ROBOTIC SEARCHING FOR STATIONARY, UNKNOWN AND TRANSIENT RADIO SOURCES

A Dissertation

by

CHANG YOUNG KIM

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2012

Major Subject: Computer Engineering

ROBOTIC SEARCHING FOR STATIONARY, UNKNOWN AND TRANSIENT RADIO SOURCES

A Dissertation

by

CHANG YOUNG KIM

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Dezhen Song
Committee Members,	Ricardo Gutierrez-Osuna
	Radu Stoleru
	Byung-Jun Yoon
Head of Department,	Duncan Walker

May 2012

Major Subject: Computer Engineering

ABSTRACT

Robotic Searching for
Stationary, Unknown and Transient Radio Sources. (May 2012)
Chang Young Kim, B.S., Korea University
Chair of Advisory Committee: Dr. Dezhen Song

Searching for objects in physical space is one of the most important tasks for humans. Mobile sensor networks can be great tools for the task. Transient targets refer to a class of objects which are not identifiable unless momentary sensing and signaling conditions are satisfied. The transient property is often introduced by target attributes, privacy concerns, environment constraints, and sensing limitations. Transient target localization problems are challenging because the transient property is often coupled with factors such as sensing range limits, various coverage functions, constrained mobility, signal correspondence, limited number of searchers, and a vast searching region.

To tackle these challenge tasks, we gradually increase complexity of the transient target localization problem such as Single Robot Single Target (SRST), Multiple Robots Single Target (MRST), Single Robot Multiple Targets (SRMT) and Multiple Robots Multiple Targets (MRMT). We propose the expected searching time (EST) as a primary metric to assess the searching ability of a single robot and the spatiotemporal probability occupancy grid (SPOG) method that captures transient characteristics of multiple targets and tracks the spatiotemporal posterior probability distribution of the target transmissions. Besides, we introduce a team of multiple robots and develop a sensor fusion model using the signal strength ratio from the paired robots in centralized and decentralized manners. We have implemented and validated the algorithms under a hardware-driven simulation and physical experiments.

To Sujin and Jun

ACKNOWLEDGMENTS

This work would not have been possible without the support of my colleagues, mentors and friends.

Specially, I would like to express my deepest gratitude to Dezhen Song for his continuous support, and mentorship, throughout the entire course of my Ph.D. As my advisor, mentor and best friend, Dez is always willing to spend time with me and inspire me to explore new problems. His professional guidance assisted, armed me with invaluable tools for life, and filled me with an exciting anticipation for the future to come.

I would also like to thank Ricardo Gutierrez-Osuna, Radu Stoleru, Byung-Jun Yoon for serving as my dissertation committee members and their insightful suggestions and inspiring discussions on advancing my research.

Thanks J. Yi, K. Goldberg, and R. Volz for being great research collaborators and their insightful discussions. Thanks D. Shell and B. Fine for sharing iRobot Create robots for the physical experiments. Thanks Yiliang for being my great roommate, labmate and friend. Thanks to Q. Hu, Z. Goodwin, H. Lee, J. Zhang, W. Li, Y. Lu, and H. Ge for their continuous help and contributions to my Ph.D. projects.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

On June 1st, 2009, Air France Flight 447 plunged into the Atlantic Ocean without a trace. Along with it were 216 passengers and 12-person aircrews. Speculations about the cause of the tragedy vary from alien attack to equipment malfunction. Finding the flight data recorder (i.e. the black box) is crucial to solving the mystery and improving air travel safety. Lying on the vast ocean floor, the black box awaited and emitted acoustic ping signals from time to time. Unfortunately, the battery of the black box depleted in 30 days and the search process missed the important time window. The following painstaking search has to rely on the extensive scan of the sea floor. The black box was eventually found by an underwater robot, almost two years after the plane crash. The important lesson is: can we be more efficient in searching for such a transient target?

By extending our motivation to multiple targets, we intend to provide a countermeasure for the potential misuse of the fast-developing sensor network technology. The wireless sensor networks is composed of a large number of low-cost miniature wireless embedded sensor nodes with self-configurable *ad hoc* networking capabilities[1]. Since the wireless sensor networks do not relay on a deployed network infrastructure as cellular or wireless LAN networks, they appear to useful tools for military applications such as surveillance networks, target monitoring and real-time information distribution as well as civilian applications such as emergency rescue operations and home networking [2]. While wireless sensor networks provide us with significant advantages in terms of time, space and cost, they provide new espionage tools that threaten

---

This dissertation follows the style of *IEEE Transactions on Robotics*.

our security when an adversary deploys the wireless sensor networks to monitor our movements, to trigger explosives, and to notify enemy agents. Besides, enemies can deploy wireless sensor networks to guard and protect critical infrastructures in battlefield including oil-field, airports and power plants. In addition to the threads in the battlefield, they can threaten civilian's privacy when someone deploys hostile sensor networks equipped with cameras, microphones, motion sensors, etc. to areas of interest to collect the useful information without notice and permission. Thus, developing a localization scheme of multiple targets in the wireless sensor networks is crucial to protect us from these threads.

Searching for objects in physical space is one of the most important tasks for humans. Searching for radio targets has a long history since Heinrich Hertz [3] discovered the directional property of radio waves in 1893. Mobile sensor networks can be great tools for the task. Transient targets refer to a class of objects which are not identifiable unless momentary sensing and signaling conditions are satisfied. The detection of transient signals in a vast searching region is often constrained by sensing range and coverage, limited number of sensors, mobility and signal correspondence. The transient property is often introduced by target attributes, privacy concerns, environment constraints, and sensing limitations. For example, a black box which helps an investigation after an airplane crash, transmits radio signals periodically to assist searching. Victims after an earthquake or a mine collapse disaster knock the rubble from time to time. A lost hiker's cell phone may still transmit signals despite its owner may have lost consciousness.

Transient target localization problems are challenging because the transient property is often coupled with factors such as sensing range limits, various coverage functions, constrained mobility, signal correspondence, limited number of searchers, and a vast searching region. Unlike a regular search, where the collocation between a target

and the sensing range of a searcher usually means the searching task is accomplished, the transient target localization requires both the collocation condition and an active signal transmitted by the target. Without knowledge of the network configuration and packet structure, signal source anonymity in multiple targets produce a data association problem.

To detect signals from transient targets, we only use Received Signal Strength (RSS) [4] even though there exist two other range measurement techniques such as Time of Arrival (TOA)[5], and Time Difference of Arrival (TDOA)[6] because these two techniques require the knowledge of the network configuration and packet structure which are not available in the unknown sensor network. Although signal fingerprinting [7] allows the searcher to distinguish between signals from different radio sources, we do not exploit the signal fingerprinting due to its expensive cost and specialized hardware. Therefore, we can only rely on RSS from intercepted signals.

Searching for the transient radio targets is a time-consuming and tedious task because the searching task can be done only by satisfying the certain space and time constrains where the target must be located within the sensing range of a searcher and when the target emits the signal. Besides, this task may be operated in the dangerous environment. For example, the black box could be located under deep sea after the airplane crash and the radio target could be posed by an enemy in the battlefield. To perform these tedious and dangerous searching tasks, we take advantage of a recent robotic technologies using a mobile robot or a team of mobile robots.

In this dissertation, we gradually increase the complexity of the transient target localization problems by relaxing assumptions close to a more realistic localization problem. Depending on the number of robots used and targets localized, the transient target localization problems are categorized by four different difficulty levels with increasing complexity such as Single Robot Single Target (SRST), Multiple Robots

Single Target (MRST), Single Robot Multiple Targets (SRMT) and Multiple Robots Multiple Targets (MRMT).

We start with the single target localization including SRST and MRST localizations. We focus on the time to search for a target because localizing single target is trivial using the triangulation once the robot or the team of robots detects the signal from the target. For many searching and rescue applications, searching time is crucial to save victim's life. To assess the searching ability of the robot, we propose the expected searching time (EST) as a primary metric. By increasing the number of targets with a single robot, the localization problem becomes SRMT localization. Localizing multiple transient targets cannot be viewed as a simple extension of the single target localization due to challenging issues such as signal anonymity, transiency, and unknown target number often couple together. Due to signal anonymity, multiple radio targets arise a data association problem to determine which radio target comes from when a robot perceives a signal. To resolve it, we assume that sensor network nodes employ a carrier sensing multiple access (CSMA)-based Media Access Control (MAC) protocol. Then, we propose a CSMA SRMT localization scheme. The CSMA SRMT localization scheme suffers from a scalability issue and dependency on the CSMA protocol. The drawbacks of the CSMA SRMT localization scheme drives us to develop a protocol independent SRMT localization scheme. The protocol independent SRMT localization scheme is insensitive to the number of radio sources and hence its algorithm has great scalability. We relax an assumption that robots know the source transmission power. To remove the dependence on the source power level, we introduce a team of multiple robots in the centralized manner under an assumption of an unlimited communication range. For the centralized MRMT localization, we develop a sensor fusion model using the signal strength ratio from the paired robots. With the limited communication range, we finally develop a decentralized MRMT localiza-

tion method including decentralized belief functions that robots use to track source locations using checkpoint-based synchronization, and a motion planning strategy to coordinate robots to ensure the existence of checkpoints and coordinated searching. Table I summaries the assumptions in terms of localization problems.

Table I.: Summary of assumptions with respect to localization problems.

	CSMA	Trans. Power	Sensing Range	Com. Range
CSMA SRMT	O	Known	Unlimit	Not Applicable
Protocol Independent SRMT	X	Known	Unlimit	Not Applicable
Centralized MRMT	X	Unknown	Limit	Unlimit
Decentralized MRMT	X	Unknown	Limit	Limit

We have implemented and validated the algorithms under a hardware-driven simulation and physical experiments and extensively tested them under different system setting in comparison to a random walk and a fixed-route patrol heuristics. The our algorithms significantly outperforms the two heuristics.

A. Organization of the Dissertation

The rest of this dissertation is organized as follows. We study the problems under different setups and constraints by gradually increasing complexity. First, we proposed SRST and MRST localizations, focusing on the EST to assess the searching ability in Chapter III. Second, we propose the CSMA SRMT localization method under the assumption that the CSMA based protocol is used among networked radio targets in Chapter IV. Third, after relaxing the CSMA based protocol, we develop the protocol independent SRMT localization method using a spatiotemporal probability occupancy grid (SPOG) in Chapter V. Fourth, our attention shifts to the centralized MRMT localization method in Chapter VI and the decentralized MRMT localization method in Chapter VII. Finally, Chapter VIII concludes this research work with a

summary and some directions for future work.

