UNIVERSITY OF CALIFORNIA
Santa Barbara

UAV Data Mule Vehicle Routing Problems In Sparse Sensor Networks

A dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Electrical and Computer Engineering

by

Jason Tony Isaacs

Committee in Charge:

Professor João P. Hespanha, Chair

Professor Upamanyu Madhow

Professor Francesco Bullo

Professor Subhash Suri

June 2012

The dissertation of
Jason Tony Isaacs is approved:

Professor Upamanyu Madhow

Professor Francesco Bullo

Professor Subhash Suri

Professor João P. Hespanha, Committee Chair

March 2012

UAV Data Mule Vehicle Routing Problems In Sparse Sensor Networks

by

Jason Tony Isaacs

For my father and my mother.

Acknowledgements

I would like to begin by thanking Prof. João Hespanha for taking a chance on a non-traditional graduate student and bringing me to Santa Barbara. You truly have a special way of approaching problems and of clearly explaining results. I hope these skills have rubbed off on me in some way. To Prof. Madhow, thank you for all of your advice and ideas over the course of my research as well as your frank feedback. To the rest of my committee, Prof. Bullo and Prof. Suri, thank you for your service and your insightful questions and comments.

I am indebted to the faculty of CCDC for providing world class training in Control Systems. A special thanks to Prof. Petar Kokotović and Prof. Karl Åström for showing infinite patience with me during my first year of graduate study. The enthusiasm and kindness that each of you displayed during our many office hour visits will serve as an inspiration for the rest of my career. To Val de Veyra, the de facto problem solver for all aspects of graduate school, thank you for always helping and for all the entertaining conversations.

I have been fortunate to enjoy the benefits of collaborating with other members of my lab. I would like to thank Daniel Klein for your great advice. The quality of our work was positively impacted by your insight, guidance, and technical skills. To Sriram Venkateswaran, thank you for all you did to make our project a success. It was a joy to experience the ups and downs of hardware

related research with you. To Shaunak Bopardikar, I am glad we had an opportunity to collaborate. Thanks for all the discussions and phone calls. I would also like to thank Johann Schweikl for providing the interface between our algorithms and the flight simulator. I am also thankful for the numerous relationships that have developed with my lab mates Josh, Max, Steven, Soheil, Alexandre, Farshad, and Hari.

My research was supported by the Institute for Collaborative Biotechnologies through contract No. W911NF-09-D-0001 from the U.S. Army Research Office. I also received generous support from the iRobot Corporation, the Institute for Collaborative Biotechnologies through contract No. W911NF-09-D-0001 from the U.S. Army Research Office, and the National Science Foundation Grants No. ECCS-0725485 and No. 0720842 to pursue opportunities teach high school students about control systems and robotics.

Finally, I would like to thank my family. To Abby, thank you for your love and support. I hope our detour through Santa Barbara has been a positive experience for you. To Ewan, thank you for reminding me everyday of the beautiful things in life. To my mother Sally, thank you for teaching me first hand that you can never be too old to learn. To my father Raymond, thank you for always being my biggest supporter.

Curriculum Vitæ

Jason Tony Isaacs

Education

2006 - 2008	MS in Electrical and Computer Engineering, University of California, Santa Barbara
1994 - 1999	BS in Electrical Engineering, University of Kentucky
1994 - 1999	BS in Engineering Physics, Eastern Kentucky University

Experience

2008 - 2012	Graduate Student Researcher, University of California, Santa Barbara
2008 - 2008	Intern, Army Research Laboratory, Adelphi, MD
2007 - 2008	Teaching Assistant, University of California, Santa Barbara
2000 - 2006	Senior Hardware Development Engineer, Lexmark International Inc., Lexington, KY

Selected Publications

D. J. Klein, S. Venkateswaran, J. T. Isaacs, T. Pham, J. Burman, J. Hespanha, and U. Madhow, "Source Localization in a Sparse Acoustic Sensor Network using UAVs as Information Seeking Data Mules," To appear *ACM Transactions on Sensor Networks.*

J. T. Isaacs, D. J. Klein, and J. P. Hespanha, "Algorithms for the Traveling Salesman Problem with Neighborhoods Involving a Dubins Vehicle," in *Proceedings of the 2011 American Control Conference,* July 2011.

D. J. Klein, J. J. Schweikl, J. T. Isaacs, and J. P. Hespanha, "On UAV Routing Protocols for Sparse Sensor Data Exfiltration," in *Proceedings of the 2010 American Control Conference,* June 2010.

J. T. Isaacs, D. J. Klein, and J. P. Hespanha, "Optimal Sensor Placement For Time Difference of Arrival Localization," in *Proceedings of the 48th IEEE Conference on Decision and Control,* December 2009.

Abstract

UAV Data Mule Vehicle Routing Problems In Sparse Sensor Networks

by

Jason Tony Isaacs

Recent advances in technology have enabled the use of wireless sensor networks for environmental monitoring and surveillance. Wireless sensor networks are particularly beneficial for monitoring environments that are unsuitable for human presence, such as those arising in the monitoring of permafrost, volcanos, forest fires, and battlefields. Mobile agents called data mules can be used to enhance sensor networks by visiting individual sensors to collect measurements. In this thesis, the mobility of the data collector is exploited to mitigate energy depletion of the stationary nodes and allow for sparse deployments at the cost of additional data latency. This motivates us to seek efficient strategies for the mobile vehicle in order to alleviate this data latency.

The policies developed here apply directly to an acoustic source localization problem in which the objective is to localize the source of a transient acoustic event using measurements from a group of unattended ground sensors. First we study the optimal sensor placement of acoustic time of arrival sensors for local-

ization using techniques from information theory and optimization. A sparse sensor configuration is shown to be optimal when maximizing the expected determinant of the Fisher information matrix for truncated, radially-symmetric source distributions. Next, we use ideas from optimal sensor placement and optimal sensor selection to adaptively adjust the route of the data mule to minimize the time to localize events of interest. When the data mule is a small fixed wing unmanned aerial vehicle (UAV), this algorithm can be improved by including the kinematic constraints of the UAV and exploring the ability of the UAV to communicate with faraway sensors within its line of sight. Finally, we provide data mule routing policies for acoustic source localization over a large area involving many sources and sensors and analyze their performance.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the evolution of technology comes greater capacity and exciting new challenges. Recent advances in energy storage, computation, sensing, wireless communication, and mobile robotics have enabled an enormous growth in the use of sensor networks in the areas of surveillance and security. In many of these applications wireless sensor networks are used in areas that are remote and hazardous for human involvement. It is common to assume that wireless sensor networks are fully connected in the sense that any node can reach any other node through multi-hop communication. However, maintaining this level of connectivity typically requires very dense deployments that may not be cost effective, practical, or even possible. Examples where this issue arises include forest fire monitoring [47], permafrost monitoring [45], and volcano monitoring [97]. In a quote from [26] the authors describe the challenges of communication among unattended ground sensors used for battlefield montitoring.

> Also, for networks on the ground, RF transmission degrades with distance much faster than in free space, which means that communication distance and energy must be well managed. [26]

In another quote from [97], the authors describe the desire to spread sensors farther apart than radio connectivity allowed.

> We attempted to position the stations as far apart as the radios on each node would allow. Although our antennae could maintain radio links of more than 400 meters, the geography at the deployment site occasionally required installing additional stations to maintain radio connectivity. [97]

In each of the above mentioned applications, a mobile robotic service vehicle could have helped alleviate some of the communication related challenges by exploiting the mobility to gather the information from the sensors. The focus of this thesis is precisely on the design of policies for routing a mobile agent to efficiently collect data from sparse sensor networks that are otherwise unable to form a connected communication network. The concept of data MULEs was introduced by [80]. In this work, mobile agents are equipped with communication devices and move randomly in the environment collecting data from sensors opportunistically when they come into radio contact. In more recent work the path of the data mule is actively planned to enhance the collection of data [15, 90, 108]. The data mules in these applications typically follow their planned paths and do not seek to interpret measurements in order to dynamically adjust their routes.

We focus our work on the problem of persistent surveillance of a battlefield using unattended acoustic sensors and an unmanned aerial vehicle as the data collector. We consider scenarios where sensors record the time at which they detect the acoustic signature of an event of interest. With these time measurements and knowledge of the position of the sensors one can estimate the location of the source of the acoustic event. We begin by studying the problem of optimal sensor placement, which provides insight into how sensor/source geometry plays a role in estimation quality. We then take as given a set of sensor positions and consider a dynamic vehicle routing (DVR) problem in which one wants to optimize the path of a data mule that moves from sensor to sensor to gather time measurement and fuses this information to localize the source of the acoustic event as quickly as possible. We develop algorithms for a wide range of regimes. On one extreme, we consider scenarios where acoustic events are infrequent and the number of sensors is small, which permits the use of highly optimized routing algorithms. Considerations of computational efficiency lead us to different algorithms that scale well with the number of sensors and with the frequency of the acoustic events, at the expense of some degradation in the time-to-detection.

1.1 Statement of Contributions

In the remainder of this section, we briefly summarize the key contributions of this thesis.

Chapter 2 - Optimal Sensor Placement: We address the problem of optimal sensor placement for localizing a source from noisy time-of-arrival measurements. In particular, we are interested in the optimal placement of M planar sensors so as to yield the best *expected* source location estimate. The main result, on maximizing the expected determinant of the Fisher information matrix for truncated, radially-symmetric source distributions, shows two features not previously observed. First, the sensors should be placed as far from the expected source position as possible. Second, the sensors should be arranged in a *splay* configuration in which neighboring sensors are separated by equal-angle increments. Specific examples are given for point, uniform, and truncated-Gaussian source density functions.

Chapter 3 - Strategies for Individual Source Localization: We propose an architecture for on-the-fly inference while collecting data from sparse sensor networks. In particular, we consider source localization using acoustic sensors dispersed over a large area, with the individual sensors located too far apart for direct connectivity. An unmanned aerial vehicle (UAV) is employed for

collecting sensor data, with the UAV route adaptively adjusted based on data from sensors already visited, in order to minimize the time to localize events of interest. The UAV acts as an information seeking data mule, not only providing connectivity, but also making Bayesian inferences from the data gathered in order to guide its future actions. The effectiveness of the proposed algorithm is verified via Monte Carlo simulations using a high fidelity flight simulator.

Chapter 4 - Dubins Traveling Salesman Problem with Neighborhoods: The results in Chapter 3 rely on the simplifying assumption that the distance between sensors is sufficiently large that one can ignore the vehicle dynamics in planning sensor-to-sensor paths and also that the radio communication radius of each sensor is much smaller than the sensor-to-sensor distances. As the sensor density increases, these assumptions are not realistic and one must take into consideration the vehicle dynamics and the sensor communication ranges. For UAVs, the key constraint imposed by the vehicle dynamics corresponds to a curvature constraint, which led us to study the problem of finding the minimum-length curvature constrained closed path through a set of regions in the plane (corresponding to the sensor communication ranges). This problem is referred to as the Dubins Traveling Salesperson Problem with Neighborhoods (DTSPN). Two algorithms are presented that transform this infinite dimensional combinatorial optimization problem into a finite dimensional asymmetric TSP by sam-

pling and applying the appropriate transformations, thus allowing the use of existing approximation algorithms. We show for the case of disjoint regions, the first algorithm needs only to sample each region once to produce a tour within a factor of the length of the optimal tour that is independent of the number of regions. We present a second algorithm that performs no worse than the best existing algorithm and can perform significantly better when the regions overlap.

Chapter 5 - Dynamic Vehicle Routing In A Large Sparse Sensor Network: The results in Chapters 3 and 4 are essentially concerned with the detection and localization of one event at a time. As the frequency of events increases this is no longer feasible and one must move towards solutions in which a vehicle gathers information about multiple events simultaneously. In such scenarios one no longer minimizes the time needed to localize a specific event and instead minimizes the expected time needed to localize a generic event. We thus consider a dynamic vehicle routing problem in which a service vehicle visits several unattended ground sensors (UGS) in order to localize randomly occurring events. These events arrive in time according to a temporal Poisson process and are uniformly distributed in a bounded convex region in the plane. The UGS can detect events occurring within a known range of their locations, but a single sensor cannot provide sufficient information to perfectly localize the event. The service vehicle must then visit multiple sensors that have detected the event

and fuse the information obtained from the different sensors to determine the precise location of the event. The goal is to minimize the expected time between when an event occurs and when it has been localized. We provide a family of routing policies for the service vehicle and a criteria to select which is the most appropriate policy, in order to minimize the expected localization time. Policy selection is based on the arrival rate of the events.

Chapter 6 - Implementation and Field Demonstrations: We report on a field demonstration and the implementation of an acoustic sensor network on which we have tested several of the algorithms described in this thesis. The demonstration focused on individual source localization and used algorithms developed in Chapter 3.

1.2 Organization of this Thesis

This Thesis is organized into seven chapters. Optimal placement for acoustic time of arrival sensors is presented in Chapter 2. Concepts from optimal sensor placement are used to develop a UAV routing algorithm for single acoustic source localization in Chapter 3. In Chapter 4, methods are presented to address the Dubins traveling salesman problem with neighborhoods. Chapter 5 considers the dynamic vehicle routing problem in an acoustic source localization

application over many sensors. A field demonstration of the work from Chapter 3 is presented in Chapter 6. Finally, conclusions and future directions are discussed in Chapter 7.

Chapter 2

Optimal Sensor Placement

2.1 Introduction

In the time-of-arrival (TOA) source localization problem, acoustic or electromagnetic radiation emitted by a source is received by many spatially distributed sensors. Each sensor records the time at which the signal arrives and relays this information to a fusion center. Using these measurements in conjunction with the known position of each sensor, a centralized algorithm can estimate the source location and event time. Source localization has a wide range of both civilian and military applications including speaker identification in a conference room [51, 59], cell-phone geo-location [23], and sniper and mortar localization [82].

A key question that arises in the source localization problem is where to place the sensors. Answering this question amounts to finding sensor arrange-

ments that provide the most valuable information to any particular unbiased localization algorithm. The localization algorithm performance is typically characterized by the Cramér-Rao lower bound (CRB) on the estimator variance, or by the Fisher information matrix (FIM). The CRB is the inverse of the FIM, and thus exists only when the FIM is non-singular.

A good sensor configuration results in a large FIM, or a small CRB. The size of these matrices can be measured by either the trace or the determinant. Physically, the trace of the CRB corresponds to the lower bound on the mean squared error of an unbiased estimate. The determinant of the CRB matrix is proportional to the lower bound on the volume of the uncertainty ellipse associated with an unbiased estimate. Here, we work with the determinant of the Fisher information matrix which is inversely proportional to the determinant of the CRB matrix.

Relevant Literature

A majority of the previous work on optimal sensor placement considers the problem of placing sensors for a *known* source location. Necessary and sufficient conditions have been found for maximizing the FIM determinant [16], and sufficient conditions are known for minimizing the CRB trace [102–104]. In these results, the critical factor is the relative angles between the sensors. The distance

between each sensor and the source is irrelevant because neither the CRB nor FIM depend on source-to-sensor distance.

Only in recent work has the optimal sensor placement problem been considered in the context of an uncertain source location. In [64], numerical optimization techniques were used to minimize the expected CRB determinant. The examples discussed suggest that optimal sensor configurations have sensors spread out along the boundary of the surveillance area. In [61], a single source is allowed to move within the surveillance area while the sensors move to optimal configurations along the boundary of the region.

Contributions

We provide solutions to the optimal sensor placement problem for an *uncertain* source location, using the expected value of the FIM determinant as an optimality criteria. In particular, results are derived for truncated, radially symmetric source distributions when the sensors are confined to either a ring, or an annular region. Through a combination of analysis and simulation, we find two new results for the case of an unknown source location. First, the distance between the sensors and the expected source position matters and should be made as large as physically possible. Second, *splay* configurations like the one depicted in Fig. 2.1, in which sensors are arranged at equal angle increments, are better than other configurations suggested in the known-source literature.

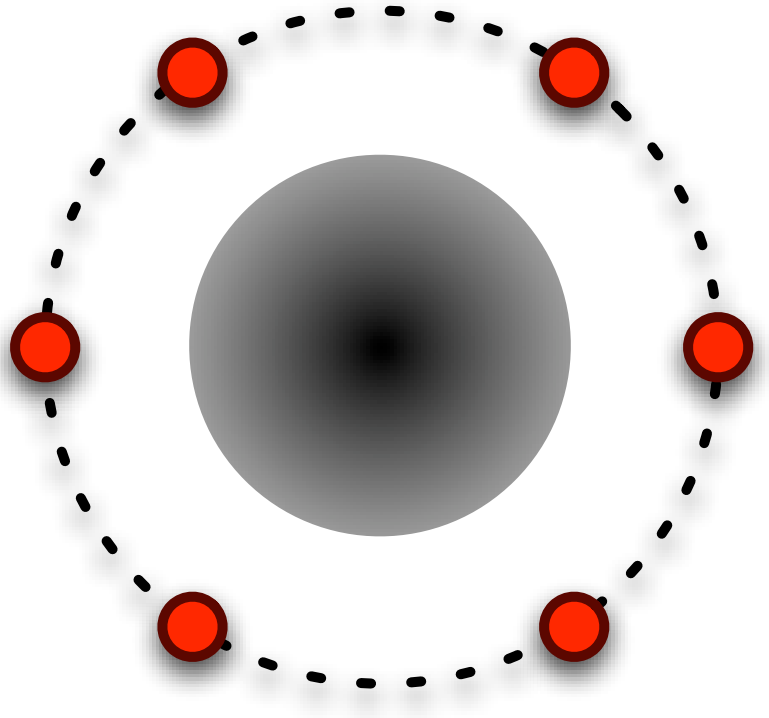


Figure 2.1: A collection of six sensors is shown in an optimal configuration to localize a source distributed according to a truncated-Gaussian density function. The optimal configuration places the sensors at the boundary of the permissible region, indicated by the dashed circle, and at equal angle increments.

Chapter Organization

This chapter is organized as follows. In Section 2.2, the problem of source localization from TDOA measurements is described. The main results concerning optimal sensor placement are presented in Section 2.3. In Section 2.4, numerical results are given to support the analytical work. Optimal sensor geometries

for point source, uniform, and truncated-Gaussian distributions are discussed. Conclusions and future directions are discussed in Section 2.5.

2.2 TDOA Source Localization

In this section, we formulate the source localization problem, describe how time-of-arrival measurements are generated, and introduce the Fisher information matrix and Cramér-Rao bound. Symmetric sensor configurations are also introduced.

2.2.1 The Source Localization Problem

The main elements of a source localization problem are a single source, located at a point $\mathbf{p} \in \mathbb{R}^2$, and M sensors located at points $\mathbf{q} := [\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_M] \in \mathbb{R}^{2\times M}$. At an unspecified event time t_0, the source emits an acoustic or electromagnetic signal which propagates outward at speed ν. Each sensor records the time at which the signal arrives, a measurement called the time of arrival (TOA). The main objective of source localization is to infer the source location from the TOA measurements.

The noisy TOA measurement of sensor i is modeled as:

$$\hat{t}_i = t_0 + \frac{d_i(\mathbf{p}, \mathbf{q})}{\nu} + \epsilon_i, \tag{2.1}$$

where

$$d_i(\mathbf{p}, \mathbf{q}) := \|\mathbf{p} - \mathbf{q}_i\|, \quad i = 1, ..., M \tag{2.2}$$

is the distance between the source and the sensor. The noise, ϵ_i, is assumed to be a zero-mean Gaussian random variable with variance σ^2.

The emission time t_0 can be eliminated by taking differences between TOA measurements. The resulting time-difference-of-arrival (TDOA) measurements can be written as:

$$\hat{t}_{ij} := \hat{t}_i - \hat{t}_j = \frac{1}{\nu} d_{ij}(\mathbf{p}, \mathbf{q}) + \epsilon_{ij}, \tag{2.3}$$

where

$$d_{ij}(\mathbf{p}, \mathbf{q}) := d_i(\mathbf{p}, \mathbf{q}) - d_j(\mathbf{p}, \mathbf{q}) \tag{2.4}$$

is the difference in source-sensor distance between sensors i and j. The noise difference, $\epsilon_{ij} := \epsilon_i - \epsilon_j$, has variance $2\sigma^2$.

A maximum of $(M-1)$ linearly independent TDOA measurements can be found for M sensors. Without loss of generality, all time of arrival differences can be taken with respect to the first sensor to generate the following $(M-1) \times 1$

vectors:

$$\hat{\mathbf{t}}(\mathbf{p},\mathbf{q}) := \left[\begin{array}{c} \hat{t}_{i1} \end{array} \right]_{(i=2,\ldots,M)}, \tag{2.5}$$

$$\mathbf{d}(\mathbf{p},\mathbf{q}) := \left[\begin{array}{c} d_{i1} \end{array} \right]_{(i=2,\ldots,M)}, \tag{2.6}$$

$$\boldsymbol{\epsilon} := \left[\begin{array}{c} \epsilon_{i1} \end{array} \right]_{(i=2,\ldots,M)}. \tag{2.7}$$

The TDOA vector, (2.5), has mean

$$\bar{\mathbf{t}}(\mathbf{p},\mathbf{q}) := E\left[\hat{\mathbf{t}}\right] = \frac{1}{\nu}\mathbf{d}(\mathbf{p},\mathbf{q}) \tag{2.8}$$

and the $(M-1 \times M-1)$ covariance matrix is defined as

$$\begin{aligned} \mathbf{Q} &:= E\left[\left(\hat{\mathbf{t}}-\bar{\mathbf{t}}\right)\left(\hat{\mathbf{t}}-\bar{\mathbf{t}}\right)^T\right] \\ &= \sigma^2\left[\mathbf{I}+\mathbf{1}\mathbf{1}^T\right], \end{aligned} \tag{2.9}$$

where $\mathbf{1} := \begin{bmatrix} 1 & 1 & \ldots & 1 \end{bmatrix}^T$.

2.2.2 The Fisher Information Matrix and Cramér-Rao Bound

The Cramér-Rao bound (CRB) provides a lower bound on the variance of an unbiased estimator and is defined as the inverse of the Fisher information matrix. The Fisher information matrix $\mathbf{F}(\mathbf{p}, \mathbf{q})$ associated with the estimation of the source location $\mathbf{p}$ is computed as [62]:

$$\mathbf{F}(\mathbf{p}, \mathbf{q}) = E\left[\left(\frac{\partial}{\partial p} \ln f(\hat{\mathbf{t}}|\mathbf{p})\right)\left(\frac{\partial}{\partial p} \ln f(\hat{\mathbf{t}}|\mathbf{p})\right)^T\right], \tag{2.10}$$

where $f(\hat{\mathbf{t}}|\mathbf{p})$ is the PDF of $\hat{\mathbf{t}}$ given $\mathbf{p}$. Assuming the noise vector $\boldsymbol{\epsilon}$ is zero mean Gaussian and independent of $\mathbf{p}$, then $f(\hat{\mathbf{t}}|\mathbf{p})$ is the PDF of a Gaussian random variable with mean $\bar{\mathbf{t}}(\mathbf{p})$ and covariance matrix $\mathbf{Q}$,

$$f(\hat{\mathbf{t}}|\mathbf{p}) = \mathcal{N}\left(\hat{\mathbf{t}}; \bar{\mathbf{t}}(\mathbf{p}), \mathbf{Q}\right). \tag{2.11}$$

This leads to the (2×2) Fisher information matrix given by Chan and Ho [25],

$$\mathbf{F}(\mathbf{p}, \mathbf{q}) = \frac{1}{\nu^2}\mathbf{G}^T\mathbf{Q}^{-1}\mathbf{G}, \tag{2.12}$$

where

$$\mathbf{G} := \begin{bmatrix} \mathbf{g}_2^T - \mathbf{g}_1^T \\ \vdots \\ \mathbf{g}_M^T - \mathbf{g}_1^T \end{bmatrix}_{(M-1\times 2)} \tag{2.13}$$

and

$$\mathbf{g}_i := \frac{\mathbf{p} - \mathbf{q}_i}{\|\mathbf{p} - \mathbf{q}_i\|}. \tag{2.14}$$

The Cramér-Rao lower bound is defined as the inverse of the Fisher information matrix, $\mathbf{CR}(\mathbf{p}, \mathbf{q}) = \mathbf{F}(\mathbf{p}, \mathbf{q})^{-\mathbf{1}}$, and is thus defined only when $\mathbf{F}(\mathbf{p}, \mathbf{q})$ is full rank. The matrix $\mathbf{CR}(\mathbf{p}, \mathbf{q})$ is a lower bound on the estimation error in the sense that, if $\mathbf{\Sigma}$ is the covariance of an unbiased estimator of $\mathbf{p}$, then $\mathbf{\Sigma} - \mathbf{CR}(\mathbf{p}, \mathbf{q})$ is positive semidefinite. It is important to note that the matrices $\mathbf{F}(\mathbf{p}, \mathbf{q})$ and $\mathbf{CR}(\mathbf{p}, \mathbf{q})$ depend significantly on the relative geometry of the source, $\mathbf{p}$, and the sensors, $\mathbf{q}$. For the remainder of the chapter, the dependence of $\mathbf{F}$ and $\mathbf{CR}$ on $\mathbf{p}$ and $\mathbf{q}$ is implicitly assumed.

2.2.3 Known Source Optimal Sensor Placement

Much recent work has focused on the problem of placing sensors relative to a *known* source location. Without loss of generality, the source can be placed at the origin. Then, denoting by θ_i the angle of $\mathbf{g}_i$, globally optimal sensor

configurations for maximization of the Fisher determinant satisfy (Theorem 1 in [16]) the following two conditions:

First moment balanced:

$$0 = \frac{1}{M}\sum_{i=1}^{M}\cos(\theta_i) = \frac{1}{M}\sum_{i=1}^{M}\sin(\theta_i). \tag{2.15}$$

Second moment balanced:

$$0 = \frac{1}{M}\sum_{i=1}^{M}\cos(2\theta_i) = \frac{1}{M}\sum_{i=1}^{M}\sin(2\theta_i). \tag{2.16}$$

The first moment condition is met when the sensors are spread about the source in an average sense. The second moment criteria ensures that the Fisher information matrix will have equal eigenvalues, resulting in a confidence ellipse that is circular. All configurations satisfying these conditions have a determinant of the Fisher information matrix equal to $M^2/4\sigma^4\nu^4$. Notice that when the source location is known, the determinant of the Fisher information matrix is independent of source to sensor distance.

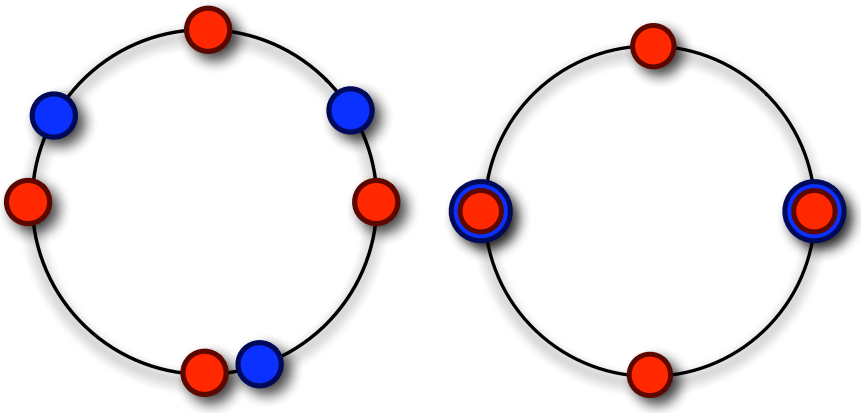
2.2.4 Symmetric Sensor Configurations

When the source location is uncertain, symmetric sensor configurations play an important role.

Definition 2.2.1 (Symmetric Configuration) *A configuration is said to be* symmetric *if for every* $i \in \{1, \ldots, M\}$*, there is an invertible mapping* $f_i : (1, \ldots, M) \to (1, \ldots, M)$ *such that*

$$\mathbf{q}_{f_i(j)} = \left[2\frac{\mathbf{q}_i \mathbf{q}_i^T}{\|\mathbf{q}_i\|^2} - \mathbf{I} \right] \mathbf{q}_j. \tag{2.17}$$

Put more simply, a configuration is symmetric when every unit vector $\mathbf{g}_i$ pointing from sensor $\mathbf{q}_i$ to the source location $\mathbf{p}$ defines an axis of reflective symmetry. Symmetric configurations need not be second moment balanced, and vice versa second moment balanced configurations need not be symmetric, as demonstrated in Fig. 2.2. Another important distinction is that symmetry is distance sensitive, whereas second moment balancing only depends on relative angles.



