant Professor in electrical and computer engineering at the Virginia Polytechnic Institute and State University, Blacksburg, VA, USA. His research interests include algorithmic and field robotics, sensor networks, and computational geometry.

Joshua Vander Hook is a Research Technologist at NASA JPL, Pasadena, CA, USA in the Estimation, Decision, and Control (347E) group. He received the B.Sc. degree in computer science from Minnesota State University, Mankato, MN, USA, in 2010 and the Ph.D. degree in computer science from University of Minnesota, Minneapolis, MN, USA in 2015. His research interests primarily include motion planning, tracking and estimation, and pursuit-evasion games. He received fellowships from University of Minnesota, the ARCS Foundation, and the Sigma Xi Research Society.

David Mulla received a Ph.D. degree in Agronomy from Purdue University with emphasis in soil physics (1983). From 1983 to 1995 he was Assistant to Full Professor in the Dept. of Crop and Soil Sciences at Washington State Univ. Since 1995 he has been Professor and Larson Chair for Soil & Water Resources in the Dept. of Soil, Water, and Climate at the Univ. of Minnesota. Since January, 2004 he has been the Director of the Precision Agriculture Center at the Univ. of Minnesota and organized the 7th and 8th International Conferences on Precision Agriculture. In 2007 he was appointed a Founding Fellow in the Univ. of Minnesotas Institute on Environment. In 2011 and 2013 he was appointed to National Academy of Sciences committees on Florida water quality standards and on Mississippi River water quality assessment and monitoring.

Volkan Isler received the B.S. degree in computer engineering from Bogazici University, Istanbul, Turkey, in 1999, and the M.S.E. and Ph.D. degrees in computer and information science from University of Pennsylvania, Philadelphia, PA, USA, in 2000 and 2004, respectively. He is an Associate Professor with the University of Minnesota, Minneapolis, MN, USA. His research interests include robotics, sensing, and geometric algorithms. Dr. Isler has served as an Associate Editor for IEEE TRANSACTIONS ON ROBOTICS and IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING. From 2009 to 2015, he was the Chair of the RAS Technical Committee on networked robots. He has received the CAREER Award from the National Science Foundation.