CHAPTER II

RELATED WORK

A. Searching for a Single Target

Searching for an object in physical space is one of the most important tasks for robots or humans. When prior information such as the spatial distribution of the target is known, this is comparable to the foraging behavior of animals [8]. For this case, the searching problem is actually a non-linear optimization problem in operations research [9] with searching time/path length as objective functions. If the searching space is a graph, Trummel and Weisinger show this is an NP-hard problem [10].

However, prior target information is often not available. If the target is continuously emitting signals, just simply scanning the entire searching space once enables the robot to find the target. Since the maximum searching time is the time to cover the entire searching space, the searching problem becomes a coverage problem [11, 12, 13]. For a known environment, a coverage problem for a single robot often employs different approaches to decompose the searching space and output a continuous path that allows the robot to cover the entire searching space. If the searching space can be modeled as a set of w-disjoint discrete choices, searching for a target with a limited sensing range and w-choice is known as a w-lane Cow-Path problem[14]. For coverage problems over a small region and considering scene topologies, the problem can be reduced to the the Art Gallery (AG) problem in computational geometry [15, 16]. In Euclidean space, if the coverage strategy is to linearly partition the searching space to form a connected path, this know as the slap method or the trapezoid [17, 18]. Independently, this method is regarded as a type of linear search problems in applied math and operations research societies [19].

While the searching time is well understood for the coverage problems [20, 21], this is not the case when the searching process depends on the signal emitted by the target because the collocation of the robot and the target does not necessarily mean that the target is found. Benkoski et al. [22] classify searching problems into two categories: stationary targets and moving targets. Our problem deals with stationary targets. However, Benkoski's classification does not include cases that the target emits transmit signals, which is different from the one-sided search that Benkoski's classification is based on.

B. Mobile Robots and Sensor Networks

The fast development of robotics and sensor network research has accumulated many related work including Simultaneous Localization and Mapping (SLAM), occupancy grid methods and multi-robot sensor coordination.

In robotics research, SLAM is defined as the process of mapping the environment and localizing robot position at the same time [23, 24, 25, 26, 27]. Although both SLAM and our research are based Bayesian methods, SLAM assumes that the environment is static or close to static. Directly applying SLAM methods to our problem is not appropriate because networked radio sources create a highly dynamic environment where the signal transmission patterns change quickly. Although recent advances in SLAM allow tracking of moving objects[28] while performing SLAM task, the environment largely remains static.

Since Elfes and Moravec [29, 30] introduce occupancy grid maps as a probabilistic sensor model, the occupancy grid has been proved to be an elegant representation of the sensor coverage for mobile robot applications such as localization and mapping[31]. Thrun and his colleagues [32] further improve occupancy grid maps to

incorporate multi-sensor fusion, an inverse sensor model, and a forward sensor model. The existing occupancy grid-based methods focus on using the spatial probabilistic representation to describe sensing uncertainty and are not capable of dealing with time-variant environments. Our work extends the occupancy grid methods into the temporal dimension to deal with the dynamic characteristics of the transient radio transmissions.

Recent advances in using a multi-robot team to perform exploration and mapping tasks mainly focuses on the coordination of the robot/sensor team [33, 34, 35, 36] under various dynamics, communication, sensing, and energy constraints. Although not directly applicable to our problem, researchers have accumulated interesting empirical results: using a team of low-cost robots usually performs faster and more fault-tolerant [37] than a single expensive robot. This really inspires our problem because we want to see if our analytical model can show similar results under similar constraints/conditions.

C. Radio Frequency-based Localization

The recent development of radio frequency-based localization can be viewed as the localization of "friendly" radio sources because researchers either assume that an individual radio source that continuously transmits radio signals (similar to a lighthouse) [38, 39, 40, 41] or assume that the robot/receiver is a part of the network which understands the detailed packet information [42, 43, 44, 45, 46, 47, 48]. However, such information is not always available in an unknown network. In a recent work [38], Letchner et al. use a network of wireless access points to localize a mobile unit. This can be viewed as a dual version of our problem. They use multiple static listeners to localize a mobile transmitter, while we try to localize multiple static transmitters

using a mobile listener. In another closely related work [45], Sichitiu and Ramadurai localize sensor network nodes with a mobile beacon. Again, the mobile beacon and the sensor network nodes share the network information.

When signal sources are not cooperative, RSS readings are the primary information for localization because RSS attenuates over distance. Since signal transmission power at the source is not available, ratios between RSS readings from dislocated listeners have been proven to be effective [49, 50, 51]. Li et al. [52] shows that at least four robots are needed at the same moment in order to localize a single source with unknown transmission power. Another approach is to use antenna arrays that can provide bearing readings. In recent work [53], Kim and Chong show how to find a radio source using two antennas with different polarizations. These approaches focus on localizing a single source at a time and hence are not concerned with the signal correspondence issue in their setup.

D. Decentralized Localization

Decentralized localization using multiple robots closely related to decentralized estimation and decentralized motion planning. For the decentralized estimation, Durrant-Whyte et al. [54, 55, 56] develop the decentralized estimation technology for the static sensor network using the information filter based on the distributed Kalman filter framework. Based on relative observations between multiple robots, researchers propose various decentralized multi-robots localization schemes by the decentralized extended Kalman filter (EKF)[57] which is decomposed into a number of smaller communicating filters and by the decentralized maximum a posteriori (MAP) estimator [58]. Leung et al. [59] also tackle the decentralized multi-robots localization problem by a concept of the checkpoint which is delayed synchronization of observation after

exchanging observations between robots. Researchers extend the checkpoint concept to Decentralized-Simultaneous Localization and Mapping (D-SLAM) [60] and the decentralized information transfer scheme [61] based on communication resources available. Capitan et al. [62] also propose a delayed synchronization for decentralized estimation using the information filter. Since the decentralized estimation methods assume that the environment is static or close to static, directly applying decentralized estimation methods to our problem is not appropriate because networked radio sources create a highly dynamic environment where the signal transmission patterns change very quickly. Note that we will extend the checkpoint concept to our problem.

In multi-robots research, Pereira et al. [63] propose the decentralized motion planning under sensing and communication constrains while keeping connectivity with the neighbors. By using decentralized multi-robots, Bhadauria et al. [64] address the Data Gathering Problem (DGP) in which multiple robots gather information from deployed sensor networks. In this work, they formulate the DGP as Travel Salesman Problem (TSP) instances and propose the two subtours motions which composes of the counterclockwise tour and the clockwise tour and ensures that two tours covers entire deployed sensor nodes. Another aspect in the decentralized motion planning is synchronization. Martinez et al. [12, 65] analyze motion synchronization of decentralized multi-robots introducing a network of locally connected agents on the circle using the agree-and-pursue algorithm. In our problem, we focus on information synchronization rather than motion synchronization.

CHAPTER III

SRST AND MRST LOCALIZATIONS*

A. Introduction

Mobile robots are often employed to perform searching tasks such as finding a black box in a remote area after an airplane crash, searching for victims after an earthquake or a mine collapse disaster, or locating artifacts on the ocean floor. The searching task is accomplished once the robot detects the signal emitted by the target. However, the robot usually has a limited sensing range. It seems straightforward that we can use the traditional coverage-based motion plans to guide the robot to cyclically scan the searching space to locate the target. However, the time to search for the target is inherently random and hence remains unknown despite its importance in many searching and rescue applications.

To address this new problem, we propose the expected searching time (EST) as a primary metric to assess the searching ability of a robot or a robot team. We model the searching process as a delayed renewal reward process [66] and derive the EST as a function of the searching space size, the signal transmission rate, and the robot sensing range. The resulting closed-form solution of the metrics can be used to analyze the searching efficiency for different robot configurations and searching plans. Since the model components can be obtained from online measurements and known robot parameters, a great benefit of the resulting model is that it is capable of predicting the EST for an ongoing searching process. This characteristic is important for time-

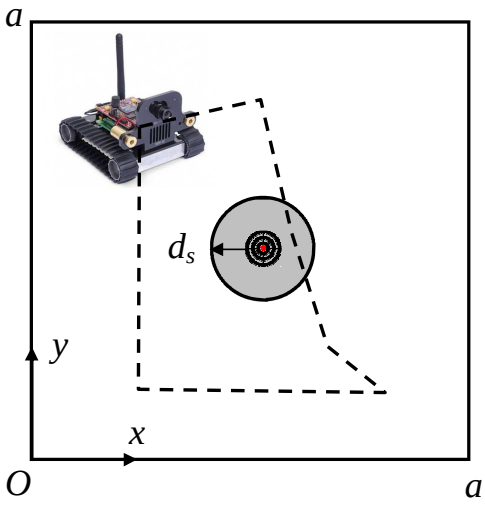


Fig. 1.: A robot attemps to search for a target (the red dot) that intermittently emits short duration signals in a square. The gray circle is the region that the robot can sense the signal from the target. The dashed line is a robot trajectory.

critical searching and rescue applications. The analytical results are confirmed in simulation and physical experiments.

B. Problem Definition

As illustrated in Fig. 1, a single robot searches for a single target in a squared 2D Euclidean space with a side length of a. Define d_s as the maximum sensing distance of the sensor on the robot. The robot travels at the constant speed of v. To formulate the problem and focus on the most relevant issues, we make the following assumptions,

1. There is no prior information about possible locations of the target. Therefore, the target is assumed to be uniformly distributed in the searching space.

2. The target transmits short duration signals periodically according to a Poisson process with a known rate λ. The signal duration is short due to energy concerns. A Poisson process is a good approximation to a general random arrival

process in stochastic modeling [66]. In some cases, the target may be a continuous beacon; but it is very difficult to be detected due to environment conditions or unreliable sensing, which can also be modeled as a target with intermittent signals.

3. During the searching process, the target is static. The searching space is much larger than the sensing distance: $a \gg d_s$.

Condition 1 (Sensing Condition)**.** *The robot cannot sense the signal unless an actively-transmitting target is within distance d_s due to the sensing range limit.*

As illustrated in Fig. 1, Condition 1 defines a circle centered at the target with the radius of d_s, which is the region that robot has a chance to sense the target. We refer to the region as "the circle" in the rest of the paper. Due to the fact that the robot does not know the location of the target, the actual position of the circle in the searching space is also unknown.

Condition 2 (Termination Condition)**.** *The searching task is accomplished as soon as the robot senses a signal sent by the target.*

Condition 2 implies that the robot cannot find an inactive target even it is collocated with the target. For example, an airplane is not be able to notice the survivor on an island if the person does not send a signal (e.g. fire or smoke). On the other hand, only one signal reception is needed in the searching process. Conditions 1 and 2 establish a new type of searching problem as oppose to a regular coverage problem. Let us define T_s as the searching time for the robot to find the target. Therefore, our problem is defined as follows,

Problem 1 (The EST Computation)**.** *Given λ, d_s, and a, calculate the EST $E(T_s)$, where $E(\cdot)$ denotes the expected value function.*

C. Modeling

One immediate question about Problem Statement 6 is whether we can obtain the EST without referring to or being limited to a particular motion plan. To address this dependency, we first characterize the motion plans based on their outcomes before modeling the EST.