(a) Second moment balanced (b) Symmetric

Figure 2.2: Both of these sensor configurations are first moment balanced in the sense that (2.15) holds, however (a) shows $M = 7$ sensors in a configuration that is second moment balanced but not symmetric, whereas (b) shows $M = 6$ sensors in a configuration that is symmetric but not second moment balanced.

2.3 Optimal Sensor Placement

In this section, we address the optimal sensor placement problem for truncated, radially symmetric source distributions with a given constraint on the distance from each sensor to the expected source position. At first, we require all sensors to be located at a distance r from the expected source position. Later, we relax this constraint and instead allow sensors anywhere within an annulus of radii $r_1 < r_2$ about the expected source position. The problem we seek to solve can be formally stated as follows:

Problem 2.3.1 *Given M sensors, a radial region $\mathcal{D} \subset \mathbb{R}^2$, and a radially symmetric probability density function $\phi(\|p\|)$ for the source position $\mathbf{p} \in \mathcal{P} \subset \mathbb{R}^2$, find the sensor locations $\mathbf{q}_i \in \mathcal{D}$ for $i = 1, ..., M$ that maximize the expected value of the determinant of the Fisher information matrix,*

$$\mathbf{q}^* = \arg\max_{\mathbf{q} \in \mathcal{D}} \int_{\mathcal{P}} det(\mathbf{F}) \phi(\|p\|) dp \tag{2.18}$$

Without loss of generality we assume that the expected location of the source is at the origin.

Our first main result considers sensors at a fixed distance from the expected source location.

Theorem 2.3.2 (Fixed Distance Sensor Placement) *Assume in Problem 2.3.1 that the region $\mathcal{D}$ is a ring of radius r, centered at the origin so that*

$$\mathbf{q} \in \mathcal{D}_r^M, \ \mathcal{D}_r := \left\{ \mathbf{x} \in \mathbb{R}^2 | \mathbf{x}^T \mathbf{x} = r^2 \right\}, \tag{2.19}$$

and the support of the PDF is a disk with radius smaller than r so that

$$\mathbf{p} \in \mathcal{P}_r, \ \mathcal{P}_r := \left\{ \mathbf{x} \in \mathbb{R}^2 | \mathbf{x}^T \mathbf{x} < r^2 \right\}. \tag{2.20}$$

Then, sensor configurations that are symmetric (2.17) *satisfy the first order necessary conditions for optimality.*

Before stating the proof of Theorem 2.3.2, we provide a lemma which allows the determinant of the Fisher information matrix to be written in a symmetric manner.

Lemma 2.3.3 *The Fisher information matrix* (2.12) *for time-of-arrival measurement differences can be expressed as:*

$$\mathbf{F} = \frac{1}{\sigma^2 \nu^2} \left(\sum_{i=1}^{M} \mathbf{g}_i \mathbf{g}_i^T - M \bar{\mathbf{g}} \bar{\mathbf{g}}^T \right), \tag{2.21}$$

and the determinant of (2.21) *is given by*

$$\begin{aligned} det(\mathbf{F}) = \frac{1}{\sigma^4 \nu^4} \Bigg[& \left(\sum_{i=1}^{M} g_{i,x}^2 - M \bar{g}_x^2 \right) \left(\sum_{i=1}^{M} g_{i,y}^2 - M \bar{g}_y^2 \right) \\ & - \left(\sum_{i=1}^{M} g_{i,x} g_{i,y} - M \bar{g}_x \bar{g}_y \right)^2 \Bigg], \end{aligned} \tag{2.22}$$

where $\bar{\mathbf{g}}$*, defined by*

$$\bar{\mathbf{g}} := \frac{1}{M} \sum_{i=1}^{M} \mathbf{g}_i, \tag{2.23}$$

is the average of the $\mathbf{g}_i$ *vectors.*

Proof: [Proof of Lemma 2.3.3] By induction, beginning with (2.12) *for* $M = 3$,

$$\mathbf{F}_3 = \frac{1}{\nu^2}\mathbf{G}^T\mathbf{Q}^{-1}\mathbf{G} \tag{2.24}$$

$$= \frac{1}{\sigma^2\nu^2}\begin{bmatrix}\mathbf{g}_2 - \mathbf{g}_1 & \mathbf{g}_3 - \mathbf{g}_1\end{bmatrix}\left[\mathbf{I} - \frac{1}{3}\mathbf{1}\mathbf{1}^T\right]\begin{bmatrix}\mathbf{g}_2^T - \mathbf{g}_1^T \\ \mathbf{g}_3^T - \mathbf{g}_1^T\end{bmatrix} \tag{2.25}$$

$$= \frac{1}{\sigma^2\nu^2}\left(\sum_{i=1}^{3}\mathbf{g}_i\mathbf{g}_i^T - 3\left(\frac{1}{3}\right)\sum_{i=1}^{3}\mathbf{g}_i\left(\frac{1}{3}\right)\sum_{j=1}^{3}\mathbf{g}_j^T\right) \tag{2.26}$$

$$= \frac{1}{\sigma^2\nu^2}\left(\sum_{i=1}^{3}\mathbf{g}_i\mathbf{g}_i^T - 3\bar{\mathbf{g}}\bar{\mathbf{g}}^T\right) \tag{2.27}$$

Assuming the lemma holds for M,

$$\mathbf{F}_M = \frac{1}{\nu^2}\mathbf{G}_M^T\mathbf{Q}_M^{-1}\mathbf{G}_M \tag{2.28}$$

$$= \frac{1}{\sigma^2\nu^2}\left(\sum_{i=1}^{M}\mathbf{g}_i\mathbf{g}_i^T - M\bar{\mathbf{g}}\bar{\mathbf{g}}^T\right), \tag{2.29}$$

we examine $M+1$ *letting* $\Gamma = \frac{1}{M+1}$ *and dropping* ν *and* σ *for brevity,*

$$\mathbf{Q}_{M+1}^{-1} = \begin{bmatrix}\mathbf{Q}_M^{-1} + \left(\frac{\Gamma}{M}\right)\mathbf{1}_{M-1}\mathbf{1}_{M-1}^T & -\Gamma\mathbf{1}_{M-1} \\ -\Gamma\mathbf{1}_{M-1}^T & \Gamma M\end{bmatrix} \tag{2.30}$$

$$\mathbf{F}_{M+1} = \begin{bmatrix} \mathbf{G}_M^T & \mathbf{g}_{M+1} - \mathbf{g}_1 \end{bmatrix} \mathbf{Q}_{M+1}^{-1} \begin{bmatrix} \mathbf{G}_M \\ \mathbf{g}_{M+1}^T - \mathbf{g}_1^T \end{bmatrix} \tag{2.31}$$

$$\begin{aligned} &= \mathbf{F}_M + \frac{\Gamma}{M} \mathbf{G}_M^T \mathbf{1}_{M-1} \mathbf{1}_{M-1}^T \mathbf{G}_M \\ &\quad - \Gamma \mathbf{G}_M^T \mathbf{1}_{M-1} (\mathbf{g}_{M+1}^T - \mathbf{g}_1^T) \\ &\quad - \Gamma (\mathbf{g}_{M+1} - \mathbf{g}_1) \mathbf{1}_{M-1}^T \mathbf{G}_M \\ &\quad + \Gamma M (\mathbf{g}_{M+1} - \mathbf{g}_1)(\mathbf{g}_{M+1}^T - \mathbf{g}_1^T) \end{aligned} \tag{2.32}$$

$$= \mathbf{F}_M + \Gamma M \left(\bar{\mathbf{g}}_M - \mathbf{g}_{M+1}\right) \left(\bar{\mathbf{g}}_M - \mathbf{g}_{M+1}\right)^T \tag{2.33}$$

$$\begin{aligned} &= \sum_{i=1}^{M} \mathbf{g}_i \mathbf{g}_i^T + \mathbf{g}_{M+1} \mathbf{g}_{M+1}^T - \Gamma \left[M^2 \bar{\mathbf{g}}_M \bar{\mathbf{g}}_M^T \right. \\ &\quad \left. + \mathbf{g}_{M+1} \mathbf{g}_{M+1}^T + M \bar{\mathbf{g}}_M \mathbf{g}_{M+1}^T + M \mathbf{g}_{M+1} \bar{\mathbf{g}}_M^T \right] \end{aligned} \tag{2.34}$$

$$\begin{aligned} &= \sum_{i=1}^{M+1} \mathbf{g}_i \mathbf{g}_i^T \\ &\quad - \Gamma \left[\left(M \bar{\mathbf{g}}_M + \mathbf{g}_{M+1}\right) \left(M \bar{\mathbf{g}}_M + \mathbf{g}_{M+1}\right)^T \right] \end{aligned} \tag{2.35}$$

$$= \sum_{i=1}^{M+1} \mathbf{g}_i \mathbf{g}_i^T - (M+1) \left[\bar{\mathbf{g}}_{M+1} \bar{\mathbf{g}}_{M+1}^T \right]. \tag{2.36}$$

The determinant of the Fisher information matrix (2.22) *follows directly from the symmetric form of the Fisher information matrix* (2.21)*,*

$$det\,(\mathbf{F}) = \frac{1}{\sigma^4\nu^4} det\left(\sum_{i=1}^{M} \mathbf{g}_i\mathbf{g}_i^T - M\bar{\mathbf{g}}\bar{\mathbf{g}}^T\right). \tag{2.37}$$

■

We are now ready to give the proof of the first theorem.

Proof: [Proof of Theorem 2.3.2] We begin by forming the Lagrangian of (2.18),

$$L(\mathbf{q}, \boldsymbol{\mu}) := \int_{\mathcal{P}_r} \det(\mathbf{F})\phi(\|p\|)dp + \sum_{i=1}^{M} \mu_i \left(\mathbf{q}_i^T\mathbf{q}_i - r^2\right), \tag{2.38}$$

where $\boldsymbol{\mu} \in \mathbb{R}^M$ is a vector of Lagrange multipliers. A point $\mathbf{q}^*$ satisfies the *Karush-Kuhn-Tucker* (KKT) [21] first order necessary optimality conditions for this Lagrangian if there exists a vector $\boldsymbol{\mu}^* \in \mathbb{R}^M$ such that:

$$0 = \mathbf{q}_i^{*T}\mathbf{q}_i^* - r^2 \quad \forall i \in \{1, 2, ..., M\} \tag{2.39}$$

$$\mathbf{0} = \boldsymbol{\nabla}\left[\int_{\mathcal{P}_r} \det(\mathbf{F})\phi(\|p\|)dp + \sum_{j=1}^{M} \mu_j^* \left(\mathbf{q}_j^{*T}\mathbf{q}_j^* - r^2\right)\right], \tag{2.40}$$

where $\boldsymbol{\nabla}$ denotes the differential operator

$$\boldsymbol{\nabla} := \left[\frac{\partial}{\partial \mathbf{q}_i}\right]_{(i=1,\ldots,M)}. \tag{2.41}$$

Condition (2.40) can be written for each i as:

$$2\mu_i \mathbf{q}_i = -\int_{\mathcal{P}_r} \frac{\partial \det(\mathbf{F})}{\partial \mathbf{q}_i} \phi(\|p\|) dp. \tag{2.42}$$

After some algebra, we find using Lemma 2.3.3 that

$$\frac{\partial \det \mathbf{F}}{\partial \mathbf{q}_i} = \frac{2}{\|\mathbf{p} - \mathbf{q}_i\|} \mathbf{g}_i^T \mathbf{F} (\mathbf{g}_i - \bar{\mathbf{g}})^{\perp} \mathbf{g}_i^{\perp}, \tag{2.43}$$

where the perpendicular operator $\perp$ is defined by

$$\mathbf{x}^{\perp} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mathbf{x}, \quad \forall\ \mathbf{x} \in \mathbb{R}^2. \tag{2.44}$$

Symmetric configurations, from Definition 2.2.1, play an important role in the integral in (2.42). To see the symmetry, consider a coordinate rotation in which the i^{th} sensor, $\mathbf{q}_i$, is placed on the positive x-axis. Then take two source

points, $\mathbf{p}$ and $\tilde{\mathbf{p}}$, for which $\tilde{\mathbf{p}}$ is $\mathbf{p}$ mirrored about the x-axis,

$$\tilde{\mathbf{p}} = \mathbf{R}\mathbf{p}, \tag{2.45}$$

where $\mathbf{R}$ is the elementary reflector about the x-axis defined as:

$$\mathbf{R} := \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{2.46}$$

Some algebra reveals

$$\frac{\partial \det \mathbf{F}}{\partial \mathbf{q}_i}\Big|_{(\tilde{\mathbf{p}},\mathbf{q})} = \mathbf{R}\frac{\partial \det \mathbf{F}}{\partial \mathbf{q}_i}\Big|_{(\mathbf{p},\mathbf{q})}. \tag{2.47}$$

To show the optimality of the i^{th} sensor (which now lies on the x-axis), we employ (2.45) to split the integral in (2.42) into two parts. When integrating over points not on the x-axis, for every $\mathbf{p}$ there exists a $\tilde{\mathbf{p}}$ for which the sum of the two integrands have equal and opposite y-components and equal x-components.

$$\frac{\partial}{\partial \mathbf{q}_i}\int_{\mathcal{P}_r} \det \mathbf{F}\phi(\|p\|)dp = \int_{\mathcal{P}_0} \frac{\partial \det \mathbf{F}}{\partial \mathbf{q}_i}\phi(\|p\|)dp + \int_{\mathcal{P}_+} [\mathbf{I} + \mathbf{R}]\frac{\partial \det \mathbf{F}}{\partial \mathbf{q}_i}\phi(\|p\|)dp. \tag{2.48}$$

In the first part, integrated over $\mathcal{P}_0 \subset \mathcal{P}_r$, $\mathbf{p}$ lies on the x-axis, and in the second part, integrated over $\mathcal{P}_+ \subset \mathcal{P}_r$, $\mathbf{p}$ has a positive y-component. Integrating over $\mathcal{P}_0$ is trivially zero because the integrand is zero at all points $\mathbf{p} \in \mathcal{P}_0$ due to the symmetry of the configuration. Notice that in (2.48) the y-component cancels because

$$\mathbf{I} + \mathbf{R} = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}, \tag{2.49}$$

indicating that the right hand side of (2.42) is aligned with $\mathbf{q}_i$. Thus, we conclude that there exists a multiplier,

$$\mu_i = -\int_{\mathcal{P}_+} \frac{\mathbf{q}_i^T}{\|\mathbf{q}_i\|^2} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \frac{\partial \det \mathbf{F}}{\partial \mathbf{q}_i} \phi(\|p\|) dp, \tag{2.50}$$

for each i so that the optimality conditions are satisfied. ■

We next consider a generalization of Theorem 2.3.2 in which the sensors are confined to an annulus of radii $r_1 < r_2$ instead of an r-radius ring, but first we introduce a particular symmetric configuration that will be used throughout the remainder of this chapter.

Definition 2.3.4 (Splay Configuration) *A symmetric configuration is said to be* splay *if*

$$\theta_j - \theta_i = \frac{2\pi}{M}, \text{ for each } j = i + 1, \tag{2.51}$$

where θ_i is the angle of the vector $\mathbf{q}_i$ with respect to the x-axis.

Theorem 2.3.5 (Variable Distance Sensor Placement) *Assume in Problem 4.2.1 that the region $\mathcal{D}$ is an* annulus *of radii $r_1 < r_2$, centered at the origin,*

$$\mathbf{q} \in \mathcal{D}_a^M, \ \mathcal{D}_a := \left\{ \mathbf{x} \in \mathbb{R}^2 | r_1^2 \leq \mathbf{x}^T \mathbf{x} \leq r_2^2 \right\}, \tag{2.52}$$

and the probability density function is truncated at a radius less than r_1 so that

$$\mathbf{p} \in \mathcal{P}_a, \ \mathcal{P}_a := \left\{ \mathbf{x} \in \mathbb{R}^2 | \mathbf{x}^T \mathbf{x} < r_1^2 \right\}. \tag{2.53}$$

Then, sensor configurations on the outer boundary of the annulus,

$$\mathbf{q}_i^T \mathbf{q}_i = r_2^2, \qquad \forall i \in \{1, 2, ..., M\}, \tag{2.54}$$

that are additionally splay (2.51) *satisfy the first order necessary optimality conditions.*

Proof: [Proof of Theorem 2.3.5] The Lagrangian for (2.18) with the inequality constraints (2.52) is now

$$L(\mathbf{q}, \boldsymbol{\mu}, \boldsymbol{\lambda}) := \int_{\mathcal{P}_a} \det(\mathbf{F})\phi(\|p\|)dp + \sum_{i=1}^{M} \mu_i \left(r_1^2 - \mathbf{q}_i^T \mathbf{q}_i\right) + \sum_{i=1}^{M} \lambda_i \left(\mathbf{q}_i^T \mathbf{q}_i - r_2^2\right), \tag{2.55}$$

where $\boldsymbol{\mu} \in \mathbb{R}^M$ and $\boldsymbol{\lambda} \in \mathbb{R}^M$, are vectors of Lagrange multipliers. The KKT first order necessary conditions for optimality are now $\forall i \in \{1, 2, ..., M\}$:

$$0 \geq r_1^2 - \mathbf{q}_i^{*T} \mathbf{q}_i^* \tag{2.56}$$

$$0 \leq \mu_i^* \tag{2.57}$$

$$0 = \mu_i^* \left(r_1^2 - \mathbf{q}_i^{*T} \mathbf{q}_i^*\right) \tag{2.58}$$

$$0 \geq \mathbf{q}_i^{*T} \mathbf{q}_i^* - r_2^2 \tag{2.59}$$

$$0 \leq \lambda_i^* \tag{2.60}$$

$$0 = \lambda_i^* \left(\mathbf{q}_i^{*T} \mathbf{q}_i^* - r_2^2\right) \tag{2.61}$$

$$\mathbf{0} = \boldsymbol{\nabla} L(\mathbf{q}, \boldsymbol{\mu}, \boldsymbol{\lambda}). \tag{2.62}$$

Here, the first six relationships are the primal feasibility, dual feasibility, and complementary slackness conditions to ensure that sensors remain in $\mathcal{D}_a$. Because the sensor configuration in which we are interested have all sensors at the

outer radius r_2 of $\mathcal{D}_a$ (2.54), $\mu_i = 0,\ \forall i \in 1, ..., M$, the relationship (2.62) is identical to (2.40) from the fixed radius problem. Thus we have already shown that symmetric, and hence splay, configurations satisfy (2.62) in the proof of Theorem 2.3.2.

What remains to be shown is that all M Lagrange multipliers, $\lambda_i,\ i = 1, \ldots, M$ have positive sign (2.60). Using (2.43), (2.62) becomes

$$\mathbf{0} = \int_{\mathcal{P}_a} \frac{2}{\|\mathbf{p} - \mathbf{q}_i\|} \mathbf{g}_i^T \mathbf{F} (\mathbf{g}_i - \bar{\mathbf{g}})^\perp \mathbf{g}_i^\perp \phi(\|p\|) dp + 2\lambda_i \mathbf{q}_i. \tag{2.63}$$

To show that each Lagrange multiplier is positive, we solve for λ_i and write the integration in polar coordinates,

$$\lambda_i = \int_0^{r_1} \int_0^{2\pi} \frac{-1}{\|\mathbf{p} - \mathbf{q}_i\| \|\mathbf{q}_i\|^2} \mathbf{g}_i^T \mathbf{F} (\mathbf{g}_i - \bar{\mathbf{g}})^\perp \mathbf{q}_i^T \mathbf{g}_i^\perp dp_\theta \phi(\rho) \rho d\rho, \tag{2.64}$$

where $\mathbf{p} = \rho \angle p_\theta$. Because the radially symmetric density function $\phi(\rho)$ is non-negative, we need only to show that the inner integral is positive for every ρ. Unfortunately, the integrand in (2.64) does not have constant sign; see Fig. 2.3 for example. Because a closed form computation of the integral appears to be intractable, we refer the reader to Fig. 2.4, which shows the inner integral plotted against ρ for various M in a splay configuration. ■

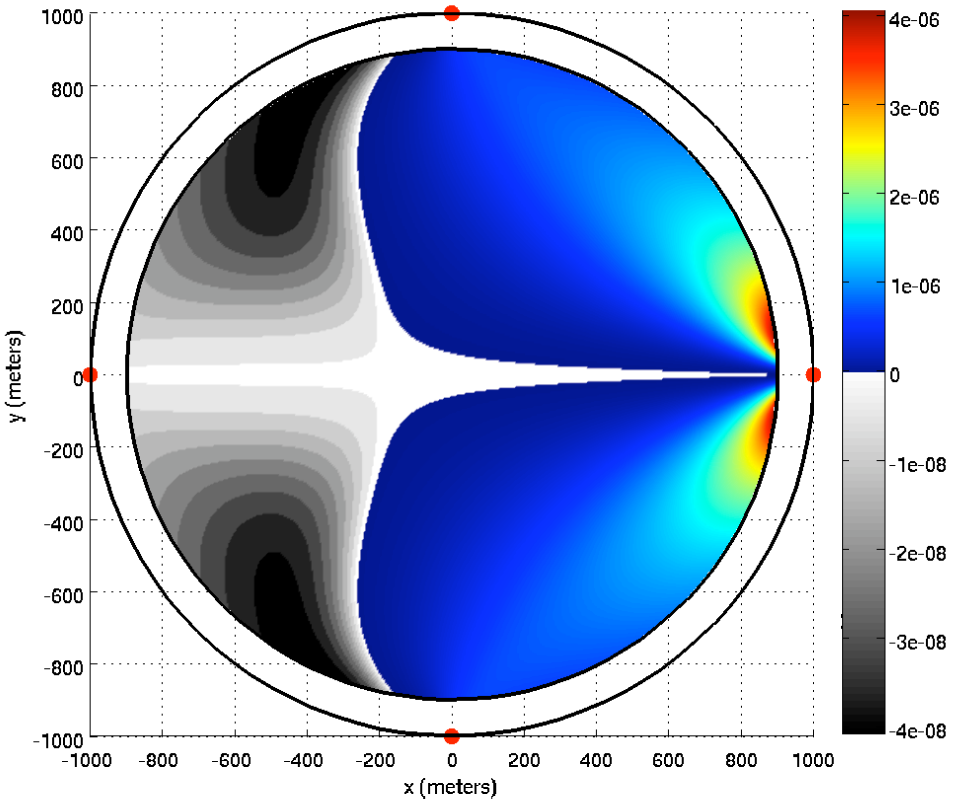


Figure 2.3: The integrand of (2.64) is plotted for $\mathbf{p}$ inside a disk of radius $r_1 = 900$ with $M = 4$ sensors placed in a splay configuration at radius $r_2 = 1000$. The important thing to note here is that the integral will be positive even though the integrand is negative for some $\mathbf{p}$. A nonlinear scaling has been applied to the colormap to show details for negative values of the integrand.

2.4 Numerical Results

In this section we include two numerical results that support Theorem 2.3.5 and motivate a conjecture that is left for future research. In the first, we study the splay configuration found in Theorem 2.3.5 and evaluate how the expected Fisher determinant varies with the outer radius r_2. In the second, we evaluate

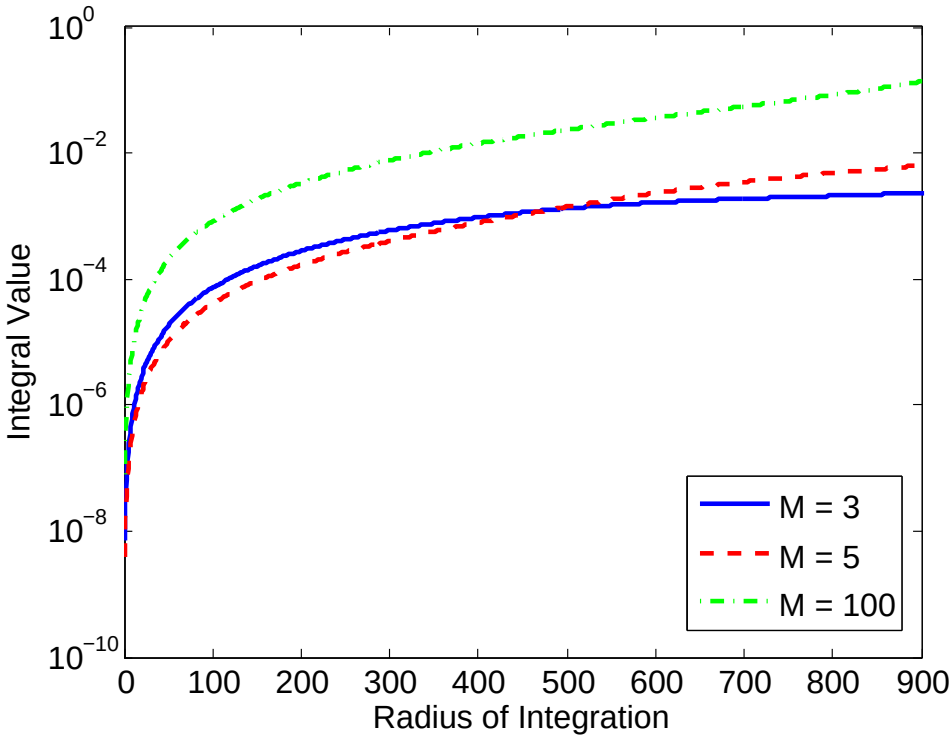


Figure 2.4: The inner integral in (2.64) is plotted vs the radius ρ of $\mathbf{p}$ for M sensors placed in a splay configuration at a radius $r_2 = 1000$. For all ρ, the inner integral evaluates to a positive number indicating that λ_i is positive. As ρ approaches zero the distibution looks more like a point mass, thus the inner integral approaches zero.

the expected Fisher determinant for a subset of configurations found in Theorem 2.3.2.

2.4.1 Expected Fisher Determinant Maximum at the Boundary

In Theorem 2.3.5 we have shown that if the region confining the sensors $\mathcal{D}_a$ is an *annulus* of radius $r_1 < r_2$, centered at the expected location of the source, then sensors placed at the boundary of the region $\mathcal{D}_a$ in a splay configuration meet the first order necessary conditions for optimality. The purpose of this section is to examine how the expected Fisher determinant changes with the outer radius of the boundary. In Fig. 2.5 it can be seen that the expected Fisher determinant increases monontonically with r_2 for both uniform and truncated-Gaussian distributions. This observation supports Theorem 2.3.5 in that if sensors are in a splay configuration about the expected source location, then it is beneficial for all the sensors to move out radially.

2.4.2 Splay Configuration Maximizes Fisher Determinant on Ring

The result of Theorem 2.3.2 states that any symmetric configuration meets the necessary conditions of first order optimality. However, the configurations with sensors at equal angle increments seem to yield higher expected Fisher determinants than other configurations satisfying the symmetry condition.

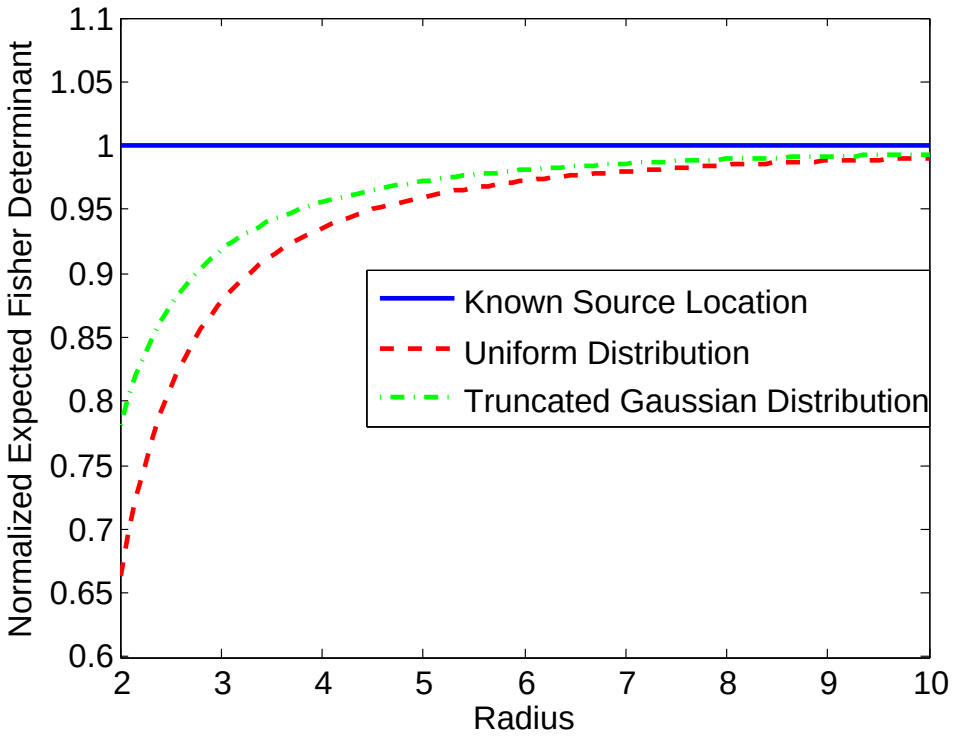


Figure 2.5: The expected value of the Fisher information determinant as a function of outer radius r_2 of region $\mathcal{D}_a$ for $M = 6$ sensors in a splay configuration.