1. Renewal Process and Characterizing Planners

Periodically, the robot trajectory planner outputs a motion plan and the robot executes the plan during the searching process. The system is naturally a repetitive scanning process during which the robot enters and leaves the circle until Condition 1 is satisfied. During the probabilistic nature, the process is a random process with a stopping time.

The repetitiveness in coverage can be modeled as a renewal process [66]. Letting inter-arrival time X_m denote the time between the $(m-1)$st and mth event, a renewal process can be viewed as a generalization of Poisson process with $\{X_m, m = 1, 2, ...\}$ being independently and identically distributed (*i.i.d.*) random variables. Note that the distributions of X_m can be any type but must be *i.i.d.*

To facilitate our analysis, we partition the continuous trajectory of the robot into many repetitive *i.i.d. tours*.

Definition 1. *A* tour *starts or restarts at the moment when the robot first enters the target circle during the single independent scanning.*

The dashed line in Fig. 1 illustrates a tour. Tours may be quite different depending upon the planner. For example, tour length varies each time if the robot follows a random walk. As another example, a deterministic planner usually has a fixed tour trajectory. An event happens when each tour starts. The tour length are

the inter-arrival time of events.

Based on Condition 2, we know that the robot does not accumulate the knowledge regarding the target location from one tour to another tour because no signal has been perceived before the moment the searching mission is accomplished. Note that a tour can contain multiple intersections with the circle in a general case. Tour definition is a way of partition the trajectory and does not have to be synchronized exactly with how the robot planner works. Hence each tour length is *i.i.d.* as long as the motion planner does not change its planning algorithm. Therefore, the searching process as a renewal process.

During each tour, the robot spends some time inside the circle and some time outside the circle, which are defined as τ_{IN} and τ_{OUT}, respectively. Hence $\tau_{\text{IN}} + \tau_{\text{OUT}}$ is the overall duration for the tour.

The characteristics of renewal process allows us to compute its limiting properties by focusing on individual periods. For example, we immediately know that the long run probability of being inside circle is $\tau_{\text{IN}}/(\tau_{\text{IN}} + \tau_{\text{OUT}})$ from the property of renewal processes.

If the trajectory only intersects the circle once in each period, when a tour begins, the robot first spends τ_{IN} inside the circle followed τ_{OUT} outside the circle. This yields an alternating renewal process. If there are many intersections between the circle and the tour, then τ_{IN} and τ_{OUT} are not two continuous time segments but a summation of many fragmented segments which can be computed by conditioning on the number of intersections. For simplicity of analysis and due to the limited space, we will focus on the single intersection case in the rest of paper. Readers can follow our methods to analyze multiple intersection cases.

2. Modeling the EST

Without loss of generality, we assume the robot starts the searching process from the origin which is on the boundaries of the searching space. It takes some time to reach the circle where the first tour starts. Define the time as delay D. From Conditions 1 and 2, we know that the robot cannot find the target in D. The searching process is a delayed alternating renewal process. Define T_s^c as the time to find the target after the robot enters the repetitive tours. Hence, the EST is

$$E(T_s) = E(D) + E(T_s^c). \tag{3.1}$$

Define N as the number of signal transmissions during τ_{IN} in a tour. Since the arrival process of the signal transmission is Poisson, N conforms to a Poisson distribution,

$$P(N = k) = \frac{e^{-\lambda\tau_{\text{IN}}}(\lambda\tau_{\text{IN}})^k}{k!}, \quad k = 0, 1, 2, ..., \infty. \tag{3.2}$$

We know that event $N > 0$ means that at least one signal transmission happens during τ_{IN}. This means the target is found. Therefore, we can compute $E(T_s^c)$ by conditioning on N,

$$E(T_s^c) = E(T_s^c|N > 0)P(N > 0) + E(T_s^c|N = 0)P(N = 0), \tag{3.3}$$

where $P(N > 0) = 1 - e^{-\lambda\tau_{\text{IN}}}$ and $P(N = 0) = e^{-\lambda\tau_{\text{IN}}}$ according to (3.2). Note that $N < 0$ is not possible due to the fact that N is a counting variable.

Now let us compute $E(T_s^c|N > 0)$. Since event $N > 0$ is equivalent to event $T_s^c \leq \tau_{\text{IN}}$, we have

$$E(T_s^c|N > 0) = E(T_s^c|T_s^c \leq \tau_{\text{IN}}) = \frac{1}{\lambda} - \frac{\tau_{\text{IN}}e^{-\lambda\tau_{\text{IN}}}}{1 - e^{-\lambda\tau_{\text{IN}}}} \tag{3.4}$$

because the conditional distribution $T_s^c|T_s^c \leq \tau_{\text{IN}}$ is a truncated exponential distribu-

tion [67, 66]. It is worth noting that (3.4) is valid only if $\tau_{\text{IN}} > 0$, which is guaranteed according to Definition 1. On the other hand, we know

$$E(T_s^c|N=0) = \tau_{\text{IN}} + \tau_{\text{OUT}} + E(T_s^c) \tag{3.5}$$

because the robot cannot find the inactive target in the current tour and has to start all over again in next tour.

Plugging (3.4) and (3.5) into (3.3) and (3.1), we have the following result.

Theorem 1. *Given the expected time $E(D)$ for the robot to reach the circle, the Poisson arrival rate λ of target signals, the traveling time τ_{IN} inside the circle, and the traveling time τ_{OUT} outside the circle, the EST of the target is*

$$E(T_s|\tau_{IN}, \tau_{OUT}) = E(D) + \frac{1}{\lambda} + \tau_{OUT}\frac{e^{-\lambda\tau_{IN}}}{1-e^{-\lambda\tau_{IN}}}. \tag{3.6}$$

This is a conditional expectation because τ_{IN} and τ_{OUT} are often random variables. Since $E(D)$ and λ are independent of τ_{IN} and τ_{OUT}, the unconditional EST can be obtained as,

$$E(T_s) = E(D) + \frac{1}{\lambda} + E\big(\tau_{OUT}\frac{e^{-\lambda\tau_{IN}}}{1-e^{-\lambda\tau_{IN}}}\big). \tag{3.7}$$

The formulation of EST given in Theorem 1 has a surprisingly succinct format revealing the relationship between the EST and the corresponding variables. To further explain (7.10), let us consider the following extreme cases:

Case 1: When $\lambda \to \infty$, it means that the target continuously transmits signals. The searching time becomes the time that it takes for the robot to enter the circle. The problem degenerates to the traditional coverage problem where $E(T_s) = E(D)$.

Case 2: When $\tau_{\text{OUT}} = 0$, it means that the signal emitted by the target is so powerful that the circle defined by d_s can cover the entire searching space. In this case, $E(D) = 0$. Hence $E(T_s) = 1/\lambda$, the mean inter-arrival time of signals. The

robot finds the target as soon as the target emits a signal.

Remark 1. *It is worth noting that (7.10) does not depend on a particular motion plan or the shape/dimension of the searching space, which makes it widely applicable in practice. Indeed, the EST can be also applied to analyze searching tasks carried by humans. In many cases, the signal transmission rate λ is known; $E(D)$ can be estimated based on observations; τ_{IN} can be estimated based on d_s and v; and τ_{OUT} can be measured based on observations that how often a robot would revisit a region with the same size of the circle. Based on the known information and online measurements, we can even predict the EST for an ongoing searching process regardless its motion plan, which is of great importance in applications where the searching time literally means life or death.*

D. Analysis of Common Searching Strategies

Theorem 1 can be used to analyze the searching performance under different robot motion plans and configurations. We begin with demonstrating how Theorem 1 can reveal the difference between two motion plans from common coverage methods: the slap method and the random walk.

1. The Slap Method Versus The Random Walk

a. The Slap Method (SM)

Also known as the trapezoidal decomposition [17, 18] in robotics research, SM sequentially scans the entire searching space back and forth. Fig. 2a illustrates the robot motion plan for the square case. The plan is a set of y-axis parallel lines (appears to be vertical lines in Fig. 2a) that cover the entire searching space. The vertical lines are inter-connected using the boundaries of the searching space to formulate

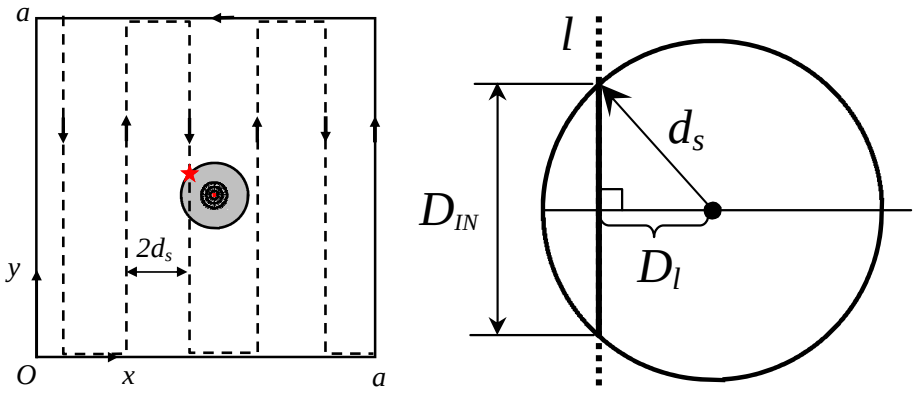


Fig. 2.: (a) A sample motion plan for the slap method. (b) An illustration of how a tour (line l) intersects the circle of the target.

a complete tour. To guarantee an intersection between the circle and the tour, the distance between adjacent vertical lines is set to be $2d_s$.

The red "$\star$" in Fig. 2a indicates the starting point of the tour. Since tours are exactly the same in the slap method, the subsequent tours start exactly at the same location. The overall tour length is approximately $a^2/(2d_s)$. Given the robot speed v, it takes

$$\tau_{\text{IN}} + \tau_{\text{OUT}} \approx \frac{a^2}{2vd_s} \tag{3.8}$$

time for the robot to finish the tour. Since the target could be anywhere in the searching space with equal probabilities, we obtain

$$E(D) \approx (\tau_{\text{IN}} + \tau_{\text{OUT}})/2 = \frac{a^2}{4vd_s}. \tag{3.9}$$

To use the result in Theorem 1, the remaining undetermined variable is τ_{IN}. Let us define D_{IN} as the distance traveled inside the circle. D_{IN} is the length of intersection when the line intersects the circle as illustrated in Fig. 2b. Here we ignore the boundary effect where the circle is not a full circle because $a \gg d_s$. Line

l in Fig. 2b is a part of the tour. When l intersects the circle, we define D_l as the distance between the center of the circle and the line. Since the target is uniformly distributed in the 2D space, $D_l \sim U(0, d_s)$ is uniformly distributed. From Fig. 2b, we know

$$\tau_{\text{IN}} = \frac{D_{\text{IN}}}{v} = \frac{2\sqrt{d_s^2 - D_l^2}}{v}. \tag{3.10}$$

Plugging (3.8), (3.9) and (3.10) into (7.10) and conditioning on D_l, we have

$$E(T_s|D_l) \approx \frac{a^2}{4vd_s} + \frac{1}{\lambda} + (\frac{a^2}{2vd_s} - \frac{2\sqrt{d_s^2 - D_l^2}}{v})\phi(\lambda, D_l), \tag{3.11}$$

where

$$\phi(\lambda, D_l) = \frac{1}{e^{\frac{2\lambda\sqrt{d_s^2 - D_l^2}}{v}} - 1}. \tag{3.12}$$

Since $a \gg d_s$, $\tau_{\text{OUT}} \gg \tau_{\text{IN}}$, and $\frac{2\sqrt{d_s^2 - D_l^2}}{v}$ is negligible if compared with $\frac{a^2}{2vd_s}$, we have

$$E(T_s|D_l) \approx \frac{a^2}{4vd_s} + \frac{1}{\lambda} + \frac{a^2}{2vd_s}\phi(\lambda, D_l). \tag{3.13}$$

Hence we have the EST for the slap method,

$$\begin{aligned} E(T_s) &= \int_{\delta=0}^{d_s} E(T_s|D_l = \delta)\frac{1}{d_s}d\delta \\ &\approx \frac{a^2}{4vd_s} + \frac{1}{\lambda} + \frac{a^2}{2vd_s}g(d_s, \lambda) \end{aligned} \tag{3.14}$$

where

$$g(d_s, \lambda) = E(\phi(\lambda, D_l)) = \int_{\delta=0}^{d_s} \frac{1}{d_s}\phi(\lambda, \delta)d\delta. \tag{3.15}$$

Let $\delta = d_s \cos\theta$, we can transform (3.15) into

$$g(d_s, \lambda) = \int_{\theta=0}^{\pi/2} \frac{1}{e^{\frac{2\lambda d_s \sin\theta}{v}} - 1} \sin\theta d\theta. \tag{3.16}$$

When λ and d_s/v are very small, (3.15) can be further simplified,

$$g(d_s, \lambda) \approx \frac{\pi v}{4\lambda d_s} - 1. \tag{3.17}$$

Remark 2. *Eq. (3.14) suggests that a fast robot (large v) with great sensing distance d_s reduces the EST. This conclusion agrees with our intuition that mobility and sensing are the key elements in searching. However, it also takes a target's cooperation to further reduce the EST. When the robot reaches its speed and sensing limit, the only way to reduce EST is to increase λ. Of course, the target usually has energy constraints and cannot arbitrarily increase λ.*

The analysis assumes the distance between vertical lines is $2d_s$, which ensures there is only one intersection between the circle and the tour. When a smaller spacing is used, the overall tour length increases and so does τ_{IN}. The analysis is slightly more complicated because it needs to be conditioned on the number of intersections between the tour and the circle. The results actually share a similar format with (3.13) and the same asymptotic properties with respect to a, v, and λ. Since our focus is to compare the asymptotic behavior of the slap method with that of the random walk, we omit the analysis here.

In the above calculation, we ignore the boundary effect on the final result. When the circle is located at the boundary of the square, distance D_{IN} cannot be computed using (3.10). Since the target has to be located within d_s distance of the boundary for the boundary case, the probability that such event happens is less than $\frac{4d_s a}{a^2} = \frac{4d_s}{a} \ll 1$, since $d_s \ll a$. Hence its impact to the final EST is ignorable because D_{IN} for such case is not significantly different from that of the non-boundary case. Therefore, we will ignore boundary effect in the rest of the paper.

b. Random Walk

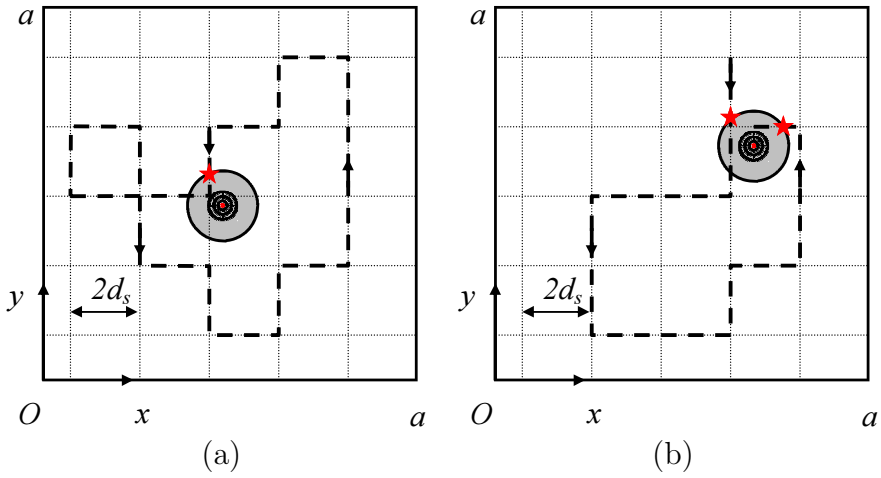


Fig. 3.: An illustration of robot motion plan based on a 2D lattice-based random walk. Two types of tours are illustrated here (a) a closed tour and (b) an unclosed tour.

Another popular motion plan is to employ a 2D random walk to traverse the searching space. As illustrated in Fig. 3, we partition the entire searching space using a 2D finite lattice with a spacing of $2d_s$ in each dimension. Denoting N_s as the number of lattice nodes, we have $N_s = \frac{a^2}{4d_s^2}$ nodes. Finer lattice is possible but usually associated with higher energy cost because the robot has to make a lot more turns.

The robot always moves from one lattice node to its neighboring node with equal probabilities. The robot does not cross the boundaries. According to [68], this is a finite 2D lattice with reflective boundaries. Recall that a tour starts at the moment the robot enters the circle. Since the robot might not enter the circle at the exactly same location in different tours, each tour is not necessarily a completely closed curve as that in the slap method case. The closed curve tour in Fig. 3(a) only happens with a probability of 1/4 whereas the unclosed tour in Fig. 3(b) has a probability of 3/4.

To compute the EST given in (7.10), we need to compute $E(D)$. Recall that the robot always starts at origin. Given the location of target (X_t, Y_t), computing the mean time that it takes the robot to follow the random walk to reach a particular location (X_t, Y_t) is the mean first passage time (MFPT)[69, 70] problem in stochastic modeling. The exact solution to this problem is expressed in the format of pseudo Green functions and cannot be explicitly analyzed. Since $a \gg d_s$, there are a large number of nodes $\frac{a^2}{4d_s^2}$ in the 2D lattice and each robot move takes $\frac{2d_s}{v}$ time. Hence we can apply the recent results of MFPT using its asymptotic format in [71],

$$E(D|X_t, Y_t) \approx \frac{a^2}{2vd_s}\left(\alpha_0 + \alpha_1 \ln\sqrt{X_t^2 + Y_t^2}\right), \tag{3.18}$$

where α_0 and α_1 are constants and can be determined by Monte Carlo methods. According to [71], α_0 and α_1 strikingly do not depend on lattice size but local transitional properties. Hence,

$$E(D) = \int_0^a \int_0^a E(D|X_t = x, Y_t = y)\frac{1}{a^2}dxdy, \tag{3.19}$$

$$\approx \frac{\alpha_0 a^2}{2vd_s} + \frac{\alpha_1}{2vd_s}\int_0^a \int_0^a \ln\sqrt{x^2 + y^2}dxdy. \tag{3.20}$$

Since

$$\int_0^a \int_0^a \ln\sqrt{x^2 + y^2}dxdy = a^2 \ln a + \frac{\pi + 2\ln 2 - 6}{4}a^2$$

we have

$$E(D) \approx \frac{a^2}{2vd_s}\left(\alpha_0 + \alpha_1 \ln a + \alpha_1\frac{\pi + 2\ln 2 - 6}{4}\right). \tag{3.21}$$

The remaining unknown term in (7.10) is $E(\tau_{\text{OUT}}\frac{e^{-\lambda\tau_{\text{IN}}}}{1-e^{-\lambda\tau_{\text{IN}}}})$. Given the robot speed v, τ_{IN} is uniquely determined by the distance in the circle D_{IN}, which is independent of the overall trajectory. Also $E(\tau_{\text{OUT}}) \approx E(\tau_{\text{OUT}} + \tau_{\text{IN}})$ given that $a \gg d_s$. Hence,

$$E(\tau_{\text{OUT}}\frac{e^{-\lambda\tau_{\text{IN}}}}{1 - e^{-\lambda\tau_{\text{IN}}}}) \approx E(\tau_{\text{OUT}} + \tau_{\text{IN}})E(\frac{e^{-\lambda\tau_{\text{IN}}}}{1 - e^{-\lambda\tau_{\text{IN}}}}). \tag{3.22}$$

Since the 2D lattice-based random walk is undirected and symmetric in transitional probability, from [72], we know that the stationary probability of visiting node i is, $\pi_i = \frac{n_d(i)}{2m_e}$ where $n_d(i)$ is the degrees of freedom of node i and m_e is the total number of edges in the lattice. For a non-boundary node, $n_d(i) = 4$ and $m_e \approx 4N_s$. Hence

$$\pi_i = \frac{1}{N_s} = \frac{4d_s^2}{a^2},$$

Without loss of generality, we assume node i is the node closest to the target. This means that π_i is the long run probability that the robot intercepts the circle. Let (x_i, y_i) be the node i's position. Given node i is the nearest node to the target and recall target location is (X_t, Y_t), then we know $|x_i - X_t| \leq d_s$ and $|y_i - Y_t| \leq d_s$ based on the lattice definition. Therefore, the conditional probability the robot is inside the circle given node i is visited is the area ratio between the circle and the square with a side length of $2d_s$: $p_{c|i} = \frac{\pi d_s^2}{4d_s^2} = \frac{\pi}{4}$ because the target is uniformly distributed in the searching space. Hence the unconditional long run probability that robot stay inside the circle is $p_c = p_{c|i}\pi_i = \frac{\pi d_s^2}{a^2}$.