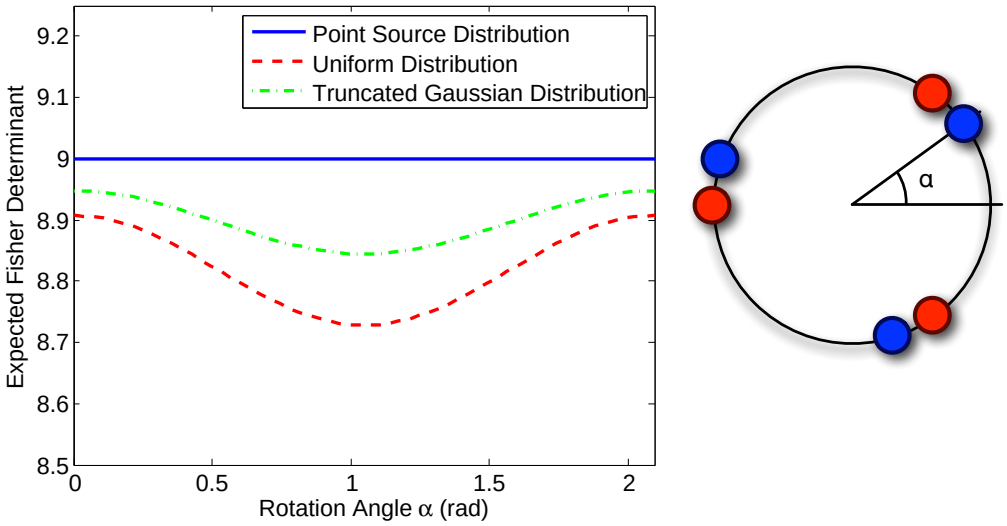
Conjecture 2.4.1 (Splay Conjecture) *Of all the symmetric configurations satisfying the first order necessary conditions in Theorem 2.3.2, splay configurations, from Definition 2.3.4, give the largest expected Fisher determinant.*

Consider the case where $M = 6$, starting in the splay configuration for which (2.15), (2.16), and (2.51) are all satisfied. If we hold the position of three of the even ordered sensors constant while rotating the odd ordered three sensors about the origin by $\alpha \in [0\ 4\pi/M]$, then (2.15) and (2.16) remain satisfied, but the symmetry conditions (2.17) are only satisfied when $\alpha = 0$, $\alpha = 2\pi/M$, and

$\alpha = 4\pi/M$. This rotation by α takes the configurations from an $M = 6$ splay configuration through the point where there are two sensors at each location of an $M = 3$ splay configuration and back to an $M = 6$ splay configuration again. All of these configurations meet the optimality conditions for a known source at the origin [16], but only $\alpha = 0$, $\alpha = 2\pi/M$, and $\alpha = 4\pi/M$ satisfy the symmetry conditions required by Theorem 2.3.2. The results in Fig. 2.6 show that the splay configuration at $\alpha = 0$ and $\alpha = 4\pi/M$ achieve the maximum expected determinant of FIM for both uniform and truncated-Gaussian distributions while the repeated configuration at $\alpha = 2\pi/M$ does not.

2.5 Summary

This chapter has addressed the optimal sensor placement problem considering a source node distributed according to a truncated radially-symmetric probability density function. The results suggest that good sensor configurations are symmetric. Overall, the results show that the distance between the source and the sensor is influential, and that splay configurations are better than other ones. These results are new and were not apparent in previous work that considered a known source location.



(a) Expected Fisher Determinant

(b) $M = 6$ Sensors

Figure 2.6: In (a) the expected value of the Fisher information determinant is plotted as a function of rotation angle α between two sets of $M = 6$ first order optimal configurations. When $\alpha = 2\pi/M$ the sensors are arranged in a configuration of duplicate sensors in an $M = 3$ splay configuration which satifies the first order necessary conditions in Theorem 2.3.5 but does not maximize the expected Fisher determinant. In (b) the $M = 3$ blue sensors are shown rotated by angle α while the $M = 3$ red sensors are held stationary.

Chapter 3

Dynamic Vehicle Routing: Single Source Localization

3.1 Introduction

In this chapter we propose an architecture for data collection and on-the-fly inference in sparse sensor networks where the sensor nodes do not have direct connectivity with each other. A data collector traverses the network, adapting its route as it collects data from each sensor node in order to draw reliable inferences as quickly as possible about the events of interest on which the sensors are reporting. We illustrate our approach for the problem of acoustic source localization. We are interested in homeland security and defense applications in which we wish to monitor large areas (of the orders of 100 square kilometers) with the purpose of detecting and localizing acoustic events, such as explosions or artillery, as quickly as possible. Such events can be detected at large ranges,

so that a density of $\sim 5-6$ sensors/1 km^2 suffices for event detection and localization, provided that we can pool data from multiple sensors. For typical sensor radio range, however, sensors in such a sparse deployment cannot communicate directly with each other. The data mule in this setting is a UAV, which re-optimizes its route as it receives data from each sensor node that it visits, with the goal of localizing the source as quickly as possible.

Relevant Literature

A data mule architecture to transport data in a disconnected sensor network was proposed, and scaling laws for this setting were investigated in [80] and in several other papers on delay-tolerant networking, including [48, 49]. However, to the best of our knowledge, this is the first paper in which the mules process the data they collect in order to guide their decisions, with a view to optimizing a specific application.

A number of algorithms have been proposed for TOA based source localization [1, 7, 25] as well as hybrid TOA-AOA based localization [17, 93]. However, most of these algorithms assume that the TOAs/AOAs are available at a single location and we believe that this is the first attempt to couple a UAV routing protocol that acquires TOAs/AOAs sequentially with the problem of source localization.

The UAV routing problem itself is a combinatorial optimization in that it seeks the "best" sensor sequence from the set of all possible sequences. Classic examples of such combinatorial optimization problems include the traveling salesman problem [91] and the vehicle routing problem [30]. However, our problem differs from static routing problems in that the value of the information held at each *unvisited* sensor depends on the sensors already visited. This leads to a dynamic vehicle routing problem which has been considered in a different context in [12, 70].

Our routing algorithm is aimed at minimizing the expected volume of the Cramér-Rao ellipse. The Cramér-Rao matrix and associated error ellipse has been used extensively as an optimization criteria in the literature. Notable examples include work in the area of sensor selection and optimal observer steering. Information-driven algorithms have been studied in the context of sensor networks [28, 107] and active sensing [50, 75], but the objective differed significantly and a data mule was not present. In [32, 40, 61, 69] the trajectories of mobile sensors are optimized using information based criteria. In these works the mobile agents were the sensors, thus the combinatorial issues of picking sensors were not addressed.

Chapter Organization

In Section 3.2, we describe the algorithms for processing the acoustic information at each sensor node and for routing the UAV. Results from a Monte Carlo simulation study using a flight simulator are presented in Section 3.3. Finally, concluding remarks are provided in Section 3.4.

3.2 Algorithms

We now describe the algorithms needed to accomplish acoustic source localization using a sparse sensor network serviced by an unmanned aerial vehicle. We begin by presenting the TOA and AOA measurement models. We then provide the key ideas behind the UAV routing algorithm that uses Bayesian inference to choose the sequence of sensors to visit based on the data gathered from sensors already visited. The routing algorithm is computationally expensive, so we introduce a "minimal sensor subsets" approach to efficiently compute the optimal UAV route.

3.2.1 TOA Measurements

TOA Model: For the purposes of localization and routing, the output of the k^{th} TOA sensor is modeled as

$$z_k = T_0 + \frac{d_k(\boldsymbol{\theta})}{\nu} + w_z, \tag{3.1}$$

where T_0 is the unknown event time, $d_k(\boldsymbol{\theta})$ is the separation distance between the source located at unknown position $\boldsymbol{\theta}$ and the k^{th} sensor, ν is the speed of sound, and w_z is noise drawn from a zero mean Gaussian distribution with standard deviation σ_z.

The first TOA measurement of an event is not immediately useful in inferring anything about $\boldsymbol{\theta}$ due to the uncertainty in the event time T_0. With two TOA measurements, the uncertainty in the event time can be eliminated by considering the time-difference-of-arrival (TDOA) between two sensors, for example with respect to sensor N_{TOA},

$$y_k := z_k - z_{N_{TOA}} = \frac{d_k - d_{N_{TOA}}}{\nu} + w_y,\ k = 1, \ldots, N_{TOA} - 1. \tag{3.2}$$

The noise w_y is distributed as a zero-mean Gaussian with standard deviation $\sigma_y = \sqrt{2}\sigma_z$.

3.2.2 AOA Measurements

AOA Model: The k^{th} of N_{AOA} AOA sensors produces an angular measurement modeled as

$$s_k = \theta_k + w_s, \tag{3.3}$$

where θ_k is the angle made by the line joining the center of the k^{th} array and the source to magnetic north, and w_s is noise drawn from a zero mean Gaussian[1] distribution with standard deviation σ_s.

3.2.3 UAV Routing

When monitoring a large area in which not all sensors detect the event, the routing algorithm could be initialized by having the UAV follow a fixed route until it encounters a sensor with an event to report. For example, the UAV could be instructed to fly a minimum time circuit, computed using a traveling salesperson solver, to minimize the maximum time between detection of the event and the UAV visit. For large areas, the elapsed time before the source is localized could be dominated by the first detection time if a single UAV is employed. However, timely detection could be ensured by partitioning such areas into smaller sub-areas, each assigned a separate data mule which could

[1] A von Mises distribution is more appropriate here because s_k is an angle, but σ_s is sufficiently small so that the Gaussian distribution is a very good approximation.

follow the algorithm proposed here. Investigation of such scenarios, including algorithms for coordinating multiple UAVs, are beyond the scope of this chapter.

Once the initial detection occurs, the key question to answer is: *in which sequence should the remaining sensors be visited so as to localize the source as quickly as possible*. The routing optimization we propose uses the latest available information about the source in a Bayesian manner. The objective of the UAV routing is to minimize the time required to determine the location of the source (up to some confidence); additional time required to fly over and image the source is not considered. We say that the source has been localized when the area of the region containing the source with a specified confidence level falls below a threshold value. Due to the threshold in the area of the confidence region, we refer to this problem as Threshold Time Minimization (TTM). This problem is challenging due to the nonlinear nature of acoustic source localization. This nonlinearity means that some sensors may carry more information about the source than others, a discrepancy that must be balanced against the time required to service each sensor. Another ramification of this nonlinearity is that sensors close to the source do not generally provide more informative data than faraway sensors. A further complicating factor is that optimal route is data dependent and thus must be computed on the fly.

To explain the routing protocol design problem in more detail, we begin by describing how to compute the confidence region using the Cramér-Rao bound. We then use this confidence region to formally write the routing optimization problem. However, directly solving this optimization problem is intractable, so we then introduce minimal sensor subsets and evaluate the overall computational complexity.

Confidence Ellipse from Cramér-Rao Bound: To avoid coming up with a routing protocol that depends explicitly on the particular estimation algorithm used to compute the source location estimate $\hat{\boldsymbol{\theta}}$ from the available data, we make use of the Cramér-Rao matrix, denoted $\mathcal{C}$, and its determinant in particular. In short, the $\mathcal{C}$ matrix is important because it is the smallest possible covariance any unbiased estimator could give using the available measurements. The Cramér-Rao matrix is defined as the inverse of the Fisher information matrix evaluated at the true value of the parameters [89],

$$\mathrm{F}(\boldsymbol{\theta}, \boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{J}}) \triangleq E_{Y,S|\mathbf{p}} \left[\left(\frac{\partial \log p(\mathbf{y}_{[\boldsymbol{\mathcal{Y}}]}, \mathbf{s}_{[\boldsymbol{\mathcal{J}}]}|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right) \left(\frac{\partial \log p(\mathbf{y}_{[\boldsymbol{\mathcal{Y}}]}, \mathbf{s}_{[\boldsymbol{\mathcal{J}}]}|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right)^T \right] \quad (3.4)$$

$$\mathcal{C}(\boldsymbol{\theta}, \boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{J}}) \triangleq F^{-1}(\boldsymbol{\theta}, \boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{J}}). \quad (3.5)$$

Here, $p(\mathbf{y}_{[\boldsymbol{\mathcal{Y}}]}, \mathbf{s}_{[\boldsymbol{\mathcal{J}}]}|\boldsymbol{\theta})$ is the posterior probability of the available TOA ($\mathbf{y}_{[\boldsymbol{\mathcal{Y}}]}$) and AOA ($\mathbf{s}_{[\boldsymbol{\mathcal{J}}]}$) measurements, respectively, given the source position, $\boldsymbol{\theta}$. The sets $\boldsymbol{\mathcal{Y}}$

and $\mathcal{J}$ contain indices of TOA and AOA measurements that have been collected by the UAV.

For problems in which the likelihood of the parameters given the data is a multivariate Gaussian, as is assumed and later verified here, the Fisher information matrix has a special form. Denoting by $\mathbf{q}_k$ the position of the k^{th} sensor, the Fisher information matrix for TDOA localization can be written as [25],

$$F(\boldsymbol{\theta}, \boldsymbol{\mathcal{Y}}) = \frac{1}{\nu^2} G_{[\boldsymbol{\mathcal{Y}}]}^T Q^{-1} G_{[\boldsymbol{\mathcal{Y}}]}, \tag{3.6}$$

where $Q = \sigma_z^2(\mathbb{I} + \mathbf{1}\mathbf{1}^T)$ of appropriate dimension and

$$G = \begin{bmatrix} \mathbf{g}_1^T - \mathbf{g}_{N_{TOA}}^T \\ \vdots \\ \mathbf{g}_{N_{TOA}-1}^T - \mathbf{g}_{N_{TOA}}^T \end{bmatrix} \quad \text{with} \quad \mathbf{g}_k = \frac{\boldsymbol{\theta} - \mathbf{q}_k}{d_k}, \tag{3.7}$$

and $d_k = \|\boldsymbol{\theta} - \mathbf{q}_k\|$ is the distance between the source and the k^{th} sensor. The notation $G_{[\boldsymbol{\mathcal{Y}}]}$ selects rows of G corresponding to available measurements. Each vector $\mathbf{g}_k$ is a unit vector pointing from the k^{th} sensor towards the source. It is important to note that the Fisher information matrix is not full rank until three non-collinear TOA sensors have been visited.

For the angle of arrival sensors, the Fisher information matrix can be written as [33],

$$F(\boldsymbol{\theta}, \boldsymbol{\mathcal{J}}) = \frac{1}{\nu^2} L_{[\boldsymbol{\mathcal{J}}]}^T R^{-1} L_{[\boldsymbol{\mathcal{J}}]}, \tag{3.8}$$

where R is an appropriately sized identity matrix scaled by σ_s^2 and

$$L = \begin{bmatrix} \mathbf{h}_1^T/d_1 \\ \vdots \\ \mathbf{h}_{N_{AOA}}^T/d_{N_{AOA}} \end{bmatrix} \quad \text{with} \quad \mathbf{h}_k = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \frac{\boldsymbol{\theta} - \mathbf{q}_k}{d_k}. \tag{3.9}$$

The combined Fisher information matrix from all visited AOA and TOA sensors is the sum of the individual 2×2 Fisher information matrices,

$$F(\boldsymbol{\theta}, \boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{J}}) = F(\boldsymbol{\theta}, \boldsymbol{\mathcal{Y}}) + F(\boldsymbol{\theta}, \boldsymbol{\mathcal{J}}), \tag{3.10}$$

from which the Cramér-Rao matrix, $\mathcal{C}$, can be computed using (3.5).

The physical intuition behind the $\mathcal{C}$ matrix can be understood through the notion of a *confidence ellipse* [89]. A confidence ellipse is an ellipsoidal region of the parameter space containing the true parameter value with a specified certainty (or confidence), much like a confidence interval for a one-dimensional variable. The overall shape of the ellipse is determined by the correlation matrix, as given by the estimation error covariance. The utility of the Cramér-Rao

matrix is that it produces a confidence ellipse that fits within the ellipse produced by any unbiased estimator. The volume of the confidence ellipse, denoted V, for a particular confidence level is proportional to the square root of the determinant of the correlation matrix.

Online minimization of the confidence ellipse volume is complicated by the fact that the $\mathcal{C}$ matrix depends on the true source location $\boldsymbol{\theta}$, which is of course unknown. One can instead compute the *expected value* of the volume of the uncertainty ellipse with respect to the posterior probability of the parameters [87],

$$\overline{V}(\boldsymbol{\theta}, \boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{J}}) := E_{P|Y,S}\left[V(\boldsymbol{\theta}, \boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{J}})\right]. \tag{3.11}$$

Threshold Time Minimization UAV Routing Protocol: The Threshold Time Minimization (TTM) protocol aims to minimize the time at which the volume of the expected uncertainty ellipse (3.11) falls below a threshold value, V_{th}. This threshold value represents the largest uncertainty which is determined by the field of view of the camera onboard the UAV and gives the area of the region in which the source is likely to be found with a specified probability.

Sensor data (TDOA and AOA values) available at the current time t are collected in vectors $\mathbf{y}_{[\boldsymbol{\mathcal{Y}}(t)]}$ and $\mathbf{s}_{[\boldsymbol{\mathcal{J}}(t)]}$, and let $\boldsymbol{\mathcal{Y}}(\tau, \mathbf{r})$ and $\boldsymbol{\mathcal{J}}(\tau, \mathbf{r})$ be vectors of indices of sensors whose data will be available at time $\tau \geq t$ along route $\mathbf{r}$.

Here, a route is a sequence of sensors to be visited by each UAV. The resulting optimization problem to accomplish this objective can be written as follows:

$$\begin{aligned} \min_{\mathbf{r}\in\mathcal{R}(t)} \quad & \tau \\ \text{subject to} \quad & \overline{V}\left(\boldsymbol{\theta}, \boldsymbol{\mathcal{Y}}(\tau, \mathbf{r}), \boldsymbol{\mathcal{J}}(\tau, \mathbf{r})\right) - V_{th} \leq 0, \end{aligned} \tag{3.12}$$

where $\mathcal{R}(t)$ is the set of all routes to unvisited sensors, and the expected uncertainty volume is computed using measurements that will be available at time τ along route $\mathbf{r}$.

The numerical computation needed to fully carry out the routing optimization can be burdensome due to the fact that the general nature of the posterior probability makes closed form computation of the expected value in (3.11) intractable. The expected value we are interested in evaluating is with respect to the posterior probability of the parameters given the data,

$$\mathcal{E} = \int p(\boldsymbol{\theta}|\mathbf{y}, \mathbf{s}) f(\boldsymbol{\theta}) dp, \tag{3.13}$$

for some function f. A standard approximation technique is to consider K samples drawn from the posterior $p(\boldsymbol{\theta}|\mathbf{y}, \mathbf{s})$, denoting the k^{th} sample by $\mathbf{m}_k$.

Then, the expected value is well approximated by

$$\mathcal{E} \approx \frac{1}{K} \sum_{k=1}^{K} f(\mathbf{m}_k). \tag{3.14}$$

This approximation approaches the true value as K becomes large by the weak law of large numbers.

Drawing samples from a posterior distribution is nontrivial, but a number of techniques like Markov Chain Monte Carlo (MCMC) are well documented in the literature [3,46]. The approach we employ is a bio-inspired MCMC technique based on a model of the motion of E. coli bacteria known as Optimotaxis [63]. Viewing the posterior probability density as a food source, the bacteria wander using a stochastic tumble-and-run mechanism that allows their positions to be seen as samples drawn from the posterior. Increasing the number of bacteria results in a better approximation of the expected value at the cost of computational resources. It is important to note that the posterior probability density changes after each measurement is collected, and thus it is necessary to resample the posterior frequently.

3.2.4 Minimal Sensor Subsets for Efficient UAV Routing

Due to the combinatorial nature of the routing problem, the optimization over all possible routes in (3.12) grows as $N!$. Exhaustively searching each and every route quickly becomes intractable. Here, we introduce the concept of "minimal sensor subsets" to efficiently find the best route. The key insight here is that the expected value of the volume of the uncertainty ellipse in (3.11) depends only on which sensors have been visited, not on the particular order in which the sensors were visited. For example, the uncertainty volume after completing route $\{6, 1, 4\}$ will be the same as that for route $\{4, 1, 6\}$, although the time required to perform these two routes could differ significantly. This special structure allows the problem of selecting sets of sensors to visit to be decoupled from the problem of selecting the order in which to visit the sensors.

Accordingly, the minimal sensor subsets approach to UAV routing proceeds in two main steps:

Step 1 A small number of unordered sets of sensors (i.e. minimal sensor subsets), one of which necessarily contains the sensors visited by optimal route, are identified using a method to be described shortly.

Step 2 The optimal route is computed for each minimal sensor subset, using exact or approximate techniques. Among these optimal routes, the UAV selects the one that can be serviced in the least amount of time.

We proceed by formally introducing minimal sensor subsets, explaining the two steps of the minimal sensor subsets approach to UAV routing, and then analyzing the complexity of the problem.

Minimal Sensor Subsets: A single minimal sensor subset is a collection of *unvisited* sensors which, when combined with information provided by sensors already visited, reduces the expected volume of the uncertainty ellipse below the user-specified threshold, V_{th}. Further, this set is minimal in the sense that the exclusion of any one unvisited sensor would increase the uncertainty volume above the threshold. More formally, we provide the following definition.

Definition 3.2.1 *Minimal Sensor Subset*

Let $\mathcal{V} \subset \{1, 2, \ldots, N\}$ be the set of sensors already visited, $\mathcal{U} = \{1, 2, \ldots, N\} \setminus \mathcal{V}$ be the set of unvisited sensors, $n = |\mathcal{U}|$ be the number of unvisited sensors, and let u be a nonempty subset of the unvisited sensors, $u \subseteq \mathcal{U}$. Denote by u^{-k} the set created by removing the k^{th} element from set u. Then, the set u is said to be minimal *if the following two conditions are met.*

1. *The volume of the uncertainty ellipse after visiting all of the sensors in the set $\mathcal{V} \cup u$ is below the volume threshold, V_{th}.*

2. *The volume of the uncertainty ellipses after visiting all but any one of the unvisited sensors, e.g. $\mathcal{V} \cup u^{-k}$ for all $k = 1, \ldots, |u|$, are all above the volume threshold, V_{th}.*

Note that the optimal UAV route is necessarily a permutation of one of the minimal sensor subsets. Visiting a subset of a minimal sensor subset would not provide enough information whereas visiting a superset would be unnecessary. We now explain an algorithm to find all minimal sensor subsets (Step 1), and then select the TTM-optimal route (Step 2).

Algorithm Step 1: Sensor subsets have a natural hierarchy, see Fig. 3.1, that allows us to make inferences that reduce the number of uncertainty volume computations that need to be made compared to an exhaustive search of the power set of unvisited sensors, $\mathcal{U}$. Specifically, every superset of a set of sensors having an expected uncertainty ellipse volume below the threshold will have an expected uncertainty ellipse volume below the threshold (adding sensors can only reduce uncertainty volume). Similarly, all subsets of a set of sensors having an expected uncertainty ellipse volume above the threshold will have an

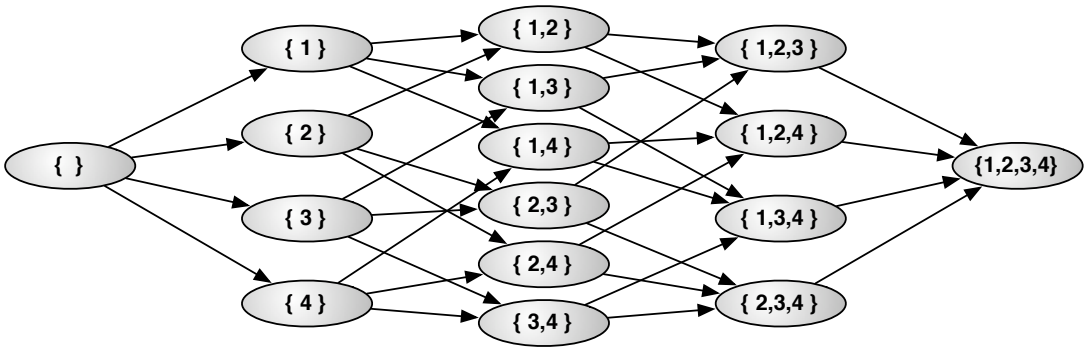


Figure 3.1: An example of the hierarchy of unvisited sensor subsets for the case of $n = 4$ unvisited sensors. The algorithm to find all minimal sensor subsets begins by evaluating the uncertainty volume of the column containing $\{1, 2\}$, making inferences after each and every evaluation.

expected uncertainty ellipse volume above the threshold (removing sensors can only increase uncertainty volume).

The algorithm begins by evaluating the uncertainty volume of all sets of unvisited sensors of size $\lfloor n/2 \rfloor$. Sets for which the expected uncertainty volume, when combined with information from already visited sensors, are above (below) the threshold are denoted with a plus (minus). Inferences are made after each evaluation by marking all supersets (subsets) with a plus (minus) accordingly. The few sets remaining without any plus or minus after evaluating all sets of size $\lfloor n/2 \rfloor$ are evaluated exhaustively, but again inferences are made after each uncertainty volume evaluation. When the algorithm terminates, we will know if

the expected uncertainty volume is above or below the threshold for every sensor subset, and thus can easily identify which are minimal.

Algorithm Step 2: Once the minimal subsets have been identified, the second step is to compute the optimal (minimum time) route for each minimal subset (note that routes do not need to be computed for non-minimum sets of sensors). To do so, we used the symmetric TSP solver *linkern* available at [4], which uses the Chained Lin-Kernighan Heuristic from [5]. For each minimal set, the solver takes as input the distances between the sensors in the minimal subset and returns the optimal order in which to visit the sensors and the route completion time. The fastest route through any one of the minimal subsets is the optimal route in the TTM optimization (3.12), with the possible exception that the TSP solver may produce a suboptimal route. In practice, we have found the approximations provided by TSP sufficient and have not performed exhaustive search.

Complexity Analysis: The computational savings of using the minimal sensor subsets approach to route determination are dramatic. There are at most 2^n unique subsets of $n = |\mathcal{U}|$ *unvisited* sensors, and most ${}_nC_{\lfloor n/2 \rfloor}$ of these subsets can be minimal[2]. The worst case performance of the algorithm to find minimal sensor subsets, described in Step 1 above, computes the uncertainty volume for

[2]Here ${}_nC_k$ denotes the number of combinations of n elements, selected k at a time.

approximately half of the sets (and makes inferences for the other half). In particular, we have the following result.