On the other hand, we can obtain the same long run probability using the Renewal Reward theorem [66] if we view τ_{IN} as the reward function for each period,

$$\frac{E(\tau_{\text{IN}})}{E(\tau_{\text{OUT}} + \tau_{\text{IN}})} = p_c = \frac{\pi d_s^2}{a^2}. \tag{3.23}$$

Plugging (3.23) into (3.22), we have

$$E\left(\tau_{\text{OUT}} \frac{e^{-\lambda\tau_{\text{IN}}}}{1 - e^{-\lambda\tau_{\text{IN}}}}\right) \approx \frac{a^2}{\pi d_s^2} E(\tau_{\text{IN}}) E\left(\frac{e^{-\lambda\tau_{\text{IN}}}}{1 - e^{-\lambda\tau_{\text{IN}}}}\right). \tag{3.24}$$

Now, we focus on the computation of τ_{IN}. Since the lattice has a spacing of $2d_s$, two scenarios exist when the tour on the lattice intersects the circle: i) the next lattice point on the tour is inside the circle and ii) the next lattice point on the tour is

outside the circle as illustrated in Fig. 4. Let us define events that scenarios i) and ii) happen as events E_i and E_o, respectively. Since the circle center is uniformly located in the searching space,

$$P(E_i) = \frac{\pi d_s^2}{4d_s^2} = \frac{\pi}{4} = 1 - P(E_o). \tag{3.25}$$

When event E_o happens, we know that the robot trajectory intersects the circle as a straight line as shown in Fig. 4(a). Hence we have

$$\tau_{\text{IN}}|E_o = \frac{D_{\text{IN}}}{v}, \tag{3.26}$$

where D_{IN} is defined in (3.10) and the right side of | is the condition for the equality to be true. This is a notation convention widely used in stochastic modeling [66].

$$E(\tau_{\text{IN}}|E_o) = \frac{\pi d_s}{2v}, \text{ and} \tag{3.27}$$

$$E\left(\frac{e^{-\lambda\tau_{\text{IN}}}}{1-e^{-\lambda\tau_{\text{IN}}}}|E_o\right) = g(d_s, \lambda). \tag{3.28}$$

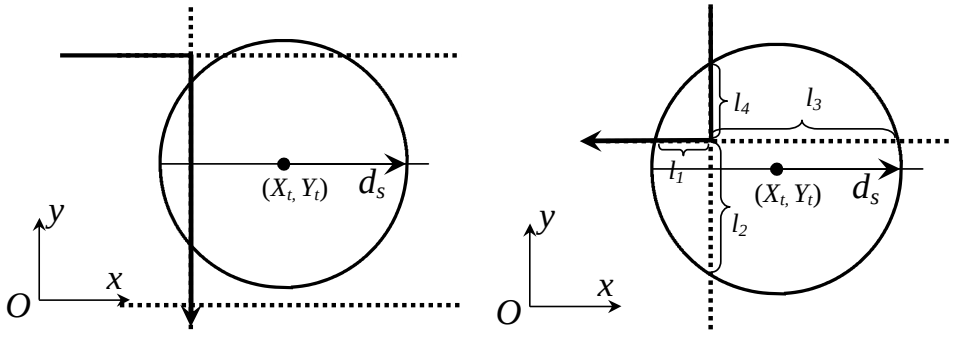


Fig. 4.: An illustration of how the robot trajectory in solid line intersects the circle. (a) Scenario i) when the nearest lattice point on tour is located outside the circle. (b) Scenario ii) when the nearest lattice point on tour is located inside the circle. The dashed line in the figure is part of the lattice.

When event E_i happens, one lattice point is inside the circle. As illustrated in Fig. 4(b), the lattice point inside the circle partitions the lattice edges inside the circle into four parts: l_1, l_2, l_3, and l_4. When a robot trajectory intersects the circle, the part of the trajectory inside the circle can be divided into two segments, which are defined as L' and L'', respectively. L' refers to the segment that the robot takes to arrive at the lattice node and L'' refers to the segment that the robot takes to leave the circle. Hence

$$\tau_{\text{IN}}|E_i = \frac{L' + L''}{v}.$$

Since L' and L'' have equal probabilities to take l_1, l_2, l_3, and l_4, there is a total of $2^4 = 16$ combinations. Conditioning on the 16 (L', L'') combinations and the circle center location (X_t, Y_t), we obtain the same results as shown in (3.27) and (3.28). Combining those results for the E_i and E_o events by conditioning on them, we have the unconditional expected values $E(\tau_{\text{IN}})$ and $E\left(\frac{e^{-\lambda\tau_{\text{IN}}}}{1-e^{-\lambda\tau_{\text{IN}}}}\right)$ coincidentally sharing the same formats as in (3.27) and (3.28), respectively. Plugging the two expectations, (3.21) and (3.24) into (7.10), we can obtain the EST for the random walk case,

$$\begin{aligned} E(T_s) \approx & \frac{a^2}{2vd_s}\left(\alpha_0 + \alpha_1 \ln a + \alpha_1 \frac{\pi + 2\ln 2 - 6}{4}\right) \\ & + \frac{1}{\lambda} + \frac{a^2}{2d_s v} g(d_s, \lambda). \end{aligned} \tag{3.29}$$

Comparing (3.29) to (3.14), we have the follow conclusion,

Corollary 1. *With the same square field side length a, the sensing range d_s, and the signal transmission rate λ, the EST value of the slap method is asymptotically smaller than that of the random walk when $a \to \infty$.*

Proof. It is straightforward because $E(T_s) = \Theta(a^2)$ for the slap method from (3.14) while $E(T_s) = \Theta(a^2 \ln a)$ for the random walk according to (3.29). □

So far our analysis are limited to cases in open 2D Euclidean space. It is possible to extend the analysis to more general cases if certain conditions are satisfied:

Remark 3. *For 2D space with obstacles: The 2D space must be fully connected. The trajectory generated by motion planner should ensure the uniform coverage of the searching space for the slap method. For the random walk, the lattice may not be regular shaped as in the paper. To ensure the long run uniform coverage, the transitional probability at each node should be symmetric. These conditions are common in many search tasks. When the conditions are satisfied, then both Theorem 1 and Corollary 1 hold.*

Remark 4. *For 3D cases: The target circle becomes the target ball. Theorem 1 can be directly applied to 3D cases but the analysis of the slap method and the random walk needs to be modified. For example, the searching space size should be measured by its volume instead of its area. The results for the 3D slap method and the 3D random walk are different but the overall approach should be similar. The extension should be straightforward. Since most searching tasks are 2D, we focus on 2D cases in this paper.*

Note that although slower than that of the slap method, random walk-based methods have memoryless property and do not require complex coordination and communication which results in more and cheaper robots. A comparison between cheap robot team and a single expensive robot is the next goal.

2. Analysis of Different Robot Configurations

Theorem 1 can also be used to analyze cases under different robot configurations. Here we compare two configurations.

A low-cost robot team (LCRT) case: We have n identically-configured low-cost robots. To coordinate the searching, we partition the searching space into n sub square fields with an area of a^2/n each and allocate one robot for each sub square field.

A single expensive robot (ASER) case: We have an expensive robot equipped with a very capable sensor that has a sensing area equal to the combination of those of the n low-cost robots. If each of the low-cost robot has a sensing range of d_s, then the area of the combined sensing region for n robots is $n\pi d_s^2$ provided that there is no overlap of sensing region between any two robots. Therefore, the sensing distance for the expensive robot is set to $d_s' = \sqrt{n}d_s$ to ensure ASER has the area of sensing coverage no less than that of LCRT at any given time.

We are now ready to compare these two robot configurations. Since the slap method is asymptotically faster than the random walk, we build on the slap method results in (3.29). For the LCRT, only one robot actually has the target in its sub field. Hence, the rest of $n-1$ robots are irrelevant in the searching process. Comparing with the original EST in (3.29), we just need to replace a with $\frac{a}{\sqrt{n}}$. Defining the searching time for the LCRT as T_s', we have

$$E(T_s') \approx \frac{a^2}{4vnd_s} + \frac{1}{\lambda} + \frac{a^2}{2vnd_s} g(d_s, \lambda). \tag{3.30}$$

Defining the searching time for the ASER as T_s'', we have

$$E(T_s'') \approx \frac{a^2}{4v\sqrt{n}d_s} + \frac{1}{\lambda} + \frac{a^2}{2v\sqrt{n}d_s} g(\sqrt{n}d_s, \lambda). \tag{3.31}$$

From (3.16), it is straightforward to see that

$$g(\sqrt{n}d_s, \lambda) \to 0 \quad \text{as} \quad n \to \infty. \tag{3.32}$$

Therefore, we have the following conclusion,

Corollary 2. *When traveling at the same velocity v, the low-cost robot team can find the target asymptotically faster than the single expensive robot does when n increases, if $1/\lambda$ is not the dominating factor in the EST.*

Proof. From (3.31) and (3.32), we know $E(T_s'') = \Theta(\frac{1}{\sqrt{n}} + \frac{1}{\lambda})$. From (3.30), we know $E(T_s') = \Theta(\frac{1}{n} + \frac{1}{\lambda})$. Hence the conclusion follows. □

If we use random walk in comparing LCRT and ASER, Corollary 2 still holds. We skip the analysis because the EST of random walk shares a similar format of that of the slap method and the proof is similar as well. Note that the conclusion is established based on the circular sensing region, the result would be different if a different sensing topology is used.

Remark 5. *This analysis also shows that if there are cost functions associated with the number of robots, different sensor options, or different velocity options available, we can use the EST results as an objective function to optimize the robot configuration for the task.*

The result in Corollary 2 is actually not very intuitive at the first sight. We have not expected such a significant difference in the comparison. This conclusion is rather interesting because it shows that an expensive robot with superior sensing capability is not as good as a large number of low-cost robots with less capable sensors when searching for targets that intermittently transmits short duration signals.

A more intuitive explanation of the result in Corollary 2 is that the ability of searching is directly determined by spatiotemporal sensory coverage (SSC) of the moving robot.

Definition 2. *Given the robot travels at a speed of v with a sensory radius of d_s, SSC is measured by the area of the swept region per unit time a_{SSC}.*

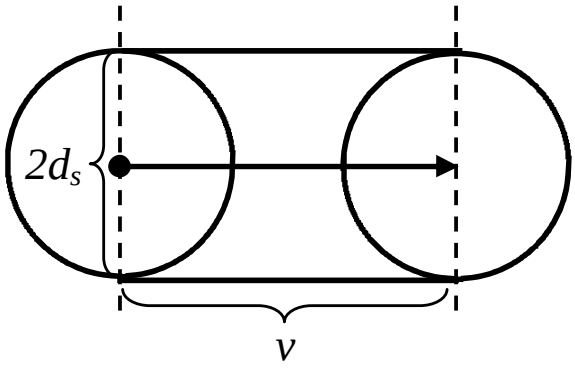


Fig. 5.: Linear spatio-temporal sensory coverage during a unit time.

For example,

$$a_{\mathrm{SSC}} = \pi d_s^2 + 2d_s v \tag{3.33}$$

for a robot travels along a line (see Fig. 5). For a LCRT with the slap method-based trajectories, SSC is the summation of n non-overlapping robots: $\pi n d_s^2 + 2n d_s v$. However, SSC for the corresponding ASER becomes $\pi n d_s^2 + 2\sqrt{n} d_s v$, which is less than that of LCRT. This is consistent with the result in Corollary 2.

When the robot does not travel along a line during the unitary time, its SSC becomes inevitably smaller than that in (3.33). For example, during the random walk, the robot makes a lot more turns than that of the slap method, which decreases SSC over the course of searching and results in a longer EST as shown in Corollary 1.

The SSC analysis can also be used to predict the behavior of two different random walk setups for the robot team. In the aforementioned LCRT setup, we partition the searching space into n equally-sized sub areas, which is defined as LCRT-P-RW with P-RW stands for the partition-based random walk. As we know, the EST of LCRT-P-RW is longer than that of LCRT with the slap method, which is abbreviated as LCRT-SM. Another possible way of coordinating n random walk based robots is that we do not partition the searching space but release all n robots at the same origin and

let them compete. The searching stops when any robot finds the target. We name the method as LCRT-C-RW where C-RW stands for the competition-based random walk. For the LCRT-C-RW process, we can compute EST using order statistic [73] since the searching time is the smallest of n *i.i.d.* random variables. However, although the resulting EST can be obtained either analytically or computationally [74], it cannot be expressed in a simple closed form format for comparison study. We will not elaborate the process. However, the introduction of SSC can quickly compare LCRT-C-RW to LCRT-P-RW:

Lemma 1. *The expected SSC of the LCRT-P-RW is always bigger than that of the LCRT-C-RW.*

Proof. In the LCRT-C-RW process, there is a non zero probability p_o that the sensing region between robots overlaps, which reduces its effective SSC whereas the LCRT-P-RW does not allow the overlap to happen. Define $a_{\text{SSC-P}}$ and $a_{\text{SSC-C}}$ as the SSC for LCRT-P-RW and LCRT-C-RW, respectively. Denote ϵ_a as the reduction of SSC caused by the overlap. Conditioning on the overlapping event E', we have

$$\begin{aligned} E(a_{\text{SSC-C}}) &= E(a_{\text{SSC-C}}|\overline{E'})(1-p_o) + E(a_{\text{SSC-C}}|E')p_o \\ &= E(a_{\text{SSC-P}})(1-p_o) + (E(a_{\text{SSC-P}}) - E(\epsilon_a))p_o \\ &< E(a_{\text{SSC-P}}), \end{aligned} \tag{3.34}$$

since ϵ_a is positive. □

Remark 6. *This result indicates that the searching efficiency of LCRT-C-RW is not as good as LCRT-P-RW. Actually, the combination of EST analysis and SSC analysis can help us to identify the most efficient coordination strategy with respect to different searching space partition, robot allocation, or robot formation methods for the robot team. Two interesting observations can be summarized here,*

- *For the robot trajectory, try to search in straight lines to maximize the SSC. An optimal control problem can be formulated if searching directions are forced to change due to searching space constraints.*
- *For the robot team, try to avoid overlapping among searching regions of robots. Partitioning the searching space is a good practice. We could also use a particular formation to achieve this.*

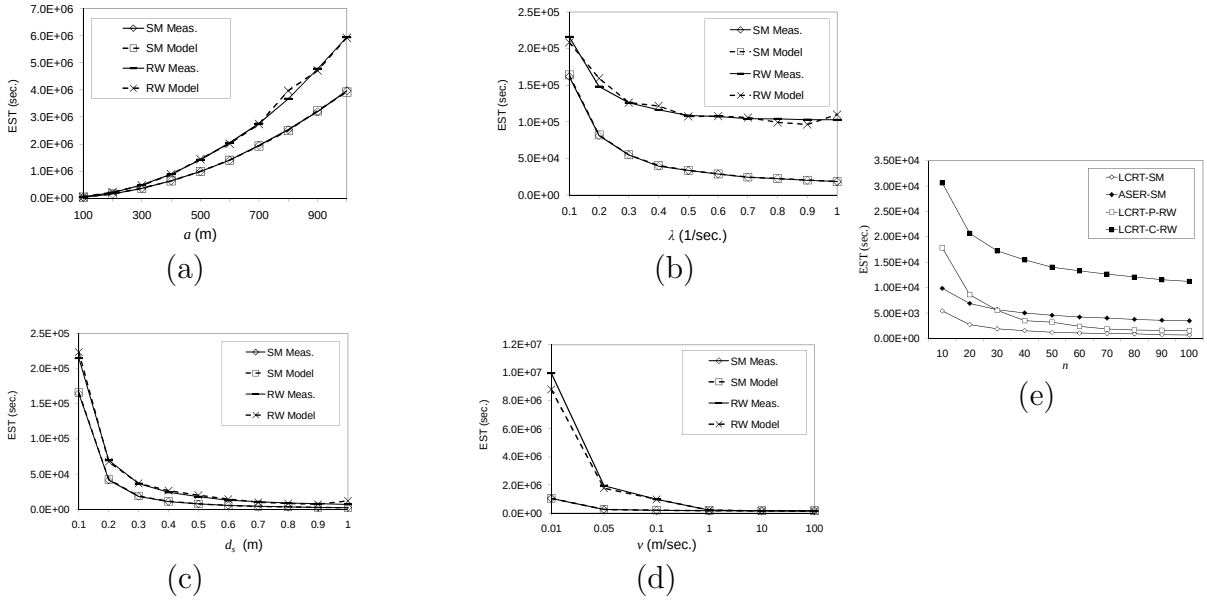


Fig. 6.: Simulation results in (a), (b), (c), and (d) for validating Theorem 1 with respect to a, λ, d_s, and v, respectively. SM stands for the slap method. RW stands for the random walk. Model means the model prediction of the EST. Meas. means the measured mean searching time. (e) Simulation results for comparing four different robot configurations. Recall that LCRT stands for the low cost robot team and ASER stands for a single expensive robot.

E. Experiments

We test our results using Monte Carlo simulation and physical experiments.

1. Simulation

Simulation allows us to examine theoretical results across complete parameter ranges without the limitation of hardware conditions. The simulation results are illustrated in Figs. 6 and 6e. Each data point in both figures is an average of 10,000 independent trials. At the beginning of each trial, we reset the robot position to be at $(0, 0)$ and generate the target location according to a 2D uniform distribution. We then run the robot according to the selected motion plan and finish the trial as soon as the target is found.

a. Validating Theorem 1 and Corollary 1

We test Theorem 1 using both the slap method and the random walk because Theorem 1 is supposed to be independent of motion plans. The simulation is set up with different a, λ, d_s, and v settings in Table II. In each setting, we collect both the model predicted EST and the measured mean searching time. The measured mean searching time is the average of the searching time over the 10k trials (the "Meas." values in Fig. 6)(a-d). The model predicted ESTs, which are the "Model" values in Fig. 6, refer to the predicted ESTs according to the measured D, λ, τ_{IN}, and τ_{OUT} values in the experiment. In other words, we record their values and average them over the 10000 trials to obtain the estimation of $1/\lambda$, $E(D)$, and $E(\tau_{\text{OUT}}\frac{e^{-\lambda\tau_{\text{IN}}}}{1-e^{-\lambda\tau_{\text{IN}}}})$. We then feed them into (7.10) to obtain the model prediction of the EST.

As illustrated in Fig. 6(a-d), the model prediction is fairly consistent with the measured mean searching time under all settings. The curve trends with respect to

Table II.: Parameter settings for results in simulation and physical experiments.

Figure	a (m)	λ (1/sec.)	d_s (m)	v (m/s)
Fig. 6(a)	100-1000	0.1	1.0	1.0
Fig. 6(b)	200	0.1-1.0	1.0	1.0
Fig. 6(c)	200	0.1	1-10	1.0
Fig. 6(d)	200	0.1	1.0	0.01-100
Fig. 6(e)	200	0.1	1.0	1.0
Figs. 9&10	100	0.1	0.2	0.20

a, λ, d_s and v in Fig. 6(a-d) are also consistent with our analysis in (3.14) and (3.29). The EST increases as the field side length a increases. The EST decreases as λ, d_s, and v increase. All figures show that the random walk is slower than the slap method. In particular, Fig. 6(a) is consistent with the asymptotic difference in Corollary 1.

b. Validating Corollary 2 and Lemma 1

We have also implemented different LCRT and ASER robot configurations. Again, the parameter settings are in the Table II. The measured ESTs for both the configurations are shown in Fig. 6(e). It is clear that the EST for the LCRT-SM is always much smaller than that of the ASER-SM. This is consistent with Corollary 2. Also, the EST for the LCRT-P-RW is always much smaller than that of the LCRT-C-RW, which is consistent with Lemma 1. What is interesting is the fact that the EST of the LCRT-P-RW is initially bigger than that of the ASER-SM but become smaller than than that of the ASER-SM as n increases. Curves in the figure also show the general trend that the EST decreases as the n increases. This is consistent with our analysis. Also, as n becomes large, the curve levels at a non-zero value. This indicates that the signal transmission rate dominates the searching time in this case. Hence it is not desirable to arbitrarily increase n because the marginal benefit would decrease. Actually, with an appropriate cost function, readers can extend the analysis to find an optimal number of robots for a given searching task. This would be very meaningful

in resource-constrained searching process.

2. Physical Experiments

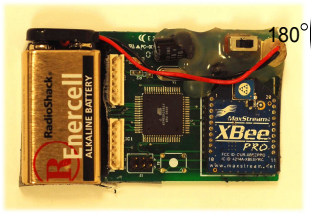
a. Hardware and Experiment Setup

In the physical experiment, we use a mobile robot to search for a buried intermittent radio beacon (see Fig. 13). This process emulates the search of a black box after a plane crash. The robot is custom made in our lab. The robot measures $50 \times 47 \times 50$ cm^3 in size. The robot has two front drive wheels and one rear cast wheel and uses a typical differential driving structure. During the experiment, the robot traveling speed is set at 20 cm/second. The maximum robot operation time between recharging is 4 hours. Buried under the floor, the target is a Zigbee radio source which is an XBee Pro node (Fig. 13(b)) with a chip antenna from Digi International Inc. (formerly known as MaxStream Inc.) We program the radio to allow it to transmit one packet every ten seconds ($\lambda = 0.1$).