Theorem 3.2.1 *The number of uncertainty value calculations used in finding all minimal sensor subsets will not be greater than*

$$\begin{cases} 2^{n-1} + \frac{\Pi_{k=n/2+1}^{n} k}{2\Pi_{k=1}^{n/2} k} & \text{if } n \text{ even} \\ 2^{n-1} + {}_nC_{\lfloor n/2 \rfloor} & \text{if } n \text{ odd.} \end{cases} \tag{3.15}$$

Proof: [Proof of Theorem 3.2.1] The main idea is that the worst case for the procedure described in Step 1 of the algorithm above occurs when there is just one minimal sensor subset. There are at most 2^n unique subsets of $n = |\mathcal{U}|$ *unvisited* sensors. Let $f(n)$ represent the number of uncertainty value calculations used in finding all minimal sensor subsets.

n **even:** For n even, there are ${}_nC_{n/2}$ sets of size $n/2$. The algorithm begins by searching these ${}_nC_{n/2}$ sets of size $n/2$, from which inferences can be made about all sets containing fewer (more) than $n/2$ elements, provided the lone minimal sensor subset contains more (fewer) than $n/2$ elements. At the completion of this step $2^{n-1} - 1/2\left({}_nC_{n/2}\right)$ sets remain to be searched. The upper bound comes from assuming that the algorithm will have to exhaustively search all remaining sets (i.e. that no further inferences are made), which is clearly an

upper bound on the number of evaluations that would actually be performed.

$$f(n) \leq ({}_nC_{n/2}) + 2^{n-1} - \frac{1}{2}({}_nC_{n/2}) \tag{3.16}$$

$$= 2^{n-1} + \frac{1}{2}({}_nC_{n/2}) \tag{3.17}$$

$$= 2^{n-1} + \frac{\Pi_{k=n/2+1}^{n} k}{2\Pi_{k=1}^{n/2} k} \tag{3.18}$$

n **odd:** For n odd, there are ${}_nC_{\lfloor n/2 \rfloor}$ sets of size $\lfloor n/2 \rfloor$ and ${}_nC_{\lceil n/2 \rceil}$ sets of size $\lceil n/2 \rceil$. The algorithm begins by searching the ${}_nC_{\lfloor n/2 \rfloor}$ sets of size $\lfloor n/2 \rfloor$, from which inferences can be made about all sets containing fewer (more) than $\lfloor n/2 \rfloor$ elements, provided the lone minimal sensor subset contains more (fewer) than $\lfloor n/2 \rfloor$ elements. At the completion of this step 2^{n-1} sets remain to be searched. The upper bound comes from assuming that the algorithm will have to exhaustively search all remaining sets (i.e. that no further inferences are made).

$$f(n) \leq ({}_nC_{\lfloor n/2 \rfloor}) + 2^{n-1} \tag{3.19}$$

■

As compared to exhaustively evaluating the cost of $n!$ routes, we compute at most ${}_nC_{\lfloor n/2 \rfloor}$ TSP solutions, after finding minimal sensor subsets. Using Chrisofides TSP approximation algorithm [27], the running time for each TSP

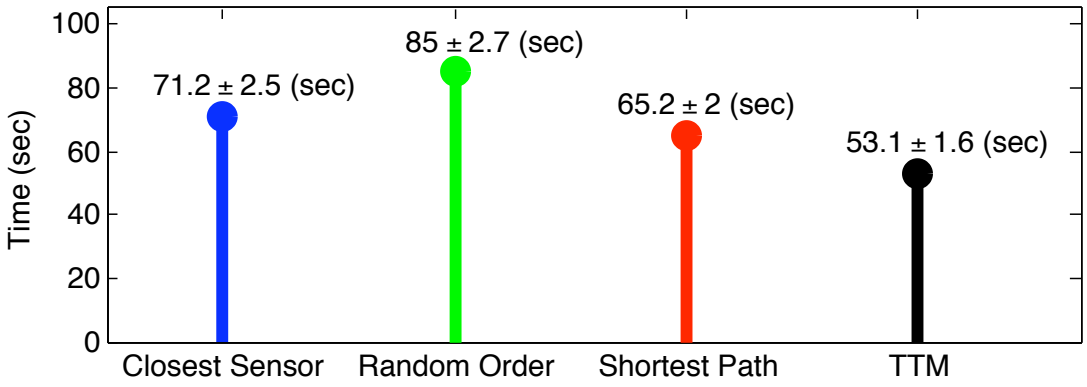


Figure 3.2: Monte Carlo simulation results comparing the TTM algorithm against three other routing protocols.

is $O(\eta_k^3)$, where $\eta_k \leq n$ is the number of sensors in the k^{th} minimal subset. It should be noted that the minimal sets approach efficiently provides results corresponding to an infinite planning horizon, thereby avoiding the need to consider a limited planning horizon for computational tractability.

3.3 Numerical Results

To gain a better understanding of the performance of the UAV routing protocol, we used the high-fidelity simulation platform FlightGear. FlightGear offers realistic vehicle dynamics, including effects from wind. The particular vehicle model we simulated was a Sig Rascal 110, capable of an airspeed of 22 m/s.

The results of 500 simulation trials are shown in Fig. 3.2 for four routing protocols that differ in how the next sensor is selected: (1) *Closest Sensor* greedily chooses the nearest sensor, (2) *Random Order* chooses a random sensor, (3) *Shortest Path* is a traveling salesperson tour, and (4) *Threshold Time Minimization* is the algorithm presented in this work. The closest sensor protocol visits sensors that are close together, and thus have less information compared to sensors that are widely-spread. The fact that the localization time is large is not surprising due to the fact that the optimization is myopic and does not consider value of the sensor information. The random order protocol takes the most amount of time to localize the source but visits relatively few sensors. This is attributed to the fact that the randomly selected sensors tend to be wide-spread, and thus have high information content although they take a long flight time to reach. The shortest path protocol localizes the source fairly quickly but tends to visit many sensors. As with the myopic closest sensor protocol, the information value of the data contained at each sensor is not considered. The threshold time minimization protocol results in a localization time that is on average 12 seconds faster than the shortest path protocol. This is a direct result of the optimization algorithm seeking to balance the utility of the information of far away sensors with the cost to reach those sensors. For each simulation trial, one AOA sensor and seven TOA sensors were randomly placed in a 1 km by 1 km area, and the

source was placed randomly in a 700 m by 700 m area with each area centered at the origin.

3.4 Conclusions

We have introduced the novel concept of data mules which adapt their future actions based upon on-the-fly inference from data that they collect. The routing protocol selects the sequence of sensors to visit that minimizes the time to reduce the expected value of the volume of the uncertainty ellipsoid to below a specified threshold. We have demonstrated, through flight simulation, that this approach is effective for rapid acoustic source localization using sparsely deployed sensors when compared to three other protocols.

Chapter 4

Algorithms for the Traveling Salesman Problem with Neighborhoods Involving a Dubins Vehicle

4.1 Introduction

Research in the area of unmanned aerial vehicles (UAV) has evolved in recent years. There is rich literature covering various areas of autonomy including path planning, trajectory planning, task allocation, cooperation, sensing, and communications. As the mission objectives of UAVs have increased in complexity and importance over the last few years, problems are starting to arise at the intersection of these disciplines. The Dubins Traveling Salesman Problem with Neighborhoods (DTSPN) combines the problem of path planning with trajectory planning while using neighborhoods to represent communication ranges or sensor

footprints. In this problem the UAV simply needs to enter a region surrounding each objective waypoint.

Relevant Literature

The path planning problem seeks to determine the optimal sequence of waypoints to visit in order to meet certain mission objectives while minimizing costs, such as the total length of the mission [20,53]. Path planning problems typically rely on approximating the cost of the mission by the length of the solution to an Euclidean Traveling Salesman Problem (ETSP), where the cost to travel from one waypoint to the next is approximated by the Euclidean distance between the two waypoints. This approximation simplifies the overall optimization but may lead to UAV routes that are far from optimal because the aircraft dynamics are not considered.

Another area of UAV research is trajectory planning, in which the goal given an initial and final waypoint pair is to determine the optimal control inputs to reach the final waypoint in minimum time given kinematic constraints of the aircraft. In 1957, Dubins showed that for an approximate model of aircraft dynamics, the optimal motion between a pair of waypoints can be chosen among six possible paths [34]. Similar results were proven later in [19] using tools from optimal control theory. In [81], the authors propose a means of choosing the optimal Dubins path without computing all six possible Dubins optimal paths.

A significant amount of research has gone into combining the problems of motion planning and path planning [57–59, 74, 77]. In these works, the dynamics of the UAV are taken into consideration by using the Dubins model when determining the optimal sequence of waypoints. This problem is typically referred to as the Dubins Traveling Salesman Problem (DTSP).

A third area of UAV related research is a version of path planning that takes into account the communication range of the aircraft or the sensor footprint of the aircraft. This problem is best described as a Traveling Salesman Problem with Neighborhoods (TSPN). Now, not only does one determine a sequence of regions but also an entry point at each region. Many researchers have addressed this problem with various regions, but most have used the Euclidean distance as the cost function [35, 36, 105]. Obermeyer was the first to tackle the TSPN with Dubins vehicle dynamics in [67] using a genetic algorithm approach, then later in [68] by using a sampling based roadmap method, which we will call RCM, that is proven to be resolution complete. In the latter method, the DTSPN is transformed into a General Traveling Salesman Problem (GTSP) with non-overlapping nodesets, and then to an Asymmetric Traveling Salesmen Problem (ATSP) through a version of the Noon and Bean transformation [66].

Contributions

Chapter 4. Algorithms for the Traveling Salesman Problem with Neighborhoods Involving a Dubins Vehicle

In this chapter, we propose two algorithms to approximate the DTSPN. The first involves the case in which the regions do not overlap. In this case, we present a simple algorithm requiring just one sample configuration in each region. The second algorithm addresses the case in which the regions intersect frequently. In this case, we propose a sampling based roadmap algorithm similar to that of [68] but use a more general version of the Noon and Bean transformation [65] in which the GTSP can contain overlapping nodesets. We show that for the same set of samples this method will produce a tour that is no longer than that of RCM from [68] and performs significantly better when the regions overlap.

Chapter Organization

The remainder of this chapter is organized as follows. In Section 4.2, the Dubins Traveling Salesman Problem with Neighborhoods is formally introduced. An approximation algorithm for the DTSPN when the regions do not overlap is described in Section 4.3. Section 4.4 describes a second algorithm for the DTSP when the regions overlap. In Section 4.5, we present a numerical study comparing our algorithm with an existing algorithm for various sized regions and various amounts of overlap. Conclusions and future work are discussed in Section 4.6.

4.2 Problem Statement

The dynamics of the UAV can be approximated by the Dubins vehicle in the plane. The state of the Dubins vehicle $\mathbf{X}$ can be represented by the triplet $(x, y, \theta) \in SE(2)$, where $(x, y) \in \mathbb{R}^2$ define the position of the vehicle in the plane and $\theta \in \mathbb{S}^1$ defines the heading of the vehicle. The vehicle dynamics are then written as,

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \nu \cos(\theta) \\ \nu \sin(\theta) \\ \frac{\nu}{\rho} u \end{bmatrix}, \tag{4.1}$$

where ν is the forward speed of the vehicle, ρ is the minimum turning radius, and $u \in [-1, 1]$ is the bounded control input. Let $\mathcal{C}_\rho : SE(2) \times SE(2) \to \mathbb{R}_+$ associate the length $\mathcal{C}_\rho(\mathbf{X}_1, \mathbf{X}_2)$ of the minimum length path from an initial configuration $\mathbf{X}_1$ of the Dubins vehicle to a final configuration $\mathbf{X}_2$, subject to the kinematic constraints in (4.1). This length, which we will refer to as the Dubins distance from $\mathbf{X}_1$ to $\mathbf{X}_2$, can be computed in constant time [81].

Let $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, ..., \mathcal{R}_n\}$ be set of n compact regions in a compact region $\mathcal{Q} \subset \mathbb{R}^2$, and let $\mathbf{\Sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ be an ordered permutation of $\{1, \ldots, n\}$. Define a projection from $SE(2)$ to $\mathbb{R}^2$ as $\mathcal{P} : SE(2) \to \mathbb{R}^2$, i.e. $\mathcal{P}(\mathbf{X}) = [x\ y]^T$,

and let $\mathbf{P}_i$ be a point in $SE(2)$ whose projection lies in $\mathcal{R}_i$. We denote the vector created by stacking all n configurations $\mathbf{P}_i$ as $\mathbf{P} \in SE(2)^n$.

The DTSPN involves finding the minimum length tour in which the Dubins vehicle visits each region in $\mathcal{R}$ while obeying the kinematic constraints of (4.1). This is an optimization over all possible permutations $\mathbf{\Sigma}$ and configurations $\mathbf{P}$. Stated more formally:

Problem 4.2.1 (DTSPN)

$$\begin{aligned} \underset{\mathbf{\Sigma},\mathbf{P}}{\textit{minimize}} \quad & \mathcal{C}_\rho(\mathbf{P}_{\sigma_n}, \mathbf{P}_{\sigma_1}) + \sum_{i=1}^{n-1} \mathcal{C}_\rho(\mathbf{P}_{\sigma_i}, \mathbf{P}_{\sigma_{i+1}}) \\ \textit{subject to} \quad & \mathcal{P}(\mathbf{P}_i) \in \mathcal{R}_i,\ i = 1, \ldots, n. \end{aligned}$$

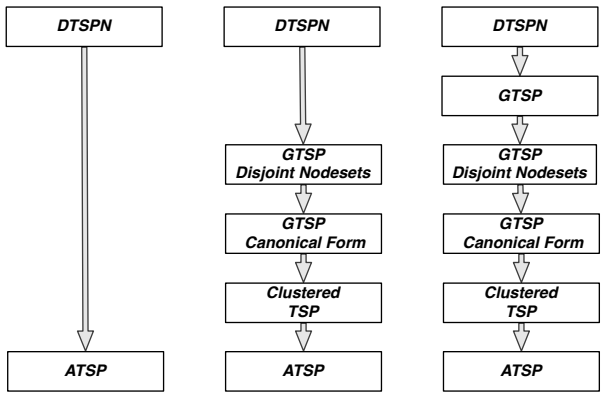
We present two algorithms to address this problem. Each algorithm involves generating a set of $m \geq n$ sample configurations $\mathcal{S} = \{\mathbf{S}_1, \ldots, \mathbf{S}_m\}$ such that,

$$\mathcal{P}(\mathbf{S}_k) \in \bigcup_{i=1}^{n} \mathcal{R}_i, \tag{4.2}$$

$$\forall i\ \exists k \text{ s.t. } \mathcal{P}(\mathbf{S}_k) \in \mathcal{R}_i. \tag{4.3}$$

Each algorithm approximates Problem 4.2.1 by finding the best sample configurations $\mathbf{P} \subseteq \mathcal{S}$ and the order $\mathbf{\Sigma}$ in which to visit them.

4.3 DTSPN with Disjoint Regions



(a) Algorithm 1. (b) RCM [68]. (c) Algorithm 2.

Figure 4.1: Three algorithms for DTSPN of increasing generality: (a) disjoint regions and disjoint nodesets, (b) overlapping regions and disjoint nodesets, and (c) overlapping regions and overlapping nodesets.

Consider the instance of Problem 4.2.1 where all the regions are disjoint. In this simpler case, we consider an algorithm that finds a feasible solution with a length that is within a factor (dependent on the minimum distance separating the regions) of the length of the optimal solution that is independent of n. This algorithm samples from each region a single Dubins configuration, computes the $n(n-1)$ Dubins distances between those configurations, and uses these costs to form a cost matrix $\mathbf{D}$ used by the ATSP solver, see Algorithm 1. The process of taking a single sample per region allows one to proceed directly to the ATSP step, as can be seen in Figure 4.1(a).

Algorithm 1 DTSPN($\rho, \mathcal{R}$)

Require: An asymmetric TSP aproximation algorithm ATSP($\mathbf{D}$), and a function to compute the optimal Dubins distance between two configurations $\mathcal{C}_\rho(\mathbf{X}, \mathbf{X}')$.

Ensure: A sequence $\hat{\mathbf{\Sigma}}$ of waypoint configurations $\hat{\mathbf{P}}$ for DTSPN over the set of disjoint regions $\mathcal{R}$.

1: **for** $i = 1$ to n **do**
2: randomly sample $\mathbf{S}_i$ s.t. $\mathcal{P}(\mathbf{S}_i) \in \mathcal{R}_i$.
3: **end for**
4: **for** $i = 1$ to n **do**
5: **for** $j = 1$ to n **do**
6: set $\mathbf{D}(i, j) := \mathcal{C}_\rho(\mathbf{S}_i, \mathbf{S}_j)$.
7: **end for**
8: **end for**
9: set $\hat{\mathbf{\Sigma}}$:=ATSP($\mathbf{D}$), $\hat{\mathbf{P}} := \mathbf{S}$
10: **return** $\hat{\mathbf{\Sigma}}$, $\hat{\mathbf{P}}$

Define ϵ and μ as the minimum and maximum Euclidean distances between any two regions in $\mathcal{R}$,

$$\epsilon := \min_{i \neq j} \min_{(\mathbf{X}, \mathbf{X}') \in \mathcal{R}_i \times \mathcal{R}_j} \|\mathcal{P}(\mathbf{X}) - \mathcal{P}(\mathbf{X}')\|_2, \tag{4.4}$$

$$\mu := \max_{i \neq j} \min_{(\mathbf{X}, \mathbf{X}') \in \mathcal{R}_i \times \mathcal{R}_j} \|\mathcal{P}(\mathbf{X}) - \mathcal{P}(\mathbf{X}')\|_2, \tag{4.5}$$

respectively. Define the diameter d_i of region $\mathcal{R}_i$, as the maximum Euclidean distance between any two points in the region $\mathcal{R}_i$,

$$d_i := \max_{(\mathbf{X},\mathbf{X}')\in\mathcal{R}_i\times\mathcal{R}_i} \|\mathcal{P}(\mathbf{X}) - \mathcal{P}(\mathbf{X}')\|_2, \tag{4.6}$$

and define d as the maximum diameter of all the regions,

$$d := \max_{i\in\{1,2,\ldots,n\}} d_i. \tag{4.7}$$

Theorem 4.3.1 (Worst case performance for Algorithm 1) *Given $\rho > 0$, and $n \geq 2$ disjoint regions $\mathcal{R}$ whose maximum diameter is d and separated by a least $\epsilon > 0$, Algorithm 1 will produce a DTSPN tour that is within the following factor of the optimum DTSPN tour. If an exact ATSP solver is used, the factor will be,*

$$\lambda := \left(\frac{2d + \mu + \kappa\rho\pi}{\epsilon}\right). \tag{4.8}$$

If instead the modified Christofides algorithm from [41] is used, the factor will be

$$\lambda \min\left(\log n, \frac{3}{2}\gamma\right) \tag{4.9}$$

where $\kappa \in [2.657, 2.658]$ *and*

$$\gamma := \left(1 + \frac{\kappa\rho\pi}{\epsilon}\right), \tag{4.10}$$

comes from [58].

The proof of Theorem 4.3.1 will make use of Theorem 3.4 from [77], which bounds the Dubins distance as,

$$\mathcal{C}_\rho(\mathbf{P}, \mathbf{P}') \leq \|\mathcal{P}(\mathbf{P}) - \mathcal{P}(\mathbf{P}')\|_2 + \kappa\pi\rho. \tag{4.11}$$

Proof: [Proof of Theorem 4.3.1] The Euclidean distance between any two configurations $\mathbf{P}_i$ and $\mathbf{P}_j$ is at most $2d + \mu$. Using (4.11),

$$\mathcal{C}_\rho(\mathbf{P}_i, \mathbf{P}_j) \leq 2d + \mu + \kappa\pi\rho. \tag{4.12}$$

Let $\mathbf{P}^*$ and $\mathbf{\Sigma}^*$ denote the optimal tour. Then, the length of the each leg of the tour provided by Algorithm 1 satisfies

$$\frac{\mathcal{C}_\rho(\hat{\mathbf{P}}_{\hat{\sigma}_i}, \hat{\mathbf{P}}_{\hat{\sigma}_{i+1}})}{\mathcal{C}_\rho(\mathbf{P}^*_{\sigma^*_i}, \mathbf{P}^*_{\sigma^*_{i+1}})} \leq \left(\frac{2d + \mu + \kappa\rho\pi}{\epsilon}\right) = \lambda, \tag{4.13}$$

because $\mathcal{C}_\rho(\mathbf{P}^*_{\sigma^*_i}, \mathbf{P}^*_{\sigma^*_{i+1}}) \geq \epsilon$.

The total tour length for a given ρ, $\mathcal{R}$, $\mathbf{\Sigma}$, and $\mathbf{P}$ is computed by summing Dubins distances along the tour,

$$\mathcal{L}_{\rho,\mathcal{R}}(\mathbf{\Sigma}, \mathbf{P}) := \mathcal{C}_\rho(\mathbf{P}_{\sigma_n}, \mathbf{P}_{\sigma_1}) + \sum_{i=1}^{n-1} \mathcal{C}_\rho(\mathbf{P}_{\sigma_i}, \mathbf{P}_{\sigma_{i+1}}), \tag{4.14}$$

so the length of the tour produced by Algorithm 1 satisfies

$$\mathcal{L}_{\rho,\mathcal{R}}(\hat{\mathbf{\Sigma}}, \hat{\mathbf{P}}) \leq \lambda \mathcal{L}_{\rho,\mathcal{R}}(\mathbf{\Sigma}^*, \mathbf{P}^*). \tag{4.15}$$

The above result holds if an exact TSP solver is used and matches the condition given in (4.8). If using an approximate ATSP solver, the bound will be slightly weaker. For example the modified Christofides' algorithm from [41] has a multiplicative error bound of $\min\left(\log n, \frac{3}{2}\gamma\right)$, resulting in an approximate DTSPN solution with the combined approximation factor given in (4.9). ■

Given n samples from n regions, this algorithm will compute the ATSP over n nodes. The worst case computational complexity of filling the ATSP cost matrix is $O(n^2)$. Then the worst case complexity for solving the ATSP using the modified version of Christofides' algorithm provided in [41] is $O(n^3)$.

4.4 DTSPN with Intersecting Regions

4.4.1 Algorithm 2

For the case in which the regions overlap, we propose a more general algorithm that seeks to take advantage of the possibility of visiting several regions at a single vehicle configuration. Algorithm 2 approximates the solution to Problem 4.2.1 by converting the DTSPN to a GTSP with overlapping nodesets, as shown in Figure 4.1(c), by sampling regions $\mathcal{R}$ with a finite set of m Dubins vehicle configurations $\mathbf{S}$. See Fig 4.2 for an example of converting the DTSPN to a GTSP with overlapping nodesets. The GTSP is then transformed into a standard ATSP through the Noon and Bean transformation [65]. A variety of solvers are available for ATSP.

The GTSP can be described with a directed graph with nodes $\mathcal{N}$ and arcs $\mathcal{A}$ where the nodes are members of predefined nodesets $\mathcal{S}$. Here each node represents an element of the vector of sampled configurations $\mathbf{S}$, and the arc connecting node $\mathbf{S}_i$ to node $\mathbf{S}_j$ represents the length of the minimum length path for a Dubins vehicle $c_{i,j} = \mathcal{C}_\rho(\mathbf{S}_i, \mathbf{S}_j)$ from configuration $\mathbf{S}_i$ to configuration $\mathbf{S}_j$. The nodeset $\mathcal{S}_k$ corresponding to region $\mathcal{R}_k$ contains all samples whose projection lies in $\mathcal{R}_k$, $\mathcal{S}_k := \{\mathbf{S}_i \mid \mathcal{P}(\mathbf{S}_i) \in \mathcal{R}_k\}$ for $i \in \{1, 2, \ldots, m\}$. The objective of the

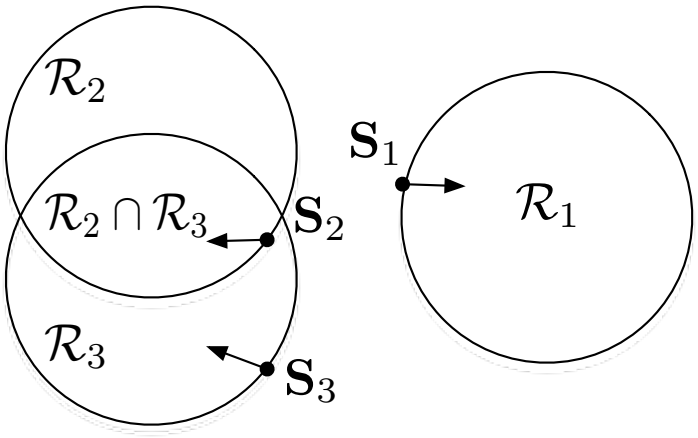
GTSP is to find a minimum cost cycle passing through each nodeset exactly one time.

4.4.2 Noon and Bean Transformation

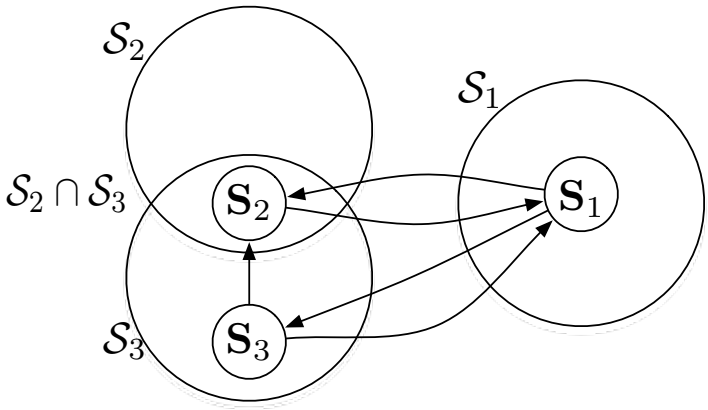
What follows is a brief summary of the Noon-Bean transformation from [65] as it is used in this work. The transformation is best described in three stages.

The first stage converts the GTSP to a GTSP with mutually exclusive nodesets. This is done by first eliminating any arcs from $\mathcal{A}$ that do not enter at least one new nodeset. Next, a finite cost $\alpha \geq \sum_{(i,j)\in\mathcal{A}} c_{i,j}$ is added to each arc cost for each new nodeset the arc enters. Next, any nodes that belong to more than one nodeset are duplicated and placed in different nodesets so as to allow each node to have membership in only one nodeset. Any arcs to and from the original nodes are duplicated as well. In addition, zero cost arcs are added between all the spawned nodes of each multiple membership node. The large cost α added to all the other arcs ensures that all spawned nodes will be visited consecutively, if at all.

The second stage takes the GTSP with mutually exclusive nodesets and eliminates any intraset arcs, leaving a GTSP in "canonical form." The third stage of the transformation converts the canonical GTSP to a "clustered" TSP as follows. The nodes in each nodeset are first enumerated. Then, a zero cost cycle



(a) Example instance of DTSPN with three circular regions and samples $\mathbf{S}_1$, $\mathbf{S}_2$, and $\mathbf{S}_3$.



(b) "GTSP with overlapping nodesets" from Algorithm 2 with edges only entering new nodesets.

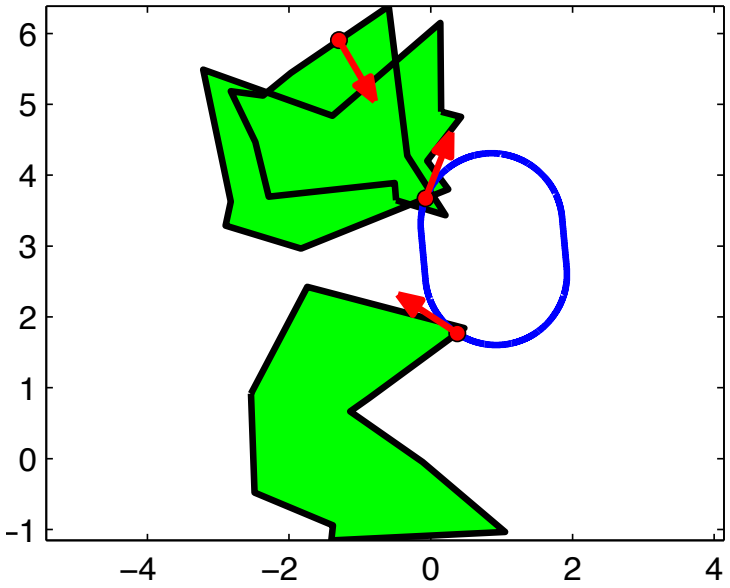
Figure 4.2: Example DTSPN with the corresponding "GTSP with overlapping nodesets".

is created for each nodeset by adding zero cost edges between consecutive nodes in each nodeset and connecting the first node to the last. The *interset* edges are then shifted so they emanate from the previous node in its cycle. Finally, the clustered TSP is converted to an ATSP by adding a finite cost $\beta \geq \sum_{(i,j)\in\mathcal{A}} c_{i,j}$ to each intercluster arc cost.