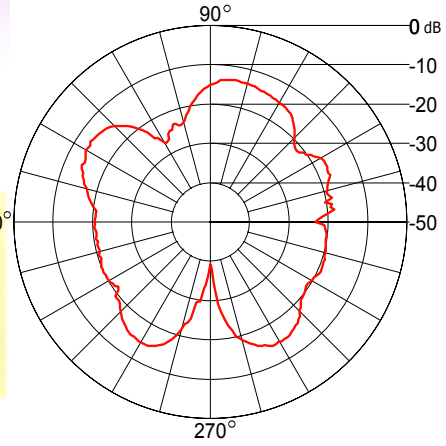
One notable difference between the simulation and the physical experiment is the shape of the sensing region. Although the radio antennas are omnidirectional, the radiation pattern (see Fig. 13(c)) is irregular. The shape of the radiation pattern is the shape of the sensing region. Since $a = 100$ meters in our settings, we decrease antenna sensitivity to ensure $a \gg d_s$. After calibration and adjusting antenna sensitivity, we approximate the reception region with a circle ($d_s = 0.2$ meters). The physical experiment parameters are summarized in Table II.

Since the robot has a slow speed and a small sensing region, recall that the field size is 10^4 m^2, it is estimated that it takes the robot at least 10^6 seconds to find the target. However, the robot only has 4-hour continuous operation time due to our hardware limitations. When the robot travels outside the sensing region, it cannot

(a)



(b)



(c)

Fig. 7.: (a) The mobile robot used in the experiment. (b) An Xbee Pro RF module used as the signal source, which is the target. (c) The RF radiation pattern of the Xbee Pro. This plot is from the product specification sheet of Digi International Inc. This is also the shape of the real sensing region.

receive signals and does not provide information for the searching process. Therefore, we setup the experiment in an "accelerated" configuration. In this configuration, we do not drive the robot according to the motion plan when the robot is far away from the signal source. Instead, we fast forward the time to the next moment that the robot is close to the signal source. Fig. 8 illustrates the setup. The shaded area in Fig. 8(a) is the region, which is a square with a side length of $4d_s$. The region fully encloses the sensing region as shown in the figure. The robot obtains its location from an overhead camera as illustrated in Fig. 8(b). The overhead camera is an Arecont Vision 3100 networked video camera. The camera recognizes color markers on the top of the robot and computes the robot position and orientation subsequently.

Configured at a 640 × 480-pixel resolution, it is capable of reaching an accuracy of ±2.0 cm with 16 Hz frame rate.

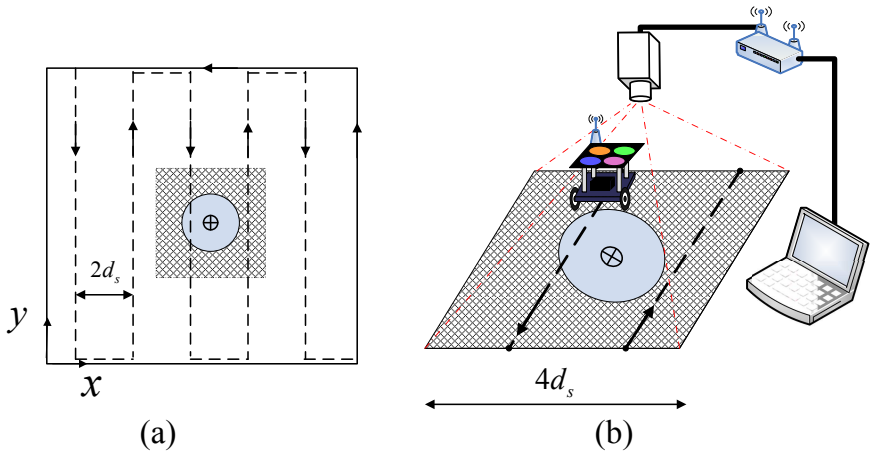


Fig. 8.: Physical experiment settings for the "accelerated" configuration. (a) The shaded area covers the circle. (b) A close-up view of the experiment setup for the shaded area and the overhead camera.

We first validate Corollary 1 in the physical experiment. The results are summarized in Fig. 9. Each entry is an average of 100 trials. The marker positions indicate the mean value of the 100 trials and the vertical bars are the range of $\pm\sigma$ with σ being the standard deviation. The entries with solid vertical bars are physical experiment results whereas the entries with dashed vertical bars represent the corresponding simulation results. As shown the results. The physical experiment results are very consistent with that of the simulation despite the significant difference in sensing region shape. Both results confirmed that the slap method has a shorter EST than that of the random walk. Unlike the simulation, we cannot repeat the experiments under a large range of various parameter settings due to hardware limitations. In the second setup, we validate Corollary 2 and Lemma 1 by comparing different LCRT and ASER configurations. The results are shown in Fig. 10. Again, we follow

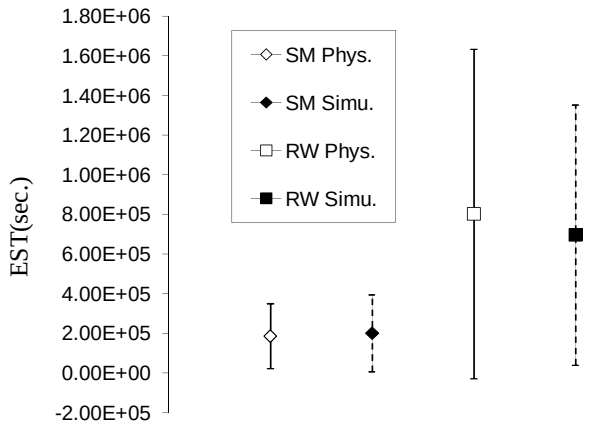


Fig. 9.: Comparing the EST of the slap method and the random walk. Note that "Phys." means the physical experiment results while "Simu." represents simulation results. The vertical bars correspond to $[-\sigma, \sigma]$ with σ as the standard deviation.

the same legend convention of mean and standard deviation as in Fig. 9. Each entry indicates the mean value and the standard deviation of 100 trials. The simulation results and the physical experiment results are consistent. The results conclude that the increasing order of EST of all configurations is,

$$\{ \text{LCRT-SM, ASER-SM, LCRT-P-RW, LCRT-C-RW, ASER-RW} \}$$

with LCRT-SM being the best and ASER-RW being the worst searching methods, respectively. This result is consistent with Corollary 2 and Lemma 1. Also, the fact that ASER-SM is faster than LCRT-P-RW and ASER-RW is slower than LCRT-C-RW is rather interesting and worth further study in the future.

F. Conclusions

We analytically modeled the expected searching time and spatiotemporal sensory coverage for a robot with a limited sensing range to search for a target that intermittently emits short duration signals. We presented the closed-form model for the EST. The

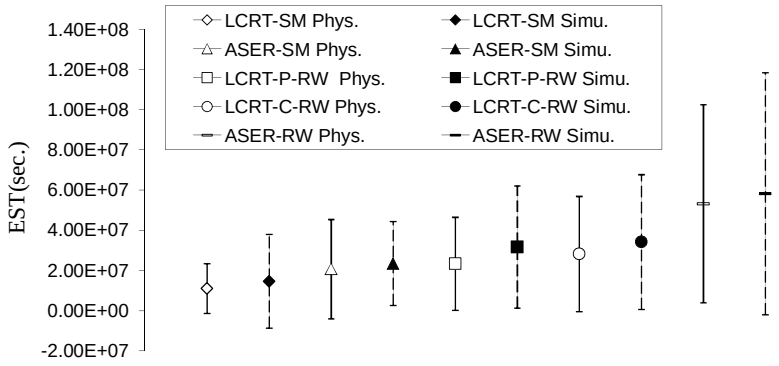


Fig. 10.: Comparing the EST of different LCRT and ASER configurations.

EST model is motion-plan independent and can be used to analyze different motion plans or robot configurations. We demonstrate the analysis process using two case studies. In the first case, we analyzed the slap method and the random walk and found that the slap method is asymptotically faster than the random walk. In the second case, our analysis revealed the interesting result that a low-cost robot team is always asymptotically faster than an expensive robot when the sensory coverage is the same. In both cases, the results demonstrated the usefulness and the capability of our EST analysis. Our theoretically results were extensively tested using simulation and physical experiments. The experimental results were consistent with the analytical models.

Chapter IV extends the localization problem to multiple targets.

CHAPTER IV

CSMA SRMT LOCALIZATION*

A. Introduction

Localizing multiple transient targets of wireless sensor networks cannot be viewed as a simple extension of the single target localization due to challenging issues such as signal anonymity, transiency, and unknown target number often couple together.

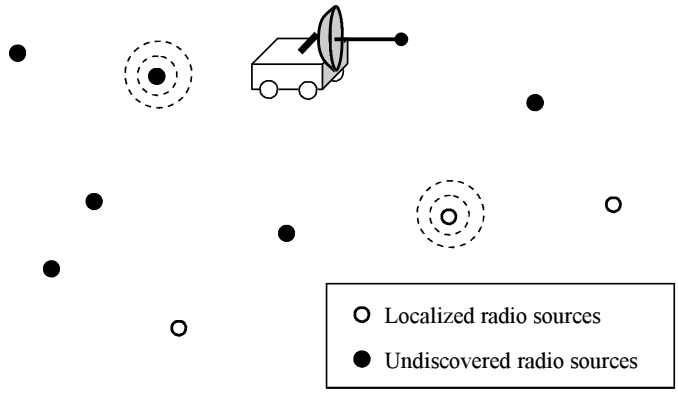


Fig. 11.: Schematics of deploying a single mobile robot to localize unknown wireless sensor network nodes. The nodes with dashed circles indicate that they are transmitting.

Due to the hardware and energy constraints, most of sensor network nodes employ carrier sensing multiple access (CSMA)-based Media Access Control (MAC) protocol or its variations. This allows us to take advantage knowledge of the CSMA MAC protocol in the localization process. However, localizing an unknown wireless

*Reprinted with permission from " Simultaneous Localization of Multiple Unknown CSMA-based Wireless Sensor Network Nodes Using a Mobile Robot with a Directional Antenna" by Dezhen Song, Chang Young Kim, and Jingang Yi, 2009, Journal of Intelligent Service Robots, vol. 2, No. 4, pp 219-233, Copyright 2012 by Springer.

sensor network is different and more difficult than localizing a constant radio beacon due to the unknown network size, transient and intermittent transmissions, and signal source anonymity.

As illustrated in Fig. 22, we are interested in developing systems and algorithms to guide a robot that is equipped with a directional radio antenna to localize unknown networked radio sources. Assisted by its on-board positioning sensors, the robot can detect spatial distribution of radio signal strengths (RSS) as it travels in the field of radio sources. Our approach builds on augmented particle filters and combines a probabilistic sensing model describing the characteristics of a directional antenna, and a CSMA model that can detect network configuration useful for localization purposes. The particle filters output the posterior probability distribution of radio sources. Based on the particle distribution, we develop a motion planning scheme to generate robot control commands to search and localize radio sources. The final localization scheme consists of two algorithms: a sensing algorithm that runs in $O(n)$ time for n particles and a motion planning algorithm that runs in $O(nl)$ time for l radio sources. We have implemented the algorithm and the results show that the algorithms are capable of localizing unknown networked radio sources effectively and robustly.