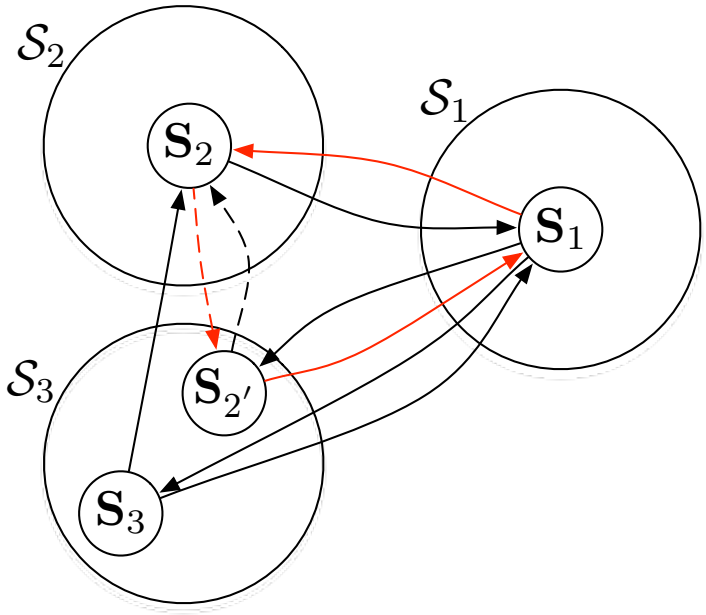
4.4.3 Performance Comparison

Algorithm 2 is very similar to the Resolution Complete Method (RCM) proposed in [68] with the key exception that we use the fact that visiting one of the samples in the intersection of multiple regions achieves the goal of visiting all the regions. Figures 4.3 and 4.4 illustrate this key difference. The RCM requires mutually exclusive nodesets for the conversion from DTSPN to a GTSP with disjoint nodesets, as depicted in Figure 4.1(b). To meet this requirement, samples are assigned directly to the nodeset of the region from whose boundary they are drawn. If multiple regions overlap and a sample lies in the intersection, Algorithm 2 assigns this sample to all the nodesets corresponding to all the intersecting regions, while RCM does not. Algorithm 2 then uses this additional information in the optimization.

Theorem 4.4.1 (Algorithm 2 performance) *Given $\rho > 0$, the set of $n \geq 2$ possibly intersecting regions, $\mathcal{R}$, and the set of m sample configurations, $\mathcal{S}$, let*

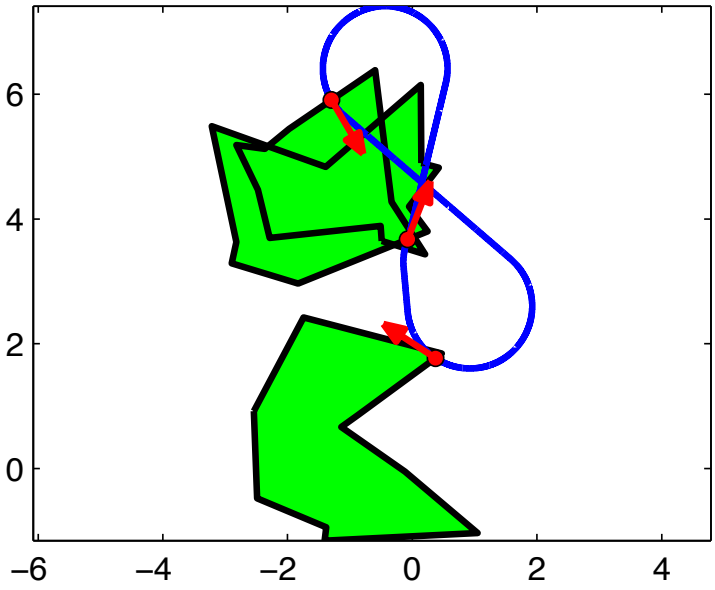


(a) Example Tour: Algorithm 2, Tour Length = 7.7.

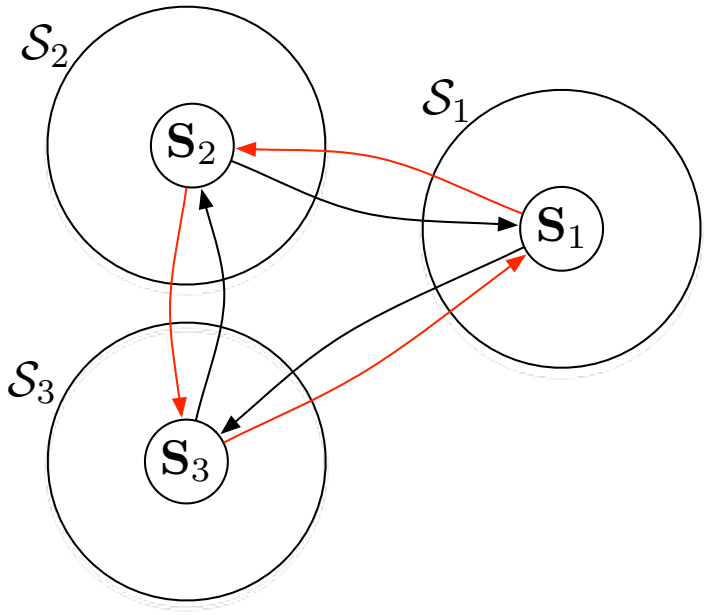


(b) "GTSP with mutually exclusive nodesets" from Algorithm 2 with duplicated node $\mathbf{S}_{2'}$ and zero cost edges between $\mathbf{S}_2$ and $\mathbf{S}_{2'}$.

Figure 4.3: Example DTSPN instance with corresponding "GTSP with mutually exclusive" nodesets from Algorithm 2.



(a) Example Tour: RCM, Tour Length = 15.4.



(b) "GTSP with mutually exclusive nodesets" from RCM with no duplicated node $\mathbf{S}_{2'}$ and nonzero cost edge from $\mathbf{S}_2$ to $\mathbf{S}_3$.

Figure 4.4: Example DTSPN instance with corresponding "GTSP with mutually exclusive" nodesets from RCM.

T_{A2} and T_{RCM} denote the tours produced by Algorithm 2 and the RCM [68], respectively. Then the length of T_{A2} is no greater than that of T_{RCM},

$$\text{length}(T_{A2}) \leq \text{length}(T_{RCM}). \tag{4.16}$$

Proof: [Proof of Theorem 4.4.1] Let $T = \{\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_n\}$ be a feasible tour, and note that both Algorithm 2 and RCM minimize the tour length plus an additive constant while ensuring that all regions are visited. The difference is that Algorithm 2 may produce tours visiting fewer than n unique samples, should some samples lie in the multiple regions. In particular, Algorithm 2 ensures that each leg of the tour enters at least one new region, by construction. Therefore, in performing the optimization Algorithm 2 will either consider T, or subset of T, in which samples at the end of legs not entering an unvisited region have been removed. Due to the Dubins distance function satisfying the triangle inequality [101], a tour that visits a redundant sample will be longer than a tour that visits a subset of the samples. The optimal tour T_{A2} cannot be longer than T_{RCM}, because both optimize over the same set of feasible tours except for the tours in which Algorithm 2 bypasses these unneeded samples. ■

The property *resolution complete method* as used in [68], dictates that the method converges to a solution at least as good as any nonisolated optimum solution as the number of sample configurations goes to infinity.

Corollary 4.4.2 (Algorithm 2 is Resolution Complete) *Given $\rho > 0$, the set of $n \geq 2$ possibly intersecting regions, $\mathcal{R}$, and the set of m sample configurations, $\mathcal{S}$ drawn from a Halton quasirandom sequence [44] as in RCM, Algorithm 2 is Resolution Complete.*

Proof: [Proof of Corollary 4.4.2] From [68], the RCM is a resolution complete method and converges as the number of samples goes to infinity, and from Theorem 4.4.1, we have shown that for the same set of sample configurations Algorithm 2 will produce a tour that is no longer than RCM. ■

4.4.4 Complexity of Algorithm 2

We have provided an algorithm that takes advantage of sample configurations that lie in overlapping regions, and we have shown that this algorithm produces a tour that is no longer than the previous best algorithms in the literature. However, the size of the ATSP is increased by the number of multiple nodeset duplicate nodes. Given m samples from n regions, this algorithm will compute the ATSP over at most mn nodes. The worst case computational com-

plexity of the Noon and Bean transformation [65] is $O(m^2n^4)$. Then the worst case complexity for solving the ATSP using the modified version of Christofides' algorithm provided in [41] is $O(m^3n^3)$.

4.5 Numerical Results

In Theorem 4.4.1 we have shown that for the same sample set, Algorithm 2 will perform no worse than the resolution complete method from [68], but at the cost of solving a larger ATSP problem when there exist samples that are contained in multiple regions. In this section, we use Monte Carlo simulation to investigate the level of performance improvement that can be gained by using Algorithm 2.

The centers of circular regions of variable but homogeneous diameters are randomly placed in a square of variable side length. By varying both the size of regions and the area in which the centers of the regions are confined we are able to vary the degree of overlap. The turning radius of the UAV ρ is set to unit radius. To solve for the tours we used the symmetric TSP solver *linkern* available at [4], which uses the Chained Lin-Kernighan Heuristic from [5]. The radii of the circular regions were varied over $\{0.5, 0.75, 1.00, \ldots, 5.5\}$, and the length of the sides of the square were varied over $\{5, 5.5, \ldots, 15\}$. For the first

test, we ran 100 trials where 10 regions were randomly placed in the bounding box and 50 samples were drawn from the boundaries of the regions. In a second test, we repeated the same test parameters with 100 samples drawn from the boundaries of the regions.

The results can be seen in Figures 4.5 and 4.6 respectively, where the ratio of the average length of the tours found by the Algorithm 2 to those found by RCM are displayed for each test configuration. In both instances, it is clear that for small regions and large bounding box (bottom right of plots), there is little to no overlap, and the two algorithms perform equivalently. The tests of interest are when the regions grow, and the bounding area shrinks (moving from bottom right to top left). For these cases we see that on average, Algorithm 2 finds tours that are nearly half the length of RCM.

4.6 Summary

This chapter introduced two algorithms addressing the Dubins Traveling Salesman Problem with Neighborhoods. When the neighborhoods or regions of interest do not intersect, we have shown that Algorithm 1 achieves an approximation factor independent of the number of regions with a worst case complexity of $O(n^3)$ due to the approximation of the ATSP. The more general case where

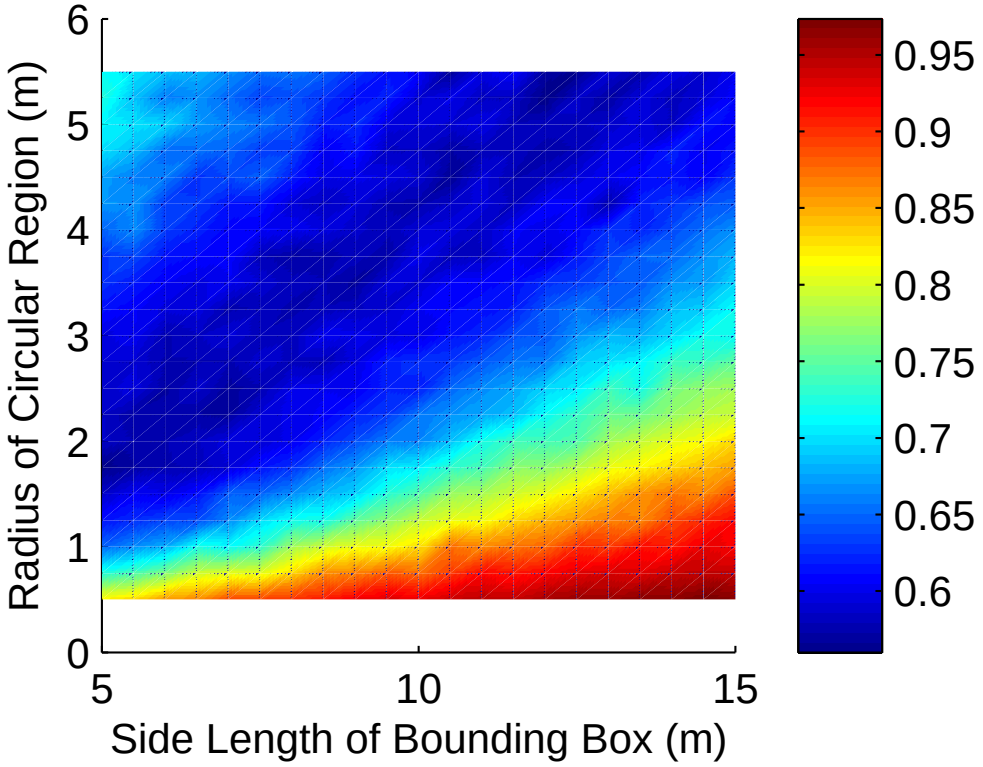


Figure 4.5: Simulation results for the 50 sample test.

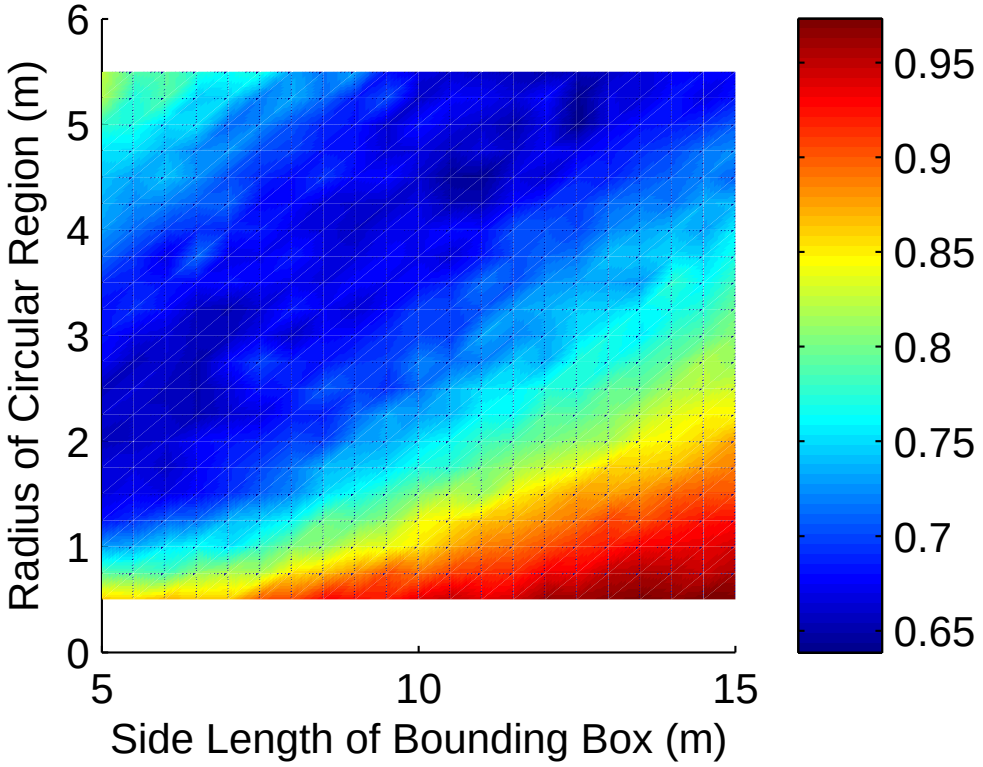


Figure 4.6: Simulation results for the 100 sample test.

the regions of interest intersect frequently was addressed in Algorithm 2. This algorithm samples the regions and then relies on the Noon and Bean transformation [65] for overlapping nodesets to transform the problem to an ATSP. We show that for the same set of samples this method will produce a tour that is no longer than that of [68] and presented numerical results that show performance improvement when there is overlap in the regions of interest.

Chapter 5

Dynamic Vehicle Routing With A Large Sparse Sensor Network

5.1 Introduction

We study the problem of Dynamic Vehicle Routing (DVR) in the context of a *data mule*, sometimes called a mobile base station, for a sparse disconnected sensor network. In this problem, the service vehicle travels between sparsely deployed unattended ground sensors of known locations in order to gather measurements and makes decisions about future actions based upon those measurements. The particular application studied here is the acoustic source localization problem in which time of arrival measurements from at least three sensors are required for estimation of the location of the acoustic event. When events occur according to a spatio-temporal Poisson point process, they are detected by sensors located within a given detection range. It is assumed in this work that

sensors are placed in such a way to provide minimum necessary coverage of the region (i.e. at least three sensors can detect events anywhere in the region). The events are considered localized and are removed from the system at the instant the service vehicle has gathered at least three measurements corresponding to the event. We provide policies for the service vehicle that seek to minimize the expected time that events remain in the system. The sequence in which the vehicle visits the sensors can have a large impact on the overall system performance. A dynamic policy that efficiently searches the space and seeks further information about a region only after a detection is shown to perform well in the low traffic regime where events occur rarely. Conversely, in the high traffic regime where events occur very often, a static policy that attempts to visit all sensors in minimum time is shown to perform well.

Relevant Literature

This work is most similar to [37, 38] in that both works consider targets which must be detected by limited range sensing before they can be serviced. The key difference is that in the previous work the limited range sensor is mounted on the service vehicle while in our work the sensors are placed at discrete points in the region. In the low traffic regime we essentially have a discrete version of the search portion of the previous work. Another key difference is in how the events are serviced once they are detected. In [37, 38], upon detection, events

must be directly visited by the service vehicle, where as in this work, the events are considered serviced when the vehicle has visited at least three sensors who have detected the events.

Another notable line of research with a similar problem setup pertains to the symmetric cyclic queues from [10, 42, 85]. This work considers multiple independent queues where customers arrive according to a Poisson process. The queues are serviced by a mobile server in cyclic order with a random on-site service time for each customer. In our work, multiple customers can be serviced simultaneously, thus eliminating the queueing aspect of the problem.

The vehicle routing problem (VRP) in which a service vehicle or vehicles must visit a set of goals in a minimum amount of time is a well studied problem, see [30]. Variants of this problem include the pick up and delivery problem studied in [76], and the VRP with time windows studied in [31,43,83]. Problems belonging to this class of combinatorial optimization problems have the characteristic that all the information is known a-priori, thus fixed policies can be computed offline. For a more complete survey of the vehicle routing problem, see [86].

The dynamic vehicle routing (DVR) problem, where complete information is not known a-priori requires routes to be altered as information is revealed to the service vehicle was introduced in [73]. The dynamic traveling repairperson

problem (DTRP) introduced by [11–13] provides the basis for much of the work on the DVR problem over the past twenty years. Several variants of this problem have been studied such as DVR with time windows and customer impatience as in [43, 70], the dynamic pick up and delivery problem in [9, 95], and the decentralized case was considered in [39]. The concept of partially dynamic vehicle routing was explored in [8, 55]. The DVR problem has received a large amount of attention in recent years; we are therefore only able to list a subset of relevant literature in this area. Please see [22] for a recent comprehensive survey.

Another related branch of literature includes research on *data mules*, sometimes called mobile base stations, that visit stationary sensors in a sensor network to harvest data. In [80], data mules were modeled as mobile nodes that moved randomly in an environment transferring data from stationary nodes to access points when possible. Non random message ferrying paths were introduced in [108]. An underwater sensor network was demonstrated in [90] that utilized autonomous underwater vehicles to service and collect data from underwater sensors. In [15], routes are planned for the message ferry that guarantee it will visit each mobile node with a certain probability even though the locations of the mobile nodes are unknown to the service vehicle. The DTRP is studied in [24] under the context that customers are serviced when the mobile collector enters a neighborhood around the customer in which communication is possible.

Other recent work has focused on path planning as well as data transmission; some examples include [14, 29, 84].

Contributions

The contribution of this chapter pertains to the dynamic vehicle routing problem over a sparse sensor network for the purpose of acoustic source localization. First, we propose and analyze a pair of policies and provide a criteria for switching between them based upon the occurrence rate of acoustic events. The first policy, we call the Search Based (SB) policy, attempts to explore the coverage area, searching for possible detections, and upon detection performs a local search to service nearby events. Both lower and upper bounds are computed for the performance of this policy. The second policy, we call the Traveling Salesman Based (TSB) policy, performs a static traveling salesperson tour through all the sensors in the network. An upper bound is computed for the performance of the TSB policy and is used along with the lower bound from the SB policy to provide a switching criteria. Second and finally, a numerical example is provided to illustrate the system performance in comparison to the theoretical bounds.

Chapter Organization

The remainder of this chapter is arranged as follows. In the next section, the dynamic vehicle routing problem over a sparse sensor network is formalized and background information is given regarding the acoustic source localization

problem. In Section 5.3, two routing policies are described along with analysis of performance and criteria for switching between policies based on the occurrence rate of events. A numerical example is presented in Section 5.4, and concluding remarks follow.

5.2 Problem Statement and Notation

This DVR problem is set in a large region containing n sensors, each equipped with a means to localize and perform time synchronization, e.g. via GPS. The sensors do not communicate directly with each other due to large separation distances, terrain or urban signal blockages, and/or concerns for battery conservation. Instead, data from each sensor will be collected by a service vehicle. The service vehicle must travel sufficiently close to the sensor to enable a brief communication channel over which the sensor data is communicated. We will refer to this communication distance as r_c. The objective is to choose for the service vehicle a sequence of sensors to service so as to enable quick identification of a vector of unknown constant parameters. In the acoustic source localization problem that we will focus on in this chapter, multiple acoustic sources generate transient acoustic events. These events arrive in the system according to a temporal Poisson process with intensity $\lambda > 0$, and the location of the sources are

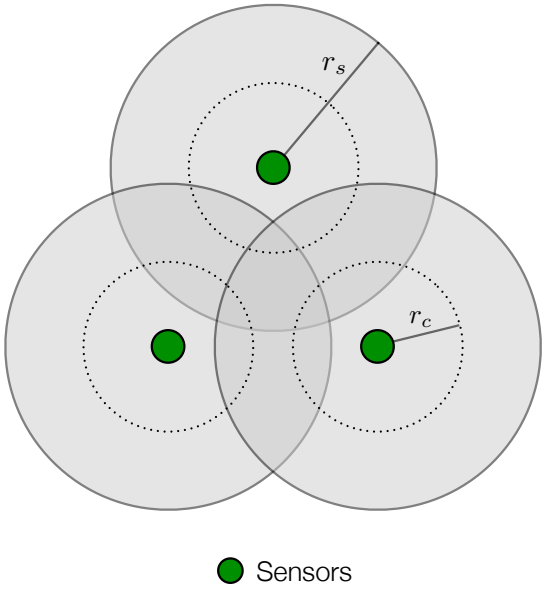


Figure 5.1: Sensors can detect acoustic events and thus measure the TOA for events occuring within the sensing radius r_s, and the sensors can only communicate with the service vehicle when it comes within the communication radius r_c.

uniformly distributed in the region. We assume that a sensor can detect with certainty events occurring within a known sensing range r_s and cannot detect an event outside this range, and that r_s is sufficiently larger than the communication radius r_c as shown in Fig. 5.1.

For the j^{th} acoustic event the uncertain parameters are the x and y coordinates of the source. The data communicated between the k^{th} sensor and the service vehicle is the time of arrival (TOA) measurement z_{jk} of the j^{th} acoustic disturbance. This TOA measurement is modeled as $z_{jk} = t_0 + d_{jk}/\nu$, where t_0

is the unknown event time, d_{jk} is the separation distance between the source of the j^{th} event and the the k^{th} sensor, and ν is the speed of sound. A single TOA measurement is not immediately useful in inferring anything about the location of the source due to the uncertainty in the event time. With two TOA measurements, the event time uncertainty can be eliminated by taking the time difference of arrival (TDOA) between two sensors, for example with respect to the n^{th} sensor,

$$y_{jk} := z_{jk} - z_{jn} = \frac{d_{jk} - d_{jn}}{\nu}. \tag{5.1}$$

which describes the hyperbola of points in the plane where the j^{th} event could have originated. The source location can then be determined by intersecting two such hyperbolas (as in Fig. 5.2) by combining measurements from three sensors. Many authors have addressed the problem of estimating a single source location from TOA measurements e.g., [25,78,79]. In this work we are interested in localizing many acoustic sources at once. This problem was addressed by [82,92]). For the remainder of this paper we will assume that an algorithm such as the one provided in [92] is available to localize multiple events when at least three TOA measurements have been obtained for each event. We assume that once measurements from three sensors detecting the event have been transmitted to the service vehicle that the event is served instantaneously. Note that several

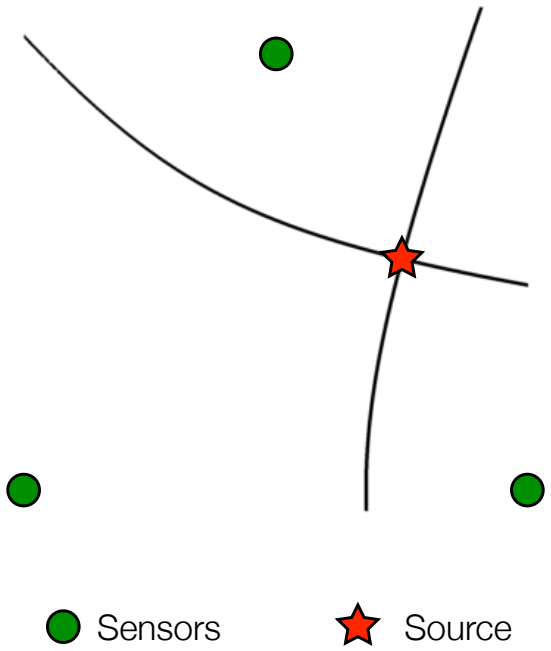
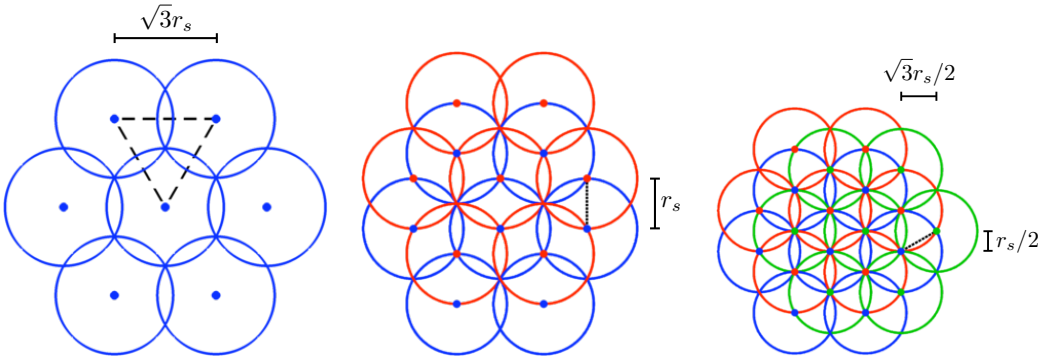


Figure 5.2: Acoustic source localization by intersection of hyperbolas.

events can be served at the same instant if a particular sensor has made multiple measurements.

In order to guarantee that at least three sensors detect events from anywhere in the region, the sensors are placed in such a way as to provide three-coverage of the region. From Theorem 4 in [106], sensors should be placed on an equilateral triangular lattice of spacing $\sqrt{3}r_s$ to achieve one coverage with minimum overlap as depicted in blue in Fig. 5.3(a). To achieve two coverage a one cover configuration can be duplicated and shifted in the y-direction by r_s as depicted in red in Fig. 5.3(b). To provide three coverage an additional one coverage



(a) One Coverage. (b) Two Coverage. (c) Three Coverage.

Figure 5.3: Sensor spacing for three-coverage is achieved by first placing sensors in a one-coverage configuration on a triangular lattice with spacing $\sqrt{3}r_s$ as in (a), then duplicating this configuration and shifting by r_s in the y-direction as in (b), and then duplicating again and shifting by $r_s/2$ in the y-direction and $\sqrt{3}r_s/2$ in the x-direction as in (c).

configuration can be added and shifted by $\sqrt{3}r_s/2$ in the x-direction and $r_s/2$ in the y-direction as depicted in green in Fig. 5.3(c). The combination of all sensors (blue, red, and green) results in three-coverage with each sensor on an equilateral triangular lattice of spacing r_s.

The service vehicle is modeled as a first-order integrator with unit speed and must travel directly over the sensor to receive the measurement as in Fig. 5.4. We make use of traveling salesman problem solutions as part of our policies. In general the traveling salesman problem is NP-complete [52]. However, in [71] the authors show that the Traveling Salesman Problem on solid triangular grid graphs (as considered here) is solvable in polynomial time. For the remainder of

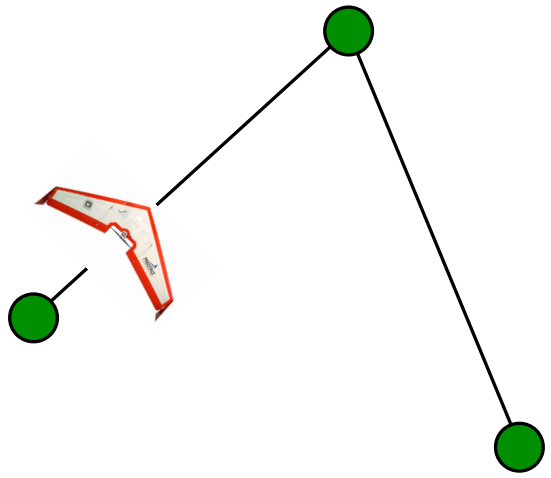


Figure 5.4: The service vehicle is modeled as a first-order integrator with constant speed.

this work, we will assume that for a given set of points on a triangular lattice, the TSP tour sequence and the TSP tour length are known.

This problem is set in a bounded convex region of interest $\mathcal{E} \subset \mathbb{R}^2$ with area $\mathcal{A}$. Given a collection of n sensors positioned in such a way as to provide minimal three-coverage of the area, let $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$ denote this set of sensors, and note that all sensors in $\mathcal{S}$ can detect an event at position e with probability one if the event occurs within a distance of the sensing radius r_s. Let $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n\}$ be a set of n circular regions with radius r_s, where $\mathcal{C}_i$ denotes the region centered at sensor s_i. Let $\mathcal{S}_{one} \subset \mathcal{S}$ denote the subset of sensors that provide one-coverage of $\mathcal{E}$, as in Fig. 5.3(a). Let $\mathcal{S}_{int} \subset \mathcal{S}_{one}$ denote the subset of one-coverage sensors whose sensing region is contained fully in the

region of interest, $\mathcal{S}_{int} := \{s_i \in \mathcal{S}_{one} \mid \mathcal{C}_i \subset \mathcal{E}\}$, and let $\mathcal{S}_{bdry} \subset \mathcal{S}_{one}$ denote the set of one-coverage sensors whose sensing region intersects the boundary of the region of interest, $\mathcal{S}_{bdry} := \{s_i \in \mathcal{S}_{one} \mid \mathcal{C}_i \not\subset \mathcal{E}\}$. Let $\mathcal{C}^{one} \subset \mathcal{C}$, $\mathcal{C}^{int} \subset \mathcal{C}^{one}$, and $\mathcal{C}^{bdry} \subset \mathcal{C}^{one}$ denote the subset of sensing regions corresponding to $\mathcal{S}_{one}$, $\mathcal{S}_{int}$, and $\mathcal{S}_{bdry}$ respectively. Let m, with $m < n$, denote the cardinality of both $\mathcal{S}_{one}$ and $\mathcal{C}^{one}$, m_{int} denote the cardinality of both $\mathcal{S}_{int}$ and $\mathcal{C}^{int}$, and m_{bdry} denote the cardinality of both $\mathcal{S}_{bdry}$ and $\mathcal{C}^{bdry}$. Let $\mathcal{Q}(t) \subset \mathcal{E}$ denote the set of positions of all acoustic events that have yet to be localized until time t, and $N(t)$ the cardinality of $\mathcal{Q}(t)$. The event originating at $q_i \in \mathcal{Q}$ arrives into the system at $t_{q_i(arrives)}$ as in Fig. 5.5(a) and is serviced and thus removed from the set $\mathcal{Q}$ at $t_{q_i(localized)}$, the instant the service vehicle visits the third sensor to have detected the event as in Fig. 5.5(b). Let W_i denote the time that event e_i at q_i is active in the system,

$$W_i := t_{q_i(localized)} - t_{q_i(arrives)}. \tag{5.2}$$

Let u denote a function which we will refer to as a policy, which outputs the next sensor to visit whenever the service vehicle is at a given sensor location. Let $\mathcal{U}$ denote the set of stabilizing policies in the sense that when the service

vehicle executes policy $u \in \mathcal{U}$, the steady state expected wait time is finite,

$$\limsup_{i \to +\infty} \mathbb{E}\left[W_i(u)\right] < \infty. \tag{5.3}$$

Equivalently, u is said to be a stabilizing policy if under its action the number of outstanding events in the system $N(t)$ remains bounded over time,

$$\limsup_{t \to +\infty} \mathbb{E}\left[N(t)\right] < \infty. \tag{5.4}$$

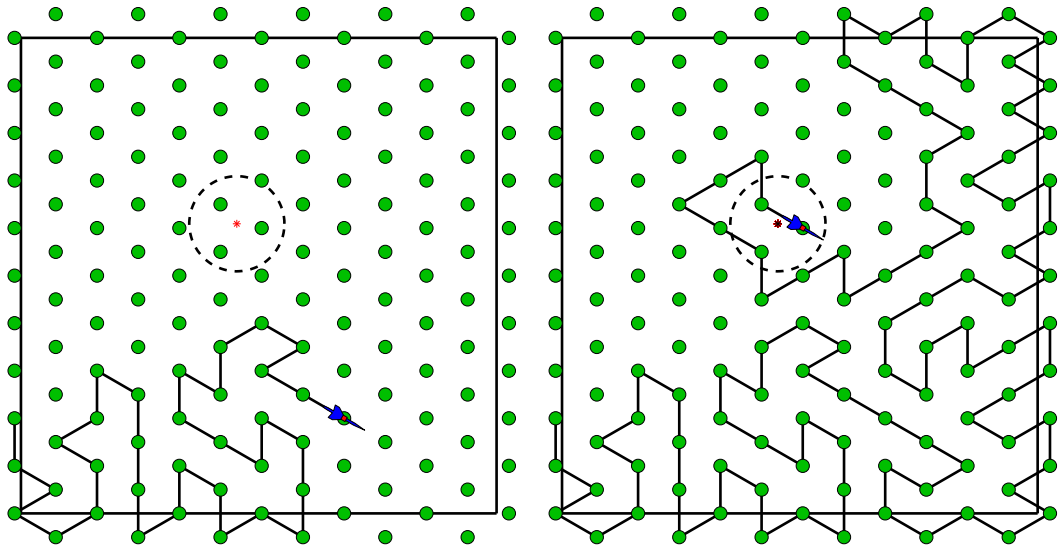
We are now ready to formally state the problem:

Problem 5.2.1 *Define $\overline{W}$ as the steady state expected wait time for the system.*

$$\overline{W} := \limsup_{k \to +\infty} \mathbb{E}\left[W_k(u)\right] \tag{5.5}$$

The goal is to find a stabilizing policy $u \in \mathcal{U}$ for the service vehicle to minimize the steady state expected wait time for the system.

$$\inf_{u \in \mathcal{U}} \overline{W} \tag{5.6}$$



(a) $t_{arrives}$. (b) $t_{localized}$.

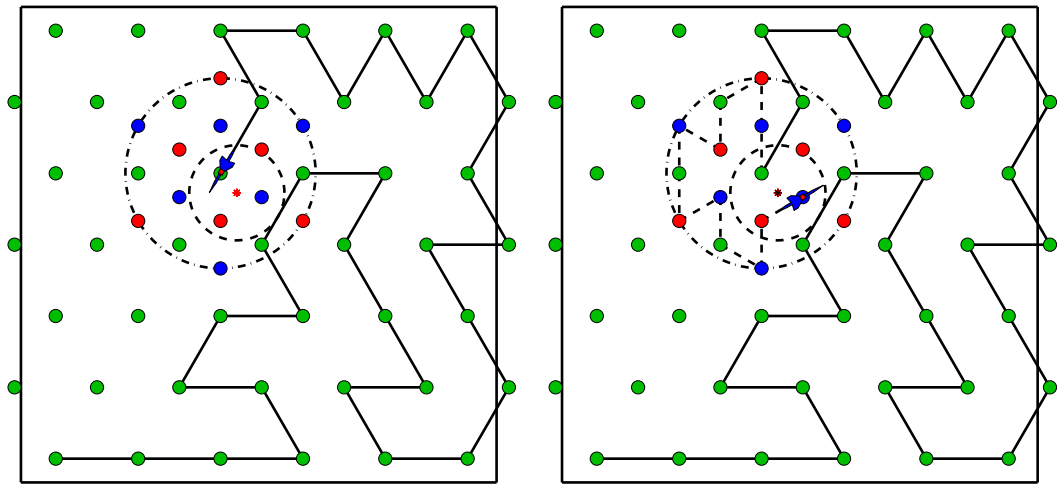
Figure 5.5: The instant an event arrives into the system as in (a), it is time-stamped as $t_{arrives}$, and the instant the service vehicle visits the third sensor to have detected the event as in (b), it is time-stamped as $t_{localized}$.

5.3 Policy Description and Analysis

We now present a pair of novel policies for the service vehicle. The first, called the Search Based (SB) policy, attempts to search over the set of one-cover sensors $\mathcal{S}_{one}$; upon arriving at a sensor that has detected one or more events, it performs a local search over neighboring sensors until all events are localized. See Fig. 5.6 for an illustrative example of the SB policy. For each sensor s_i in the set $\mathcal{S}_{one}$, there exists a set of neighboring sensors within $2r_s$ of s_i, and we will call this set $\mathcal{S}^i_{local}$. An example local ETSP tour over $\mathcal{S}^i_{local}$ can be found in Fig. 5.7. Due to the sensors being placed on a uniform triangular grid, the length of this tour is equal to the product of the cardinality of the set $\mathcal{S}^i_{local}$ and the spacing of the grid r_s,

$$T^i_{local} = |\mathcal{S}^i_{local}| r_s. \tag{5.7}$$

When this local search tour is completed, the service vehicle resumes the original search over the one-cover sensors. It is intuitive to expect that this policy turns out to be effective in the low traffic regime when the occurrence rate of events λ is very small. The second policy, that we will call the Traveling Salesman Based (TSB) policy, attempts to quickly search over all sensors and localizes events when possible. Intuitively, this policy seems to be effective in the high traffic regime when λ is very large. This implies the existence of a switching point



(a) Service vehicle arrives at detecting sensor. (b) Service vehicle arrives at the third detecting sensor, thus localizing the event.

Figure 5.6: The service vehicle executing the Search Based policy. In (a) the service vehicle is searching over the set $\mathcal{S}_{one}$ (greens nodes). When the service vehicle reaches a sensor that has detected an event, it adapts its route to include all sensors within a $2r_s$ radius of the detecting sensor. This radius is indicated by the larger circle. In (b) the service vehicle is performing the local tour and has just visited the third sensor detecting the event, thus completing the localization.

which we will call $\lambda_{critical}$, above which one would switch from the SB policy to the TSB policy. The SB policy is summarized in Algorithm 2.

Algorithm 2 The Search Based Policy SB($\mathcal{S}$)

Require: A Euclidean TSP solver ETSP(S_{one}).

1: Perform ETSP tour over $\mathcal{S}_{one}$.
2: **if** at least one event has been detected by the current sensor in the tour **then**
3: Perform local ETSP tour over all sensors within $2r_s$ of detecting one-cover sensor.
4: Resume ETSP tour over $\mathcal{S}_{one}$ upon completion of local TSP tour.
5: **end if**
6: Repeat.

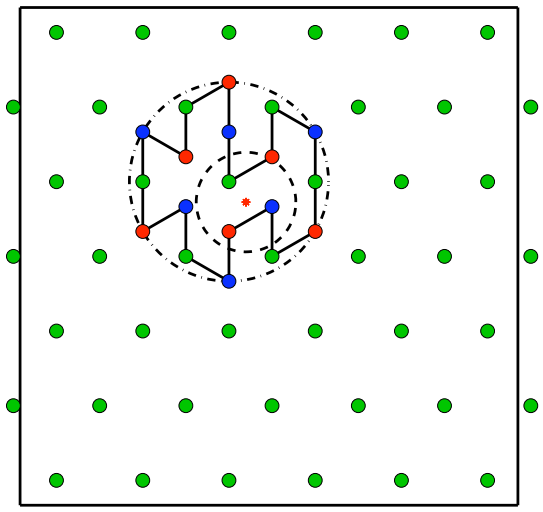


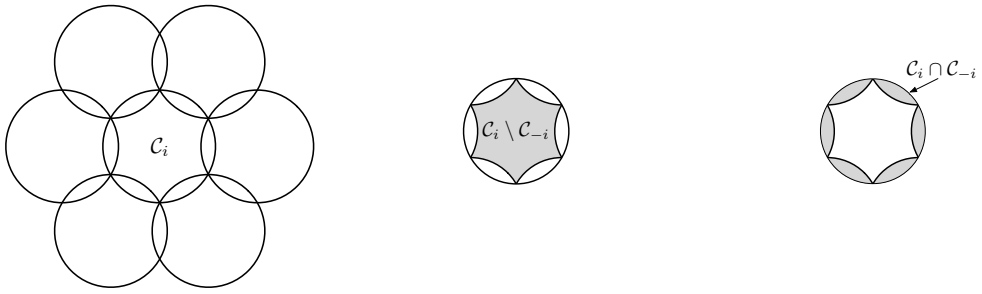
Figure 5.7: The local tour over the set $\mathcal{S}^i_{local}$ of sensors within $2r_s$ of the detecting one-cover sensor s_i .

It is difficult to analyze the performance of Algorithm 2 directly. Here we seek both lower and upper bounds on the steady state expected wait time under the Search Based policy described in Algorithm 2 as functions of the event occurrence rate λ.

Theorem 5.3.1 (Lower Bound for SB Policy) *Suppose that:*

1. *Events occur uniformly random in space and Poisson in time with rate* $\lambda > 0$, *and*

2. *The one-cover sensors are located on a uniformly spaced triangular grid with spacing* $3\sqrt{r_s}$.



(a) $\mathcal{C}_i$ (b) $\mathcal{C}_i \setminus \mathcal{C}_{-i}$ (c) $\mathcal{C}_i \cap \mathcal{C}_{-i}$

Figure 5.8: In (a) the detection region $\mathcal{C}_i$ of sensor s_i can be decomposed into two regions. In (b) the region $\mathcal{C}_i \setminus \mathcal{C}_{-i}$, that is only covered by a single one-cover sensor. In (c) the region $\mathcal{C}_i \cap \mathcal{C}_{-i}$, that is covered by two one-cover sensors.

Then, the steady state expected wait time under the Search Based policy described in Algorithm 2 is lower bounded by,

$$\begin{aligned}\overline{W} \geq & \alpha \left(\frac{T_{int}(m_{int}-1)}{2} + \frac{T_{local} m_{int}(m_{int}-1)}{2} \right. \\ & \left. - \frac{T_{local}}{1-e^{-\lambda\alpha T_{int}/m_{int}}} \left(m_{int} - \frac{1-e^{-\lambda\alpha T_{int}}}{1-e^{-\lambda\alpha T_{int}/m_{int}}} \right) \right) \end{aligned} \tag{5.8}$$

where m_{int} is the number of interior one-cover sensors, T_{int} is the length of the Euclidean Traveling Salesman tour through the interior one-cover sensors $\mathcal{S}_{int}$, and T_{local} is the length of the local tour corresponding to any interior one-cover sensor s_i as in (5.7)*.*

Proof: [Proof of Theorem 5.3.1] Consider a tagged event e_k. There are four disjoint possibilities:

1. e_k is in a region of the type $\mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}$, a region in the interior of the environment and covered by a single one-cover sensor as in Fig. 5.8(b).

2. e_k is in a region of the type $\mathcal{C}_i^{int} \cap \mathcal{C}_{-i}^{one}$, a region in the interior of the environment and covered by two one-cover sensors as in Fig. 5.8(c).

3. e_k is in a region of the type $\mathcal{C}_i^{bdry} \setminus \mathcal{C}_{-i}^{one}$, a region on the boundary of the environment and covered by a single one-cover sensor as in Fig. 5.8(b).

4. e_k is in a region of the type $\mathcal{C}_i^{bdry} \cap \mathcal{C}_{-i}^{one}$, a region on the boundary of the environment and covered by two one-cover sensors as in Fig. 5.8(c).

Let us call these events as E_1, E_2, E_3, and E_4, respectively. Therefore, we can write the expected wait time for detection as,

$$\mathbb{E}[W_k] = \sum_{i=1}^{m_{int}} \mathbb{E}[W_k \,|\, e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}] \mathbb{P}(e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}) \tag{5.9}$$

$$+ \sum_{i=1}^{m_{int}} \mathbb{E}[W_k \,|\, e_k \in \mathcal{C}_i^{int} \cap \mathcal{C}_{-i}^{one}] \mathbb{P}(e_k \in \mathcal{C}_i^{int} \cap \mathcal{C}_{-i}^{one}) \tag{5.10}$$

$$+ \sum_{i=1}^{m_{bdry}} \mathbb{E}[W_k \,|\, e_k \in \mathcal{C}_i^{bdry} \setminus \mathcal{C}_{-i}^{one}] \mathbb{P}(e_k \in \mathcal{C}_i^{bdry} \setminus \mathcal{C}_{-i}^{one}) \tag{5.11}$$

$$+ \sum_{i=1}^{m_{bdry}} \mathbb{E}[W_k \,|\, e_k \in \mathcal{C}_i^{bdry} \cap \mathcal{C}_{-i}^{one}] \mathbb{P}(e_k \in \mathcal{C}_i^{bdry} \cap \mathcal{C}_{-i}^{one}) \tag{5.12}$$

$$\geq \sum_{i=1}^{m_{int}} \mathbb{E}[W_k \,|\, e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}] \mathbb{P}(e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}) \tag{5.13}$$

Due to the one-cover sensors being located on a uniform triangular grid, the area of $\mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}$ is the same for all elements of $\mathcal{C}^{int}$. This area of this region is

$$\mathcal{A}_{sub} := \left(3\sqrt{3} - \pi\right) r_s^2. \tag{5.14}$$

We define the ratio of this area to the total area of $\mathcal{E}$ as,

$$\alpha := \mathcal{A}_{sub}/\mathcal{A}. \tag{5.15}$$

Due to events arriving uniformly randomly in $\mathcal{E}$, the probability of event e_k arriving in $\mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}$ can be written as,

$$\mathbb{P}(e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}) = \alpha \qquad \forall\ i \in \{1, 2, \ldots, m_{int}\}. \tag{5.16}$$

Considering only events of type E_1, we present an alternative policy that will result in a lower expected steady state wait time. First, take the original one tour (through all one-cover sensors) and identify the interior nodes along the tour. For every consecutive pair of interior nodes along the tour, if there is any boundary node on the piece of the tour between them, remove all boundary node(s), and connect those two interior nodes directly. Repeat until there is no boundary node left on the tour. This results in a tour that visits only sensors

in the set $\mathcal{S}_{int}$. Now, clearly the length of the remaining tour is less than the original one tour.

We are left to evaluate the lower bound on this new interior search policy. Let $\mathcal{C}_i^{int}$ denote the region covered by the i^{th} interior one-cover sensor. Consider any 'tagged' event e_k which occurs when the vehicle is anywhere on the path between the m_{int}^{th} interior one-cover sensor and the first interior one-cover sensor. We call this instant t_0, and let t_i denote the time instant at which the vehicle reaches s_i. Clearly,

$$(i-1)\frac{T_{\text{int}}}{m_{int}} \leq t_i - t_0, \tag{5.17}$$

because in the best case, the vehicle would not perform any local TSP tours at any intermediate sensor locations. For any event A, let I_A denote the indicator function of A. Then, conditioned on the event that $e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}$, the wait time W_k until localization of e_k satisfies

$$W_k \geq (i-1)\frac{T_{int}}{m_{int}} + T_{local}\sum_{j=1}^{i-1} I_{\mathcal{C}_j^{int}\setminus\mathcal{C}_{-j}^{one}\text{is non-empty in}[to,tj]}. \tag{5.18}$$

Taking the expected value,

$$\Rightarrow \mathbb{E}\left[W_k \,|\, e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}\right] \geq (i-1)\frac{T_{int}}{m_{int}} + T_{local}\sum_{j=1}^{i-1}\mathbb{P}(\mathcal{C}_j^{int} \setminus \mathcal{C}_{-j}^{one} \text{is non-empty in} [t_0, t_j]) \tag{5.19}$$

$$\Rightarrow \mathbb{E}\left[W_k \,|\, e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}\right] \geq (i-1)\frac{T_{int}}{m_{int}} + T_{local}\sum_{j=1}^{i-1}(1 - e^{-\lambda\alpha(t_j - t_0)}), \tag{5.20}$$

where we used Poisson Assumption,

$$\Rightarrow \mathbb{E}\left[W_k \,|\, e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}\right] \geq (i-1)\frac{T_{int}}{m_{int}} + T_{local}\sum_{j=1}^{i-1}(1 - e^{-\lambda\alpha(j-1)T_{int}/m_{int}}), \tag{5.21}$$

where we used LHS of (5.17).

Returning to the inequality in (5.13),

$$\mathbb{E}\left[W_k\right] \geq \sum_{i=1}^{m_{int}} \mathbb{E}[W_k \,|\, e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}]\mathbb{P}(e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}) \tag{5.22}$$

$$\geq \alpha \sum_{i=1}^{m_{int}} \mathbb{E}\left[W_k \,|\, e_k \in \mathcal{C}_i^{int} \setminus \mathcal{C}_{-i}^{one}\right] \tag{5.23}$$

$$\geq \alpha \left(\sum_{i=1}^{m_{int}} (i-1)\frac{T_{int}}{m_{int}} + \sum_{i=1}^{m_{int}} \sum_{j=1}^{i-1} T_{local}(1 - e^{-\lambda\alpha(j-1)T_{int}/m_{int}}) \right) \tag{5.24}$$

$$\geq \alpha \left(\sum_{i=1}^{m_{int}} (i-1)\frac{T_{int}}{m_{int}} + \sum_{i=1}^{m_{int}} \sum_{j=1}^{i-1} T_{local} - \sum_{i=1}^{m_{int}} \sum_{j=1}^{i-1} T_{local} e^{-\lambda\alpha(j-1)T_{int}/m_{int}} \right) \tag{5.25}$$

$$\geq \alpha \left(\frac{T_{int}(m_{int}-1)}{2} + \frac{T_{local} m_{int}(m_{int}-1)}{2} - T_{local} \sum_{i=1}^{m_{int}} \sum_{j=1}^{i-1} e^{-\lambda\alpha(j-1)T_{int}/m_{int}} \right) \tag{5.26}$$

$$\geq \alpha \left(\frac{T_{int}(m_{int}-1)}{2} + \frac{T_{local} m_{int}(m_{int}-1)}{2} - \frac{T_{local}}{1 - e^{-\lambda\alpha T_{int}/m_{int}}} \left(m_{int} - \frac{1 - e^{-\lambda\alpha T_{int}}}{1 - e^{-\lambda\alpha T_{int}/m_{int}}} \right) \right). \tag{5.27}$$

Upon noting that (5.27) holds uniformly for all events, we arrive at (5.8).

■

Next, we establish an upper bound on the steady state expected wait time under the Search Based policy.

Theorem 5.3.2 (Upper Bound for SB Policy) *Suppose that:*

1. *Events occurrence is uniformly random in space and Poisson in time with rate* $\lambda > 0$, *and*

2. *The one-cover sensors are located on a uniformly spaced triangular grid with spacing* $3\sqrt{r_s}$.

Then, the steady state expected wait time under the Search Based policy described in Algorithm 2 is upper bounded by,

$$\overline{W} \leq \frac{1}{2}\left(\left(\frac{T_{one}}{m} + T_{lmax}\right)(m+1) - T_{l_{max}}(m-1)\, e^{-\lambda(T_{one}+mT_{l_{max}})/m}\right). \quad (5.28)$$

Here m *is the number of one-cover sensors,* T_{one} *is the length of the Euclidean Traveling Salesman tour through the one-cover sensors* $\mathcal{S}_{one}$, *and* $T_{l_{max}}$ *is defined as,*

$$T_{l_{max}} := \max_{i \in \{1,2,\ldots,m\}} T^i_{local}, \quad (5.29)$$

where T^i_{local} *is the length of the local tour corresponding to sensor* s_i *as in* (5.7).

Proof: [Proof of Theorem 5.3.2] Let $\mathcal{C}_i$ denote the region covered by the i^{th} one-cover sensor. Consider any 'tagged' event e_k which occurs when the vehicle is located anywhere on the path between the m^{th} one-cover sensor and the first one-cover sensor. We call this instant t_0 and let t_i denote the time instant at

which the vehicle reaches s_i. Clearly,

$$t_i - t_0 \leq T_{\text{one}} + mT_{l_{max}}, \tag{5.30}$$

because in the worst case, the vehicle would have to perform a local TSP at every sensor location.

For any event A, let I_A denote the indicator function of A. Then, conditioned on the event that $e_k \in \mathcal{C}_i$, the wait time W_k until localization of e_k satisfies

$$W_k \leq i\frac{T_{one}}{m} + T_{l_{max}}\left(1 + \sum_{j=1}^{i-1} I_{\mathcal{C}_j \text{is non-empty in}[to,tj]}\right) \tag{5.31}$$

$$\mathbb{E}\left[W_k \,|\, e_k \in \mathcal{C}_i\right] \leq i\frac{T_{one}}{m} + T_{l_{max}}\left(1 + \sum_{j=1}^{i-1} \mathbb{P}(\mathcal{C}_j \text{is non-empty in}[to,tj])\right) \tag{5.32}$$

$$\mathbb{E}\left[W_k \,|\, e_k \in \mathcal{C}_i\right] \leq i\frac{T_{one}}{m} + T_{l_{max}}\left(1 + \sum_{j=1}^{i-1} \left(1 - e^{-\lambda(t_j - t_0)/m}\right)\right) \tag{5.33}$$

$$\mathbb{E}\left[W_k \,|\, e_k \in \mathcal{C}_i\right] \leq i\frac{T_{one}}{m} + T_{l_{max}}\left(1 + \sum_{j=1}^{i-1} \left(1 - e^{-\lambda(T_{one} + mT_{l_{max}})/m}\right)\right). \tag{5.34}$$

Since

$$\mathbb{E}\left[W_k\right] \leq \frac{1}{m}\sum_{i=1}^{m} \mathbb{E}\left[W_k \,|\, e_k \in \mathcal{C}_i\right], \tag{5.35}$$

we obtain,

$$\mathbb{E}\left[W_k\right] \leq \frac{1}{2}\left(\left(\frac{T_{one}}{m} + T_{lmax}\right)(m+1) - T_{l_{max}}(m-1)\, e^{-\lambda(T_{one}+mT_{l_{max}})/m}\right). \quad (5.36)$$

We arrive at the expression in (5.28) by noting that equation (5.36) holds uniformly for all events. ■

Next, we formalize the TSB policy, a summary can be found in Algorithm 3.

Algorithm 3 The Traveling Salesman Based Policy TSB($\mathcal{S}$)

Require: A Euclidean TSP solver ETSP($\mathcal{S}$).

1: Perform ETSP tour over $\mathcal{S}$ localizing each event after three detections have been aquired.
2: Repeat.

Theorem 5.3.3 (Upper Bound for TSB Policy) *Suppose that:*

1. *Events occurrence is uniformly random in space and Poisson in time with rate $\lambda > 0$, and*

2. *The sensors are located on a uniformly spaced triangular grid with spacing r_s.*

3. *$T_{three} := ETSP(\mathcal{S})$ is the cost of the Traveling Salesman Problem (TSP) Tour on n sensors.*

Then, the steady state expected wait time under the Traveling Salesman Based policy described in Algorithm 3 is upper bounded by,

$$\overline{W} \leq \frac{T_{three}}{2}. \tag{5.37}$$

Proof: [Proof of Theorem 5.3.3] Consider any 'tagged' event e_k and denote the set of sensors which detected this event as $\mathcal{S}^k_{detect}$. Due to Algorithm 3 following a fixed policy, the time between subsequent visits of all sensors in $\mathcal{S}^k_{detect}$ is upper bounded by T_{three}. Due to the properties of a Poisson process, the time of occurrence t_0 for event e_k is distributed uniformly in time interval between the i^{th} and $(i+1)^{th}$ visit of sensors in S^k_{detect},

$$t_0 \in ((i)T_{three}, (i+1)T_{three}] \quad i \in \mathbb{Z}^*. \tag{5.38}$$

Thus,

$$\mathbb{E}[t_0] = \frac{(2i+1)T_{three}}{2}, \tag{5.39}$$

$$\mathbb{E}[t_k] \leq (i+1)T_{three}. \tag{5.40}$$

From (5.2), we have,

$$\mathbb{E}[W_k] \leq \frac{T_{three}}{2}. \tag{5.41}$$

Upon noting that (5.41) holds uniformly for all events we arrive at (5.37). ■

Given knowledge of the event occurrence rate λ, the planner for the service vehicle would ideally switch between Algorithm 2 and Algorithm 3 depending on λ. Let $\lambda_{critical}$ denote this switching point. When $\lambda < \lambda_{critical}$, the planner should pick Algorithm 2 (SB), and when $\lambda \geq \lambda_{critical}$ the planner should pick Algorithm 3 (TSB). Unfortunately, computing $\lambda_{critical}$ would require analytical solutions for $\overline{W}$ as a function of λ for each policy. We propose instead to use the bounds from Theorems 5.3.1 and 5.3.3 to compute the appropriate switching point. This point, which we will call λ_{switch}, is precisely the point where the lower bound for SB (5.8) crosses the upper bound for TSB (5.37).

5.4 Numerical Example

To illustrate the system performance of the SB and TSB policies in comparison to the theoretical bounds, the expected steady state wait time $\overline{W}$ was calculated for each policy for a range of occurrence rates λ. For this example the total number of sensors $|\mathcal{S}| = 149$, and the sensing radius $r_s = 1$. The plot in Fig. 5.9 shows $\overline{W}$ versus λ for each policy along with all related bounds. The upper and lower bounds for Search Based policy are shown in red lines and the possible performance is shown by the green shaded area between the curves.

The actual performance of the SB policy is represented by the blue line in the the shaded region. The upper bound for the static Traveling Salesman Based policy is shown as the constant black line, and the actual performance of the TSB policy is plotted as the blue line and indicates that the upper bound is relatively tight. Finally, the point where $\overline{W}$ under the SB policy crosses that of the TSB policy is labeled as $\lambda_{critical}$, and the point where the lower bound for the SB policy crosses the upper bound for the TSB policy is labeled as λ_{switch}. It should be noted that while the lower bound on the SB policy is relatively tight in the low traffic regime, it is loose in the medium to high traffic regime, and this results in a gap between $\lambda_{critical}$, and λ_{switch}.

5.5 Summary

In this chapter we have studied the problem of dynamic vehicle routing for data mules which service sparsely deployed wireless sensor networks. We have shown that in the low traffic regime ($\lambda \to 0^+$), significant improvements in $\overline{W}$ are achieved over a static policy when the service vehicle adaptively adjusts its route based on data it has collected. We have provided both upper and lower bounds on the performance of this search based policy (SB). A static policy that visits all the sensors in the network as quickly as possible was proposed for the

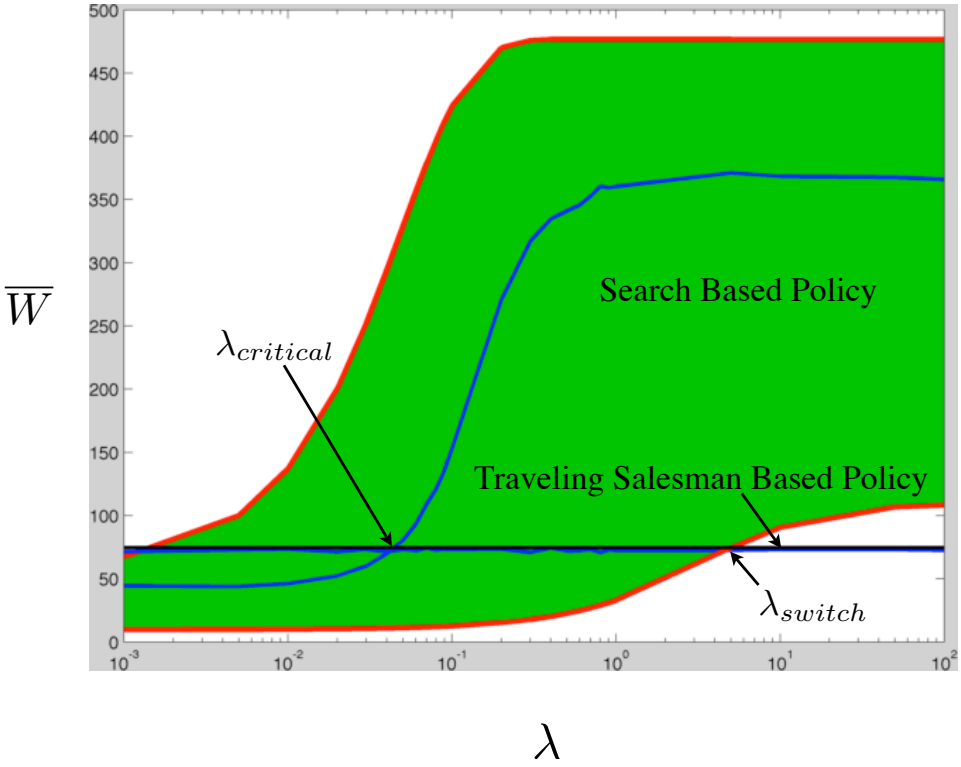


Figure 5.9: A numerical example of the steady state expected wait time $\overline{W}$ for both Algorithms 2 and 3 along with bounds from Theorems 5.3.1, 5.3.2, and 5.3.3.

high traffic case as $\lambda \to +\infty$. An upper bound on the performance $\overline{W}$ for the traveling salesman based policy (TSB) was established. With a lower bound on $\overline{W}$ for SB and an upper bound for TSB, the planner can then switch from the SB to the TSB when the arrival rate is above a critical value $\lambda \geq \lambda_{switch}$. This point is precisely the point where the lower bound for SB and the upper bound for TSB intersect.

Chapter 6

Field Demonstrations

6.1 Introduction

The algorithms from Chapter 3 were demonstrated at the Center for Interdisciplinary Remotely-Piloted Aircraft Studies (CIRPAS) facility at Camp Roberts using a Fury UAV from Aeromech Engineering, Inc. and a heterogeneous collection of sensor nodes consisting of several single microphone sensors, providing time of arrival (TOA) measurements, as well as a microphone array from the US Army Research Lab, providing angle of arrival (AOA) measurements. The UAV route was recalculated after each sensor measurement was processed within a Bayesian framework, and a downward looking camera on the UAV was used to image the source (a Zon Mark IV propane cannon) once successful localization was deemed to have taken place.

We developed a modular system architecture, shown in Fig. 6.1, which enabled a smooth transition from simulation to deployment: algorithms that did not work effectively were replaced quickly, and a flight simulator was used in place of the real UAV during development and testing. Each sensor was fitted with one or more sensitive microphones and processed the live audio stream to achieve two goals: (1) detect an event in the presence of ambient noise and (2) produce a Time-of-Arrival (TOA) or an Angle-of-Arrival (AOA) measurement corresponding to each event. Local signal processing captured "useful" data succinctly: large raw audio files that would have required enormous bandwidths to transmit were reduced to TOA/AOA information that was easily transmitted over relatively slow links. Each sensor forwarded its TOA or AOA information to the UAV when it came within radio range. Instead of employing a fixed route for data collection, routing was coupled with localization to make smart decisions about where the UAV was to go next. Specifically, freshly acquired information was combined with prior data in a Bayesian fashion in order to decide which sensor to visit next to locate the source as quickly as possible, or if the uncertainty regarding the source location was small enough, to go to the estimated source location. A significant reduction in time to localize came in scenarios in which data from all sensors was not required to localize the source up

to the desired accuracy. From our field deployment, we found that this situation was the norm rather than the exception.

Related Work: Over the past decade, sensor networks have been *deployed* to address a wide variety of problems such as habitat monitoring [60], source localization [2, 96], sniper detection [82], classification and tracking [6], indoor location services [72], structural monitoring [100], and volcanic studies [97]. An acoustic sensor network was used in [2,96] to localize marmots and woodpeckers from their calls. This deployment, which used AOA sensors (in contrast to our hybrid AOA-TOA system) was on a much smaller scale than ours, with maximal sensor separations on the order of 10m and 150m respectively. Furthermore, unlike our sparse deployment, the arrays were close enough to exchange recorded waveforms and estimate an Approximate Maximum Likelihood (AML) source direction in multipath environments. The large scale deployment in [6] also investigates a heterogeneous sensor network to detect, classify, localize, and track targets. However, it employs magnetic and radar sensors (in contrast to acoustic sensors), and the deployment is dense enough for the sensors to form a connected network. The deployment closest in spirit to ours is the sniper detection network in [82], where a network of TOA sensors detect and localize a sniper based on the muzzle blast and shock wave. However, the nodes in [82] are deployed densely on a smaller spatial scale, so that they can form a connected network to exchange

data, and they use TOA sensors alone, unlike our hybrid AOA-TOA approach. A hybrid AOA-TOA approach has been considered in [94], but again on a small scale.

Organization: In Section 6.2, we present each component in detail, focusing on the hardware. Results from field tests are presented in Section 6.3.

6.2 System Components for Field Demonstration

In this section, we describe the system components, focusing primarily on the hardware choices made and the rationale behind these choices. We begin with the acoustic source and then describe the components that go into each sensor: GPS, laptop, microphone, microphone array and the radio used for communication. We finally describe the UAV used in the field deployment and the base station.

6.2.1 Acoustic Source

We used a Zon Mark IV propane cannon to create acoustic events with characteristics similar to live artillery. The sound level at the muzzle of the cannon is approximately 120 dB. A TOA sensor node (to be described shortly)

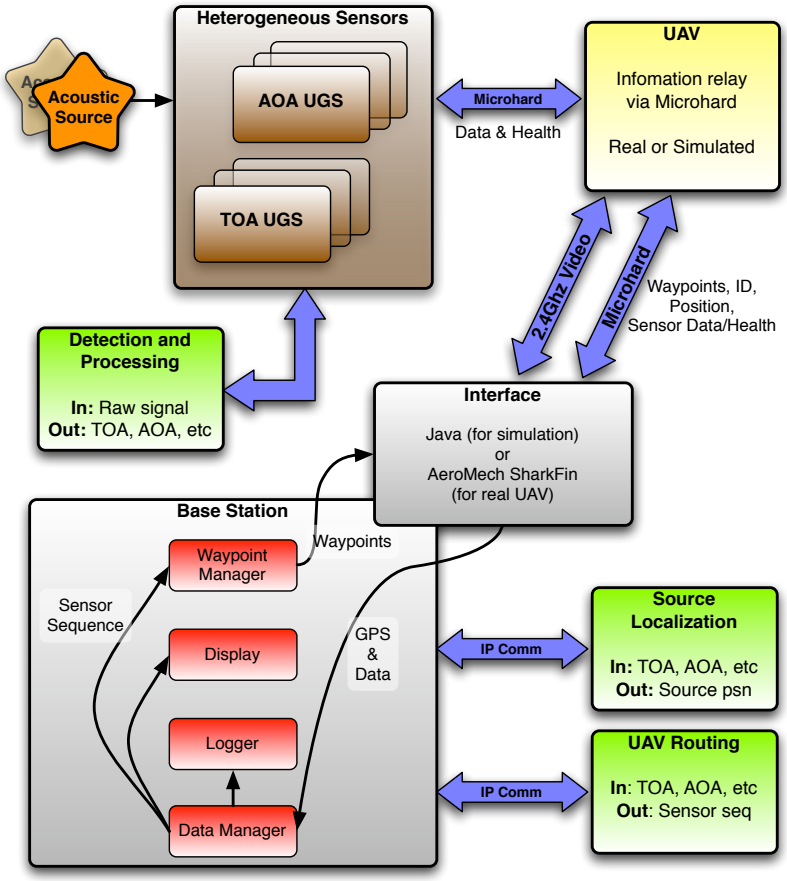


Figure 6.1: The modular software architecture used for simulation and field testing.

was placed close to the cannon to record the true event location and time. Fig. 6.2 shows the propane cannon mounted in the bed of a pickup truck.

6.2.2 Time Synchronization and Localization

Localizing an acoustic source based on TOAs requires the sensors to have a common notion of space and be accurately synchronized in time. Given the

Figure 6.2: The propane cannon used to create acoustic disturbances.

rapidly dropping cost of GPS receivers, both synchronization and sensor localization can be achieved economically by equipping each sensor with a GPS unit. However, the choice of the GPS unit is critical, especially for accurate timing. Furthermore, the GPS receivers that provide tight timing synchronization are not "plug-and-play" units. Consequently, in addition to a good choice of the GPS unit, hardware modifications and software choices (e.g. FreeBSD operating system) are needed to achieve μs level timing accuracy across nodes.

Time Synchronization: To achieve μs level timing synchronization, it is critical to choose a GPS unit that has a Pulse Per Second (PPS) output. Consequently, we chose the Garmin 18-LVC unit. The PPS output is a logical signal which indicates the start of a second very precisely. This is done by making the rising edge of the logical signal coincide with the second, whose value is reported in a separate string. When used in conjunction with the Network Time

Protocol (NTP) to perform filtering and set the system clock, the PPS signal helps us achieve μs level timing synchronization. Note that NTP for our purpose is simply an application *running separately at each node*, since nodes are unable to communicate directly. More details on achieving timing synchronization on the order of μs with GPS units can be found in [98].

GPS Based Localization: We convert the latitude-longitude output of the GPS unit to a local two dimensional Cartesian coordinate system that is then used by the localization and routing algorithms. We use the geodetic-to-East North Up (ENU) conversion rules described in [99] for this purpose. The location of each sensor is then filtered with a moving average filter to account for GPS measurement noise.

6.2.3 Time of Arrival Sensor

To expedite the field demonstration for proof of concept, we built the TOA sensor with off-the-shelf components. The resulting sensors are extremely heavy-weight and not energy efficient, but the sensors could easily be downsized to a small form factor using standard mote hardware.

Hardware: We interfaced a condenser microphone to a laptop to acquire and process the raw audio signals. *Microphone:* We chose the multipattern Samson C03U microphone for its sensitivity and flat frequency response over

a wide range of frequencies (50 Hz - 5000 Hz). A foam windscreen prevented unwanted noise. *Laptop and Software:* We chose a Dell Latitude E5500 laptop running FreeBSD and custom Java applications to process and store the raw audio signals. *Power Source:* The laptop was powered for an entire day by a marine deep cycle battery through an inverter. *Radio:* The radio was an n920 unit from Microhard Systems, Inc., described in Section 6.2.5. We had six such TOA sensors, with one of them positioned very close to the propane cannon to obtain ground truth. Therefore, we had $N_{TOA} = 5$ TOA sensors to make measurements. One of these sensors is shown in Fig. 6.3(a).

6.2.4 Angle of Arrival Sensor

The AOA sensor is an Acoustic Transient Detector System (ATDS) provided by the Army Research Lab (ARL). It consists of four microphones located at the corners of a triangular pyramid as shown in Fig. 6.3(b). A compass was used to orient one of the "arms" of the array along magnetic north. The Earth's magnetic declination needs to be taken into account while converting the AOA back to the "global" East North Up coordinate system; on the days of our tests, the magnetic north was oriented $13°32'$ east of geographic north at Camp Roberts.

(a) Time of Arrival Sensor.

(b) Angle of Arrival Sensor.

Figure 6.3: (a) The time-of-arrival sensor consisting of a Dell laptop, Samson microphone, Garmin GPS, Microhard radio, battery and an inverter. (b) The angle-of-arrival sensor provided by the Army Research Lab.

6.2.5 Communication

Communication Topology & Protocols: Each sensor transmits data such as event TOAs/AOAs and sensor location as well as sensor status information like battery life and temperature over a radio link to the UAV. Due to UAV payload and control restrictions, we were unable to perform computations onboard the UAV. Instead, we performed the signal processing envisioned for

the UAV using a base station on the ground. Thus, Microhard radios were used to establish a network for communication between the sensors deployed in the field, the airborne UAV and a base station. The radio onboard the UAV acted as a relay, ferrying messages between the sensors and the base station. Let us now look at the typical flow of information in this network.

The sensors first transmit information to the UAV, which relays this information to the base station. The base station provides an acknowledgment upon receiving the data. The radio onboard the UAV forwards the acknowledgment to the sensor, concluding the information flow for the event under consideration. In case the sensor does not receive an acknowledgment, the data is retransmitted a maximum of five times.

Emulation of Short-Range Radio Links: Our system emulates short-range radio links typical of long-life sensor nodes, low-profile antennas, rough terrain and low-flying UAVs. However, the emulation was simplified by taking advantage of high-power radios, high-altitude UAV, and relatively benign terrain of operation for our demonstration, which meant that most of the sensors had radio connectivity with the UAV. Thus, in our experiments the sensor can transmit TOAs/AOAs to the UAV right after detection, which simplifies the message exchange protocol. This connectivity to the base station has the additional benefit of enabling the debugging of the system as physical access to the sensors is

a very laborious process. In order to emulate short radio ranges, we "unlock" sensor data only when the UAV comes within an "effective" communication footprint of the sensor. This footprint is roughly a circular region within 200 m of the sensor, but we account for NLoS effects using a terrain map to further reduce the footprints, see Fig. 6.4(a). To facilitate simultaneous transmissions, the radios were set in a Point-to-Multipoint TDMA configuration, and built in multiple access schemes resolved potential conflicts. The Microhard Nano n920 radios that we employed operate in the 902-928 MHz band and employ frequency hopped spread spectrum for communication. While these "high power" radios have an advertised range of 100 km in LoS scenarios, in non-line of sight (NLoS) settings, which is typical of field deployments, we found that the communication radius of these radios is less than 500 m. And of course, as described above, we restrict the effective communication range to 200 m or less.

6.2.6 Unmanned Aerial Vehicle

For the flight test, we used a Fury UAV (shown in Fig. 6.4(b)) from AeroMech Engineering, Inc. This is a high-performance UAV capable of 18 hours of sustained flight and speeds up to 40 m/s. The payload on the UAV consists of a Microhard radio and a downward looking (nadir) camera. The camera is used to image the acoustic source once it is located. While our eventual plan is to

process the TOAs/AOAs onboard the UAV, we did not have control over the payload and could not locate the base station functionality on the UAV.

The interface to the Fury UAV is achieved by communicating with SharkFin, the UAV's ground control software and visualization package. For Monte Carlo simulation studies, we interfaced the base station to FlightGear, an open source flight simulator, which provides realistic vehicle dynamics and also models the effects of wind.

6.2.7 Base Station

The base station performs the critical tasks of source localization and UAV routing from the TOA/AOA measurements that are unlocked by the UAV. In addition, it provides a number of services such as interfacing to the UAV (real or simulated), data logging, display, waypoint management, and debugging output such as sensor temperature, remaining battery life, GPS positioning and NTP timing statuses.

AeroMech modified SharkFin to enable communication of UAV routes, fly-over commands, and status communications via a TCP connection. However, range safety and liability concerns prevented us from routing the UAV in a completely automated fashion. A human operator observed the routes (i.e. sequence of sensor waypoints to visit) output by our UAV routing protocol and manu-

ally specified a "safe" flight path for the UAV. The human operator essentially always adhered to the output of our UAV routing protocol.

6.3 Experiments and Results

We have conducted two types of demonstrations/experiments. The first demonstration consisted of a pair of flight tests where we deployed 6 sensors over a 1 km^2 region and localized acoustic sources within this area. This emulates a portion of a large sensor network responsible for localizing an acoustic source and illustrates the efficacy of the overall system. In the second set of tests, we fired a propane cannon repeatedly to characterize the statistical performance of the detection algorithms in a realistic environment.

The flight and statistical tests used the facilities of the Center for Interdisciplinary Remotely-Piloted Aircraft Studies (CIRPAS). McMillan Airfield, Camp Roberts, CA provides CIRPAS dedicated airspace for UAV testing that is remote from populated areas and free of interference from commercial or military air traffic. The McMillan Airfield is located near the southern boundary of Camp Roberts at an elevation of 280 m and is surrounded by lightly wooded rolling hills and open grasslands. Fig. 6.3(b) provides a rough idea of the terrain near the airfield.

Table 6.1: Localization Results

Date	Nov. 3, 2009	
Trial	Source Location Error (m)	# of Sensors Visitied
1	20.3	5
2	4.3	6
3	4.5	4
Date	Nov. 6, 2009	
Trial	Source Location Error (m)	# of Sensors Visitied
1	5.6	4
2	2.0	5
3	1.8	5
4	2.6	4
5	19.6	5
6	10.4	4

6.3.1 Flight Tests

We conducted flight tests on November 3^{rd} and 6^{th}, 2009 at Camp Roberts with five TOA sensors and one AOA sensor. The sensor locations on the two days are shown in Fig. 6.3.1. The distance between the sensors and the propane cannon varied between 96 m and 880 m on November 3^{rd} and 31 m and 603 m on November 6^{th}.

Flight Test Protocol: In its "default" mode, the UAV was commanded to loiter about a sensor in the center of the surveillance region. We chose the "loiter sensor" to be the AOA sensor since it identifies (with high probability) a large portion of the field where the source *cannot* lie; on the other hand, a

single TOA measurement with no prior information is of no immediate use in localizing the source because of the uncertainty about the event time. When an event is detected at the AOA sensor, the TTM algorithm is used to determine the sequence of sensors the UAV should visit. Event TOAs obtained by visiting other sensors are combined with any available measurements to arrive at a better estimate of the source location; if the volume of the uncertainty ellipse drops below the pre-defined threshold V_{th}, the UAV is instructed to abandon its route and fly over the estimated source location. Otherwise, a new route is computed using the TTM routing algorithm until sufficient confidence is obtained on the source location.

The time consumed in locating the source was dominated by the time taken by the human operator to specify the course (see Section 6.2.7) and therefore, does not convey the time savings that would be obtained by a fully automated implementation. We therefore report only on the accuracy of source localization (and the number of sensors visited before the event was localized) from the field experiments. The time savings from our data-driven routing algorithm are quantified using results from a flight simulator, described in Chapter 3.

Results: We conducted nine flight tests spread over two days. The results are shown in Table 6.1. We find that the estimated source location is always within 20 m of the true source and on a few occasions, the error is only a

couple of meters. Upon review, we discovered that the two instances where the source location error exceeded 15 m were due to large errors in the AOA estimates. Subsequent tests showed that such errors in the AOA estimates could be caused by the signal arriving along multiple directions (after reflections off nearby objects). These outlier estimates can be handled in a Bayesian framework by giving a low confidence to measurements from the AOA sensor. However, this was not part of the original measurement model and we could not repeat the hardware experiments with a more appropriate model. Instead, we omitted the erroneous data and found that this reduced the localization error to a few meters. On eight of the nine trials the source is "localized" without needing to visit all six sensors. Here, we deemed the source to be localized when the volume of the 5% uncertainty ellipse fell below a pre-determined threshold of 420m^2 (the threshold was chosen so that a UAV could, with high probability, localize the source with a single fly-over). Note that localization time is our primary routing objective, and that choosing to skip some of the sensors is simply a byproduct.

6.3.2 Statistical Tests

We performed further field tests to address three issues: (a) quantify the performance of the detection algorithm statistically, (b) validate the TOA model (3.1) and (c) obtain statistics of localization accuracy. To this end, we conducted

Table 6.2: Detection Performance

(a) Jan. 28, 2010 Morning

Sensor:	#1	#2	#3	#4
False	24	49	10	6
Missed	2	109	14	4
Correct	382	275	370	380

(b) Jan. 28, 2010 Afternoon

Sensor:	#1	#2	#3	#4
False	64	28	17	52
Missed	4	23	9	2
Correct	284	265	279	286

two sets of tests with TOA sensors on January 28, 2010; the picture in Fig. 6.6 shows the sensor locations in the morning with S_0 providing the ground truth. In the afternoon, the propane cannon and S_0 exchanged their location with S_4. The duration of each test was roughly three hours with the propane cannon being fired twice a minute. We will now describe the results of postprocessing the recorded data which provides answers to the questions raised.

Detection Performance: We characterize the performance of the detection algorithm by the traditional false alarm and missed detection metrics: a cannon shot that was not detected at a sensor is said to be a missed detection, while a positive detection made when there was no shot in reality is called a false alarm. Ground truth was obtained from a sensor placed right next to the propane cannon. To separate the performance of the detection algorithm from timing issues, we deemed a detection to be correct if it fell within ± 0.045 seconds (corresponding to approximately "3σ") of the expected TOA. If there were no detections in this window, then the cannon shot was deemed to be missed. Any

detections outside this window were declared to be false alarms. The results are shown in Table 6.2 and a few points are worth noting: (1) During the morning tests, the cannon shot was successfully detected at *all four* sensors on 270 occasions out of 384 shots. In the afternoon, we detected 257 out of 288 shots at all four sensors. (2) In the morning, Sensor 2 contributes to a large fraction of the misses, primarily because of significant foliage that lies between the sensor and the propane cannon (seen in Fig. 6.6). Surprisingly, a small change in the propane cannon location in the afternoon improves the probability of detection at Sensor 2 dramatically: 92% of the shots are detected at Sensor 2, in spite of the foliage that is still present between the sensor and the cannon. (3) At the other sensors, the performance of the matched filtering style algorithm is very good with more than 95% of the shots being detected.

TOA & Localization Statistics: In the model, the expected TOA at sensor k for an event occurring at T_0 is given by $T_0 + d_k(\boldsymbol{\theta})/\nu$ where $d_k(\boldsymbol{\theta})$ is the distance between the source at $\boldsymbol{\theta}$ and sensor k, and ν is the speed of sound taken to be 340.29 m/s. Furthermore, the variation around the mean is assumed to be Gaussian. To verify this model, we placed a sensor close to the propane cannon to provide the event time T_0 and the distance was estimated from GPS data. We found a significant bias on the order of 20 ms in the measurements. Curiously, the bias also changed sign over the course of the day: the measured TOAs were

larger than the expected TOAs in the morning and became smaller than the expected TOAs as the day went on. This led us to speculate that the change in speed of sound with temperature was the reason behind the bias. In [18], it is shown that speed of sound depends on the temperature $T(^{\circ}\mathrm{C})$ as

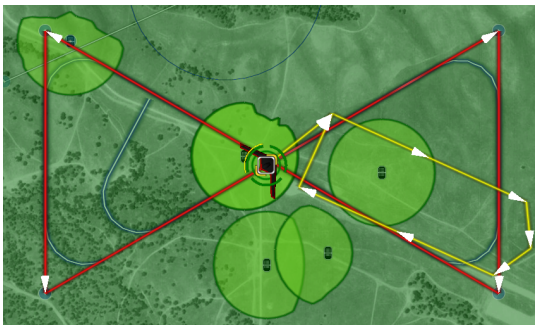
$$\nu(T) = 331.45\sqrt{1 + \frac{T}{273.15}} \text{ m/s.} \tag{6.1}$$

While we did not directly measure the temperature during the test, we obtained hourly temperature measurements from a nearby weather station [88]. The speed of sound estimates corresponding to these temperature readings based on (6.1) can be found in Table 6.3. We see that the speed of sound varied by as much as 6 m/s over the day, leading to TOA errors that grow with distance: at a 1 km range, assuming a nominal value for the speed of sound leads to a TOA error of 50 ms which roughly translates to 17 m of location error.

We recomputed the TOA statistics taking the temperature variations into account and plot the results for the morning and afternoon tests in Figures 6.7(a) and 6.7(b), respectively. The localization error histograms along with the 95% confidence ellipses can be seen in Figures 6.8(a) and 6.8(b) respectively. Localization errors were computed only using shots in which all four sensors detected the event.

For the tests in the morning, the bias is significantly reduced from 20 ms to less than 10 ms when temperature variations are taken into account. For the tests in the afternoon, there is a significant bias in the measurements even after correcting for temperature variations. The measured TOAs are biased from the expected TOAs by as much as 22 ms. The only explanation we can provide for the bias in the measurements is that the registered TOAs are caused by sound waves reflected off objects, rather than the direct path between the source and the sensor. However, we cannot prove this assertion rigorously.

The experiments illustrate that temperature variations and the propagation environment can have a significant effect on the bias of the TOA measurements. This bias causes slightly more than 5% of the localization errors to fall outside the 95% confidence ellipse. In spite of these inaccuracies in modeling the sound propagation, the localization is robust in that the maximum error during the entire day of testing was less than 14 m.



(a) Sharkfin interface with sensor footprints.

(b) AeroMech Fury.

Figure 6.4: This image shows a portion of the SharkFin interface (a) and the Fury UAV (b). The bright green blobs are communication regions for their respective sensors and the red hourglass is a figure-eight UAV flight pattern, used to image the source once localized.

(a) Configuration for Nov. 3^{rd}, 2009.

(b) Angle of Arrival Sensor.

Figure 6.5: In both cases the position of the acoustic source is indicated by the megaphone symbol, and the positions of the sensor nodes are indicated by the microphone symbols.

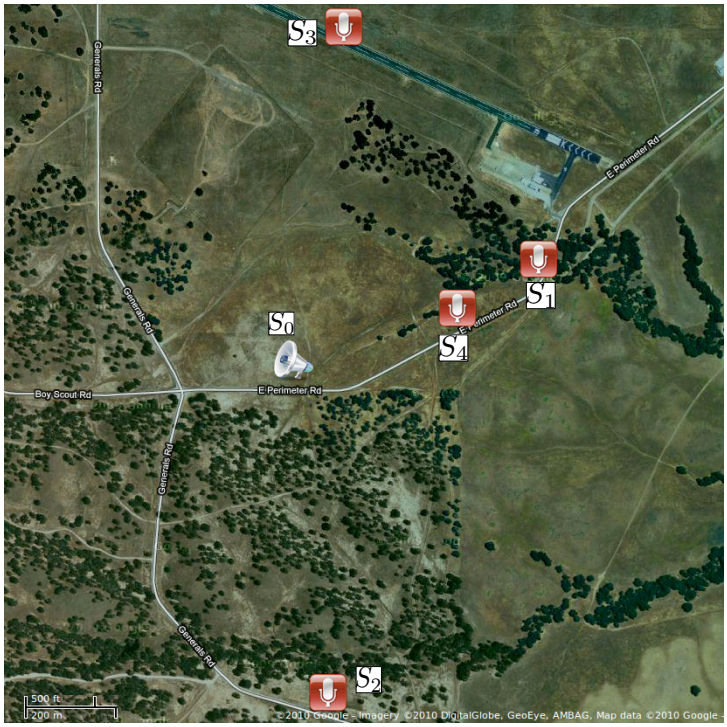
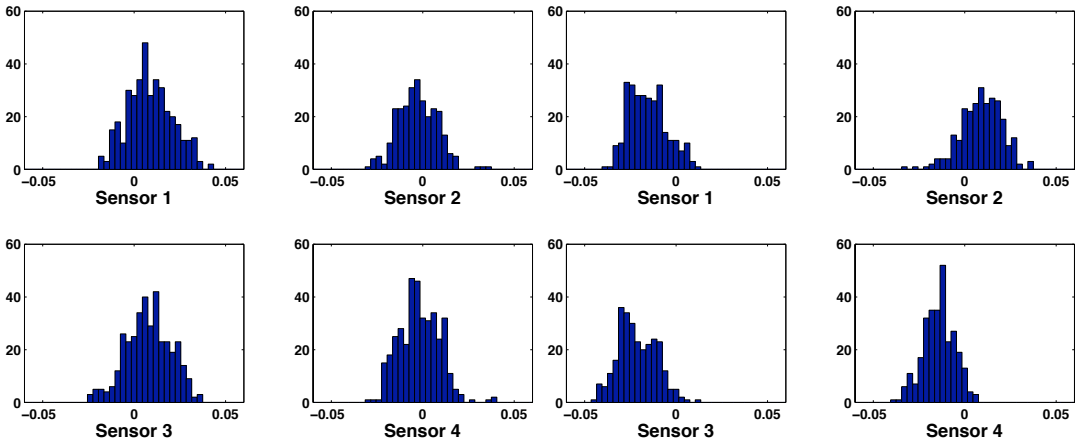


Figure 6.6: Sensor configuration for January 28, 2010 morning tests. In the afternoon tests, the location of sensor 4 was switched with the position of the acoustic source.

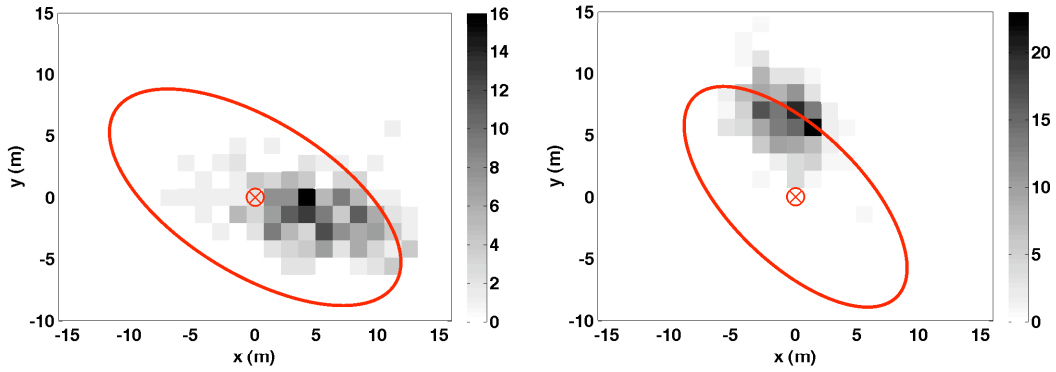
Table 6.3: Speed Of Sound Estimates

Date	Jan. 28, 2010	
Time	Temperature (°C)	ν (m/s)
10:20	7.004	335.6725
10:53	7.782	336.1384
11:53	11.729	338.4912
1:53	15.620	340.7950
3:53	16.731	341.4504
5:53	12.785	339.1181
7:53	10.617	337.8301



(a) TOA error histogram for morning tests. (b) TOA error histogram for afternoon tests.

Figure 6.7: Distribution of TOA error (in seconds) for each of four sensors.



(a) Localization histogram for morning tests. (b) Localization histogram for afternoon tests.

Figure 6.8: In both plots the true source location is indicated by the red x, and the 95% confidence ellipse centered at the true source location is shown in red.

Chapter 7

Conclusions and Future Directions

7.1 Summary

The coupling of static unattended ground sensors and mobile service vehicles has been explored. In particular, mobile data collectors enable sparse deployments of stationary sensors and extend the life of the network by reducing computation and communication requirements of the static nodes. This sensor network paradigm was applied directly to the problem of acoustic source localization using a small fixed wing UAV to collect measurements from stationary time of arrival acoustic sensors.

Sparse deployments were shown to be desirable for time of arrival sensors in acoustic source localization applications. We presented a vehicle routing strategy for the case of few sensors and seldom events that minimizes localization

time by adaptively selecting sensors to visit based upon observations from those recently visited. We later examined the case of many sensors in an area too large for all sensors to detect every event. In this context, the service vehicle must incorporate a search component into the routing policies. In problems such as these involving routing UAVs through a sequence of stationary sensors, costs can be reduced by including kinematic constraints of the UAV and regions in the environment where the communication is available to sensors on the ground in the optimization. We extended the work originated by Obermeyer in [68] on DTSPN to better take advantage of regions in the environment where communication is available with multiple sensors. Finally we have reported on the implementation and demonstration of an acoustic sensor network supported by a UAV data mule.

7.1.1 Future Directions

Optimal Sensor Placement: There are many directions in which this work could be extended. First, we are seeking to extend Theorems 2.3.2 and 2.3.5 to include any radially symmetric source distribution that may extend outside the area where sensors may be placed. Also, further analytical results could be provided regarding the advantage of splay states over other symmetric and balanced states.

In another direction, the radial symmetry of the source distribution should be relaxed. In an urban environment, the source distribution is unlikely to be radially symmetric. Further, sensor constraints other than a ring or annulus could be considered.

Strategies for Individual Source Localization: A natural next step is to develop architectures for coordinating multiple UAVs for covering larger deployment areas, under reasonable assumptions for their capabilities for communicating with each other. For larger deployment areas, it may also be useful to have a two-tier approach to the data provided by the sensors, such as a long-range, very low bit rate (even one bit), "alert" signal that draw in UAVs when the sensor detects an event of interest, together with shorter-range, higher bit rate communication that provides detailed measurements based on which the UAVs can make inferences. An important open problem is deriving bounds on the performance of the TTM algorithm. The TTM algorithm can potentially be improved by including UAV kinematic constraints in the shortest path portion of the optimization. This would ensure that the costs of paths planned for the UAV would be closer to the true cost of the UAV to fly these paths. It is also of interest to develop a better understanding of the acoustic channel (e.g., incorporating the effect of multipath) as well as of incorporating various sensing modalities into our architecture. Finally, it is of theoretical interest to reformu-

late the UAV routing problem to consider the impact of future measurements on future actions, and to find provably good approximations to such a problem.

Dubins Traveling Salesman Problem with Neighborhoods: There are many directions in which this work may be extended. First, it is of interest to understand the number of samples necessary to ensure that solutions meet acceptable performance targets. Second, we are seeking to extend these methods to include multiple UAVs. Third, it is of interest to include the amount of data to be downloaded at each sensor and radio bandwidth when planning paths. Finally, we are currently seeking to extend the work in [35] to obtain a constant factor approximation for the DTSPN.

Dynamic Vehicle Routing In A Large Sparse Sensor Network: A natural next step is to extend the analysis presented here to the case where sensors are deployed randomly, and a k-coverage algorithm such as the one presented in [54] is used to determine the sets $\mathcal{S}_{one}$ and $\mathcal{S}_{three}$. This provides the difficulty that lengths of the traveling salesman solutions are no longer known and must be approximated. We feel that while this would loosen the bounds on $\overline{W}$, the same analysis approach would go through. Another direction is to develop architectures for coordinating multiple UAVs for covering larger deployment areas, under reasonable assumptions for their capabilities for communicating with each other. Another potential improvement of the analysis presented here would be

to include UAV kinematic constraints and communication regions when computing the TSP portion of the policies. This would ensure that the costs of paths planned for the UAV would be closer to the true cost of the UAV to fly these paths.

Bibliography

[1] J. Abel and J. Smith. The spherical interpolation method for closed-form passive source localization using range difference measurements. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 471–474, Dallas, TX, April 1987.

[2] A.M. Ali, K. Yao, T.C. Collier, C.E. Taylor, D.T. Blumstein, and L. Girod. An empirical study of collaborative acoustic source localization. In *International Symposium on Information Processing in Sensor Networks*, pages 41–50, Cambridge, MA, April 2007.

[3] C. Andrieu, N. De Freitas, A. Doucet, and M.I. Jordan. An introduction to MCMC for machine learning. *Machine Learning*, 50(1-2):5–43, 2003.

[4] D. Applegate, R. Bixby, V. Chvátal, and W. Cook. Concorde TSP solver. Website: http//www.tsp.gatech.edu/concorde.

[5] D. Applegate, W. Cook, and A. Rohe. Chained Lin-Kernighan for large traveling salesman problems. *INFORMS Journal on Computing*, 15(1):82–92, 2003.

[6] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, et al. A line in the sand: A wireless sensor network for target detection, classification, and tracking. *Computer Networks*, 46(5):605–634, 2004.

[7] A. Beck, P. Stoica, and Jian Li. Exact and approximate solutions of source localization problems. *IEEE Transactions On Signal Processing*, 56(5):1770–1778, 2008.

[8] R. W. Bent and P. Van Hentenryck. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52(6):977–987, 2004.

[9] G. Berbeglia, J.-F. Cordeau, and G. Laporte. Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1):8–15, 2010.

[10] D. Bertsekas and R. Gallager. *Data Networks*. Prentice-Hall, 1987.

[11] D. J. Bertsimas and G. van Ryzin. A stochastic and dynamic vehicle routing problem in the Euclidean plane. *Operations Research*, 39(4):601–615, 1991.

[12] D. J. Bertsimas and G. van Ryzin. Stochastic and dynamic vehicle routing in the Euclidean plane with multiple capacitated vehicles. *Operations Research*, 41:60–76, 1993.

[13] D. J. Bertsimas and G. van Ryzin. Stochastic and dynamic vehicle routing with general demand and interarrival time distributions. *Advances in Applied Probability*, 25(4):947–978, 1993.

[14] D. Bhadauria, O. Tekdas, and V. Isler. Robotic data mules for collecting data over sparse sensor fields. *Journal of Field Robotics*, 28(3):388–404, 2011.

[15] M. M. Bin Tariq, M. Ammar, and E. Zegura. Message ferry route design for sparse ad hoc networks with mobile nodes. In *ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 37–48, Florence, Italy, May 2006.

[16] A. N. Bishop, B. Fidan, B. D. O. Anderson, P. N. Pathirana, and K. Doğançay. Optimality analysis of sensor-target geometries in passive lo-

calization: Part 2 - time-of-arrival based localization. In *International Conference on Intelligent Sensors, Sensor Networks and Information*, pages 13–18, Melbourne, Australia, December 2007.

[17] A. N. Bishop, B. Fidan, K. Doğançay, B. D. O. Anderson, and P. N. Pathirana. Exploiting geometry for improved hybrid AOA/TDOA-based localization. *Signal Processing*, 88(7):1775–1791, 2008.

[18] D. A. Bohn. Environmental effects on the speed of sound. *Journal of the Audio Engineering Society*, 36(4):223–231, 1988.

[19] J.-D. Boissonnat, A. Cérézo, and J. Leblond. Shortest paths of bounded curvature in the plane. *Journal of Intelligent and Robotics Systems*, 11(1-2):5–20, 1994.

[20] S. D. Bopardikar, S. L. Smith, F. Bullo, and J. P. Hespanha. Dynamic vehicle routing for translating demands: Stability analysis and receding-horizon policies. *IEEE Transactions on Automatic Control*, 55(11):2554 –2569, 2010.

[21] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[22] F. Bullo, E. Frazzoli, M. Pavone, K. Savla, and S. L. Smith. Dynamic vehicle routing for robotic systems. *Proceedings of the IEEE*, 99(9):1482 –1504, 2011.

[23] J. J. Caffery and G. L. Stuber. Overview of radiolocation in CDMA cellular systems. *IEEE Communications Magazine*, 36(4):38–45, 1998.

[24] G. D. Çelik and E. Modiano. Dynamic vehicle routing for data gathering in wireless networks. In *IEEE Conference on Decision and Control*, pages 2372–2377, Atlanta, GA, December 2010.

[25] Y. T. Chan and K. C. Ho. A simple and efficient estimator for hyperbolic location. *IEEE Transactions on Signal Processing*, 42(8):1905–1915, 1994.

[26] C.-Y. Chong and S. P. Kumar. Sensor networks: Evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003.

[27] N. Christofides. Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Report 388, Carnegie-Mellon University, Graduate School of Industrial Administration, Pittsburgh, PA, 1976.

[28] M. Chu, H. Haussecker, and F. Zhao. Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks. *Interna-*

tional Journal of High Performance Computing Applications, 16(3):293, 2002.

[29] D. Ciullo, G. D. Çelik, and E. Modiano. Minimizing transmission energy in sensor networks via trajectory control. In *International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, pages 132–141, Avignon, France, June 2010.

[30] G. B. Dantzig and J. H. Ramser. The truck dispatching problem. *Management Science*, 6:80–91, 1959.

[31] M. Desrochers, J. Desrosiers, and M. Solomon. A new optimization algorithm for the vehicle routing problem with time windows. *Operations Research*, 40(2):342–354, 1992.

[32] K. Doğançay. Optimal receiver trajectories for scan-based radar localization. In *Information, Decision and Control*, pages 88–93, Adelaide, Australia, February 2007.

[33] K. Doğançay and H. Hmam. Optimal angular sensor separation for AOA localization. *Signal Processing*, 88(5):1248–1260, 2008.

[34] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–516, 1957.

[35] A. Dumitrescu and J. S. B. Mitchell. Approximation algorithms for TSP with neighborhoods in the plane. *Journal of Algorithms*, 48(1):135 – 159, 2003.

[36] K. Elbassioni, A. V. Fishkin, N. H. Mustafa, and R. Sitters. Approximation algorithms for Euclidean group TSP. In *International Colloquim on Automata, Languages and Programming*, pages 1115–1126, Lisbon, Portugal, July 2005.

[37] J. J. Enright. *Efficient Routing of Multi-Vehicle Systems: Limited sensing and nonholonomic motion constraints*. PhD thesis, University of California Los Angeles, 2008.

[38] J. J. Enright and E. Frazzoli. Cooperative UAV routing with limited sensor range. In *AIAA Conference on Guidance, Navigation, and Control*, Keystone, CO, August 2006.

[39] E. Frazzoli and F. Bullo. Decentralized algorithms for vehicle routing in a stochastic time-varying environment. In *IEEE Conference on Decision*

and Control, pages 3357–3363, Paradise Island, Bahamas, December 2004.

[40] E. Frew, C. Dixon, B. Argrow, and T. Brown. Radio source localization by a cooperating UAV team. In *AIAA Infotech@Aerospace*, pages 10–20, Arlington, VA, September 2005.

[41] A. Frieze, G. Galbiati, and F. Maffioli. On the worst-case performance of some algorithms for the asymmetric traveling salesman problem. *Networks*, 12:23–39, 1982.

[42] S.W Fuhrmann. Symmetric queues served in cyclic order. *Operations Research Letters*, 4(3):139 – 144, 1985.

[43] M. Gendreau, F. Guertin, J.-Y. Potvin, and E. Taillard. Parallel tabu search for real-time vehicle routing and dispatching. *Transportation Science*, 33:381–390, 1999.

[44] J. H. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2:84–90, 1960.

[45] A. Hasler, I. Talzi, C. Tschudin, and S. Gruber. Wireless sensor networks in permafrost research - concept, requirements, implementation and

challenges. In *International Conference on Permafrost*, pages 669–674, Fairbanks, AK, July 2008.

[46] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[47] M. Hefeeda and M. Bagheri. Forest fire modeling and early detection using wireless sensor networks. *Ad Hoc Sensor Wireless Networks*, 7:169–224, 2009.

[48] D. Henkel and T. X. Brown. On controlled node mobility in delay-tolerant networks of unmanned aerial vehicles. In *International Symposium on Advance Radio Technolgoies*, pages 7–9, Boulder, CO, March 2005.

[49] D. Henkel, C. Dixon, J. Elston, and T. X. Brown. A reliable sensor data collection network using unmanned aircraft. In *International Workshop on Multi-hop Ad Hoc Networks: From Theory to Reality*, pages 125–127, Florence, Italy, May 2006.

[50] G. M. Hoffmann and C. J. Tomlin. Mobile sensor network control using mutual information methods and particle filters. *IEEE Transactions on Automatic Control*, 55(1):32–47, 2010.

[51] Y. Huang, J. Benesty, and G. W. Elko. Passive acoustic source localization for video camera steering. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 909–912, Istanbul, Turkey, June 2000.

[52] R. M. Karp. Reducibility among combinatorial problems. *Complexity of Computer Computations*, 40(4):85–103, 1972.

[53] D. J. Klein, J. Schweikl, J. T. Isaacs, and J. P. Hespanha. On UAV routing protocols for sparse sensor data exfiltration. In *American Control Conference*, pages 6494–6500, Baltimore, MD, June 2010.

[54] S. Kumar, T. H. Lai, and J. Balogh. On k-coverage in a mostly sleeping sensor network. In *International Conference on Mobile Computing and Networking*, pages 144–158, Philadelphia, PA, October 2004.

[55] A. Larsen, O. Madsen, and M. Solomon. Partially dynamic vehicle routing-models and algorithms. *The Journal of the Operational Research Society*, 53(6):637–646, 2002.

[56] R. C. Larson and A. R. Odoni. *Urban Operations Research*. Prentice Hall, 1981.

[57] J. Le Ny, E. Feron, and E. Frazzoli. On the curvature-constrained traveling salesman problem. *IEEE Transactions on Automatic Control*, To appear.

[58] J. Le Ny, E. Frazzoli, and E. Feron. The curvature-constrained traveling salesman problem for high point densities. In *IEEE Conference on Decision and Control*, pages 5985–5990, New Orleans, LA, December 2007.

[59] W.-K. Ma, B. Vo, S. S. Singh, and A. Baddeley. Tracking an unknown time-varying number of speakers using TDOA measurements: A random finite set approach. *IEEE Transactions on Signal Processing*, 54(9):3291–3304, 2006.

[60] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. Wireless sensor networks for habitat monitoring. In *ACM International Workshop on Wireless Sensor Networks and Applications*, pages 88–97, Atlanta, GA, 2002.

[61] S. Martínez and F. Bullo. Optimal sensor placement and motion coordination for target tracking. *Automatica*, 42(4):661–668, 2006.

[62] J. M. Mendel. *Lessons in Estimation Theory for Signal Processing, Communications, and Control*. Prentice-Hall, 1995.

[63] A. R. Mesquita, J. P. Hespanha, and K. Åström. Optimotaxis: A stochastic multi-agent on site optimization procedure. In M. Egerstedt and B. Mishra,

editors, *Hybrid Systems: Computation and Control*, number 4981 in Lecture Notes in Computer Science, pages 358–371. Springer-Verlag, 2008.

[64] J. Neering, M. Bordier, and N. Maizi. Optimal passive source localization. In *International Conference on Sensor Technologies and Applications*, pages 295–300, Valencia, Spain, October 2007.

[65] C. E. Noon and J. C. Bean. An efficient transformation of the generalized traveling salesman problem. Technical Report 89-36, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, 1989.

[66] C. E. Noon and J. C. Bean. An efficient transformation of the generalized traveling salesman problem. Technical Report 91-26, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, 1991.

[67] K. J. Obermeyer. Path planning for a UAV performing reconnaissance of static ground targets in terrain. In *AIAA Conference on Guidance, Navigation, and Control*, Chicago, IL, August 2009.

[68] K. J. Obermeyer, P. Oberlin, and S. Darbha. Sampling-based roadmap methods for a visual reconnaissance UAV. In *AIAA Conference on Guid-*

ance, Navigation, and Control, Toronto, ON, Canada, August 2010.

[69] Y. Oshman and P. Davidson. Optimization of observer trajectories for bearings-only target localization. *IEEE Transactions on Aerospace and Electronic Systems*, 35(3):892–902, 1999.

[70] M. Pavone, N. Bisnik, E. Frazzoli, and V. Isler. A stochastic and dynamic vehicle routing problem with time windows and customer impatience. *Mobile Networks and Applications*, 14(3):350–364, 2009.

[71] V. Polishchuk, E. M. Arkin, and J. S. B. Mitchell. Hamiltonian cycles in triangular grids. In *Canadian Conference on Computational Geometry*, pages 63–66, Kingston, ON, Canada, August 2006.

[72] N.B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *International Conference on Mobile Computing and Networking*, pages 32–43, Boston, MA, August 2000.

[73] H.N. Psaraftis. Dynamic vehicle routing problems. *Vehicle Routing: Methods and Studies*, 16:223–248, 1988.

[74] S. Rathinam, R. Sengupta, and S. Darbha. A resource allocation algorithm for multiple vehicle systems with non-holnomic constraints. *IEEE Transactions on Automation Science and Engineering*, 4(1):98–104, 2007.

[75] A. Ryan and J. K. Hedrick. Particle filter based information-theoretic active sensing. *Robotics and Autonomous Systems*, 58(5):574–584, 2010.

[76] M. W. P. Savelsbergh and M. Sol. The general pickup and delivery problem. *Transportation Science*, 29:17–29, 1995.

[77] K. Savla, E. Frazzoli, and F. Bullo. Traveling salesperson problems for the Dubins vehicle. *IEEE Transactions on Automatic Control*, 53(6):1378 –1391, 2008.

[78] H. Schau and A. Robinson. Passive source localization employing intersecting spherical surfaces from time-of-arrival differences. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(8):1223–1225, 1987.

[79] R.O. Schmidt. A new approach to geometry of range difference location. *IEEE Transactions on Aerospace and Electronic Systems*, 8(6):821–835, 1972.

[80] R.C. Shah, S. Roy, S. Jain, and W. Brunette. Data MULEs: Modeling a three-tier architecture for sparse sensor networks. In *IEEE International Workshop on Sensor Network Protocols and Applications*, pages 30–41, Anchorage, AK, May 2003.

[81] A. M. Shkel and V. Lumelsky. Classification of the Dubins set. *Robotics and Autonomous Systems*, 34(4):179–202, 2001.

[82] G. Simon, M. Maróti, Á. Lédeczi, G. Balogh, B. Kusy, A. Nádas, G. Pap, J. Sallai, and K. Frampton. Sensor network-based countersniper system. In *ACM Conference on Embedded Networked Sensor Systems*, pages 1–12, Baltimore, MD, November 2004.

[83] M. M. Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35:254–265, 1987.

[84] R. Sugihara and R. K. Gupta. Path planning of data mules in sensor networks. *ACM Transactions Senor Networks*, 8:1:1–1:27, 2011.

[85] H. Takagi. Mean message waiting times in symmetric multi-queue systems with cyclic service. *Performance Evaluation*, 5(4):271–277, 1985.

[86] P. Toth and D. Vigo, editors. *The Vehicle Routing Problems*. SIAM Society for Industrial and Applied Mathematics, 2002.

[87] D. Uciński. *Optimal Measurement Methods for Distributed Parameter System Identification*. CRC Press, 2004.

[88] Weather Underground. History for CA62, California, 2010. `http://www.wunderground.com/history/airport/CA62/2010/1/28/`

DailyHistory.html?req_city=NA&req_state=NA&req_statename=NA [Online; accessed 25-March-2010].

[89] H. L. Van Trees. *Detection, Estimation, and Modulation Theory Part I.* John Wiley & Sons, 1968.

[90] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke. Data collection, storage, and retrieval with an underwater sensor network. In *ACM Conference on Embedded Networked Sensor Systems*, pages 154–165, San Diego, CA, November 2005.

[91] V. V. Vazirani. *Approximation Algorithms.* Springer, 2001.

[92] S. Venkateswaran and U. Madhow. Space-time localization from times of arrival. In *Allerton Conference on Communication, Control, and Computing*, pages 1544–1551, Monticello, IL, September 2011.

[93] S. Venkatraman and J. Caffery Jr. Hybrid TOA/AOA techniques for mobile location in non-line-of-sight environments. In *IEEE Wireless Communications and Networking Conference*, pages 274–278, Atlanta, GA, March 2004.

[94] P. Volgyesi, G. Balogh, A. Nadas, C.B. Nash, A. Ledeczi, K. Pence, T. Bapty, J. Scott, and The Nashville Police. Shooter localization and

weapon classification with soldier-wearable networked sensors. In *Conference on Mobile Systems, Applications, and Services*, pages 113–126, San Juan, Puerto Rico, June 2007.

[95] H. A. Waisanen, D. Shah, and M. A. Dahleh. A dynamic pickup and delivery problem in mobile networks under information constraints. *IEEE Transactions on Automatic Control*, 53(6):1419–1433, 2008.

[96] H. B. Wang, C. E. Chen, A. M. Ali, S. Asgari, R. E. Hudson, K. Yao, D. Estrin, and C. Taylor. Acoustic sensor networks for woodpecker localization. In F. T. Luk, editor, *SPIE Advanced Signal Processing Algorithms, Architectures, and Implementations*, volume 5910, August 2005.

[97] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh. Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing*, 10(2):18–25, 2006.

[98] P. M. White. Using a Garmin GPS 18 LVC as NTP stratum-0 on Linux 2.6, 2012. `http://time.qnan.org/` [Online; accessed 20-March-2012].

[99] Wikipedia. Geodetic system — wikipedia, the free encyclopedia, 2012. `http://en.wikipedia.org/w/index.php?title=Geodetic_system&oldid=480516513` [Online; accessed 20-March-2012].

[100] N. Xu, S. Rangwala, K. K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin. A wireless sensor network for structural monitoring. In *ACM Conference on Embedded Networked Sensor Systems*, pages 13–24, Baltimore, MD, November 2004.

[101] S. Yadlapalli, W. A. Malik, S. Rathinam, and S. Darbha. A Lagrangian-based algorithm for a combinatorial motion planning problems. In *IEEE Conference on Decision and Control*, pages 5979–5984, New Orleans, LA, December 2007.

[102] B. Yang. Different sensor placement strategies for TDOA based localization. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 1093–1096, Honolulu, HI, April 2007.

[103] B. Yang and J. Scheuing. Ćramer-Rao bound and optimum sensor array for source localization from time differences of arrival. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 961–964, Philadelphia, PA, March 2005.

[104] B. Yang and J. Scheuing. A theoretical analysis of 2D sensor arrays for TDOA based localization. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 901–904, Toulouse, France, May 2006.

[105] B. Yuan, M. Orlowska, and S. Sadiq. On the optimal robot routing problem in wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, 19(9):1252–1261, 2007.

[106] H. Zhang and J. C. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc & Sensor Wireless Networks*, 1:89–123, 2005.

[107] F. Zhao, J. Shin, and J. Reich. Information-driven dynamic sensor collaboration for tracking applications. *IEEE Signal Processing Magazine*, 19(2):61–72, 2002.

[108] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 187–198, Tokyo, Japan, May 2004.