B. System Design

1. System Architecture

Fig. 12 illustrates the system architecture. Whenever the directional antenna intercepts a transmission, the RSS reading of the transmission enters the system along with the current robot/antenna configuration. The antenna model provides information to particle filters regarding the potential location of the radio source. The CSMA model

updates its estimated number of potential radio sources, each of which corresponds to a particle filter. There are multiple parallel particle filters running with each particle filter corresponding to the spatial distribution of a potential radio source. Hence the spatial distribution of each radio source is represented by the particle distribution of each particle filter. The particle filters are updated based on the antenna model outputs. After each update of the particle filters the system determines if all radio sources are detected. If not, the motion generation algorithms plans robot motion to search for more radio sources.

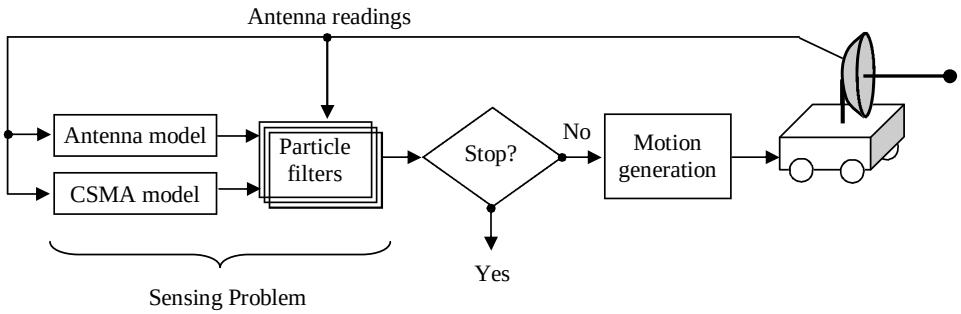
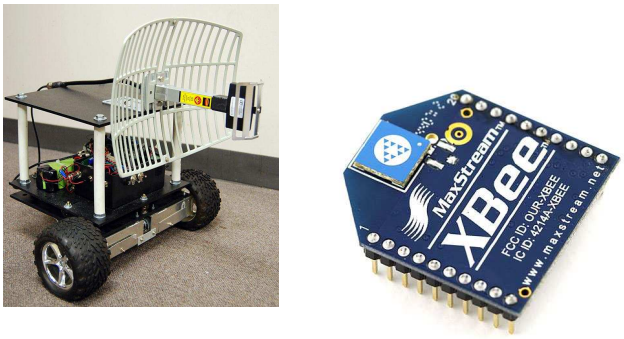


Fig. 12.: System architecture.

2. Hardware

As illustrated in Fig. 13, the robot used in the system is custom made in our lab. The robot measures $50 \times 47 \times 50$ cm^3. The robot has two frontal drive wheels and one rear cast wheel and uses a typical differential driving structure. The robot can travel at a maximum speed of 50 cm/second. The directional antenna is a HyperGain HG2415G parabolic directional antenna with a maximum gain of 15dBm at 2.4GHz. It is an off-the-shelf product from L-com Global Connectivity. The radio sources are 2.4GHz XBee nodes from Maxstream. Each XBee node has a chip antenna and the transmission power of 1 mW.



(a) (b)

Fig. 13.: Hardware of the system. (a) the robot and the directional antenna. (b) Radio sources

C. Problem Definition

We are now ready to formulate our localization problem. To formulate the problem and focus on the most relevant issues, we have the following assumptions:

1. Assumptions

1. The robot and radio sources are in a free 2D space.

2. The received signal does not contain information about the signal source. In fact, the robot usually cannot decode the packet at the MAC level due to the unknown network.

3. Network traffic is light and each transmission is short. These are the typical characteristics of a low power sensor network. Actually, this assumption makes localization more difficult. If traffic is heavy and the transmission duration is lengthy, the robot can localize the active radio source by simply "riding the

wave". In fact, most low power sensor nodes have a packet length magnitude of 10ms.

4. The directional antenna on the robot has high sensitivity and can listen to all traffic. This is the advantage that the robot has.

5. The radiation pattern of the radio sources is circular. This assumption simplifies the modeling process. Later we will show that the proposed method also works for non-circular radiation patterns.

6. The radio sources are stationary. At this stage, we do not consider mobile nodes.

7. All radio sources transmit at the same power level. This assumption is not necessarily true for the most general case. For cases with different unknown power levels, we can use a pair of orthogonal antennas to extract directional information of the radio source regardless of the variation of transmission power level. Hence the proposed method can be easily generalized to cases with different transmission power levels.

8. The robot can accurately execute its motion command. Our focus here is not to study the effect of imprecise motion.

Assumptions 2) and 3) differentiate localizing an unknown wireless sensor network node from localizing a "friendly" continuous radio beacon. Due to the transient and intermittent transmission pattern along with signal source anonymity, the robot cannot simply triangulate the signal source. Since only one robot is considered, the single perspective makes it more difficult than cases with multiple robots or receivers.

2. Nomenclature

- k: a discrete time index variable.
- i: a particle index variable, $i \in \mathbf{I} = \{1, ..., n\}$, where n is the total number of particles and $\mathbf{I}$ is the particle index set. $n \leq n_{\max}$ is not a fixed number, where $n_{\max}$ is maximum number of particles.
- m: an index variable for radio sources, $m \in \mathbf{M} = \{1, ..., l\}$, where l is the total number of radio sources and $\mathbf{M}$ is the radio source index set.
- $\mathbf{x}_m^k$: the location of the m-th radio source at time k. The variable is a random state variable because we do not know the actual location.
- $\mathbf{X}^k$: the joint state for all radio sources at time k, $\mathbf{X}^k = \{\mathbf{x}_1^k, ..., \mathbf{x}_l^k\}$.
- $\mathbf{s}_m^k$: a set of particles for the m-th radio source, $\mathbf{s}_m^k = \{w_{m,i}^k, \vec{x}_{m,i}^k | i \in \mathbf{I}\}$, where each particle has an assigned relative weight $w_{m,i}^k$ and a potential radio source location $\vec{x}_{m,i}^k = [x_{m,i}^k, y_{m,i}^k]^T \in \mathbb{R}^2$.
- $\mathbf{S}^k$: the joint particle set at time k, $\mathbf{S}^k = \{\mathbf{s}_1^k, ..., \mathbf{s}_l^k\}$.
- Z^k: the RSS reading at time k, $Z^k \in [1, 255] \cap \mathbb{N}$.
- $\mathbf{Z}^k = \{Z^1, Z^2, ..., Z^k\}$: the set of all RSS values at time k.
- $\mathbf{u}^k$: a robot/antenna position and orientation at time k, $\mathbf{u}^k = [x^k, y^k, \theta^k]^T \in \mathbb{R}^2 \times \mathbb{S}$, where $\mathbb{S} = (-\pi, \pi]$ is the orientation angle set.

3. Problem Definition

Based on the assumptions, we define our localization problem as follows.

Problem 2 (Localization Problem)**.** *Given all received RF signal strengths* $\boldsymbol{Z}^k$*, compute the number of radio sources,* l*, and estimate the position of each radio source* $\boldsymbol{X}^k$*.*

Since we apply a particle filter approach to address the problem, properly designed particle filters should represent the spacial distribution of $\mathbf{X}^k$. Hence the overall problem can be broken down into the following two subproblems.

Problem 3 (Sensing Problem)**.** *Given all received RF signal strengths* $\boldsymbol{Z}^k$*, compute the number of radio sources,* l*, and the conditional probability of sensor locations* $p(\boldsymbol{X}^k|\boldsymbol{Z}^k)$*.*

Problem 4 (Motion Planning Problem)**.** *Given* $p(\boldsymbol{X}^k|\boldsymbol{Z}^k)$*, plan* $\boldsymbol{u}^{k+1}$ *for next period.*

D. Sensing Problem

The sensing problem is to compute $p(\mathbf{X}^k|\mathbf{Z}^k)$. As illustrated in Fig. 12, there are three major components in the sensing problem: antenna model, CSMA modeling, and particle filters.

1. Antenna Model

From antenna theory, bearing and distance are the two most important variables that determines the radiation pattern distribution in the 2D space for a given antenna. Recall that $\mathbf{u}^k = [x^k, y^k, \theta^k]^T$ is the robot antenna configuration when the radio transmission is sensed at time k and $\mathbf{x}_m^k = [x_m, y_m]^T$ is the m-th radio source position. Define d_m^k and ϕ_m^k as the distance and the bearing from robot to the m-th radio source

position, respectively,

$$d_m^k = \sqrt{(x^k - x_m)^2 + (y^k - y_m)^2},$$
$$\phi_m^k = \text{atan2}(x^k - x_m, y^k - y_m) - \theta^k.$$

Also from antenna theory [75], when the m-th radio source is transmitting, the expected RSS Z^k of the directional antenna is approximated as,

$$E(Z^k) = 10\{\log_{10} C - \beta \log_{10}(d_m^k) + \log_{10} s(\phi_m^k)\}, \tag{4.1}$$

where C is a constant depending on radio transmission power and $(d_m^k)^{-\beta}$ is the signal decay function. The directivity of the antenna is captured by the term $s(\phi_m^k)$, which describes the radiation pattern of the antenna. We obtain $C = 1.77$ and the decay factor $\beta = 2.65$ from antenna calibration. Our β value conforms to the widely-accepted notion that the decay factor is between 2 and 4 [75]. The units of $E(Z^k)$ are dBm. From antenna theory and the results from antenna calibration, we perform curve-fitting to obtain the radiation pattern function,

$$s(\phi_m^k) = \begin{cases} \cos^2(4\phi_m^k), & \text{if} -20^\circ \le \phi_m^k \le 20^\circ \\ \cos^2(80^\circ), & \text{otherwise.} \end{cases} \tag{4.2}$$

Eqs. (4.1) and (4.2) describe the expected RSS given that the radio transmission is from m-th radio source. However, the sensed RSS is not a constant but a random variable due to the uncertainties in radio transmissions. From the antenna calibration, we know that Z^k conforms to the truncated normal distribution with a density function of

$$g(z) = \frac{\frac{1}{\sigma} f(\frac{z - E(Z^k)}{\sigma})}{F(\frac{z_{max} - E(Z^k)}{\sigma}) - F(\frac{z_{min} - E(Z^k)}{\sigma})}, \tag{4.3}$$

where the value of σ is 3.3 by the antenna calibration, z is the sensed RSS, $f(\cdot)$ is the probability density function (PDF) of a normal distribution with zero mean

and unit variance, $F(\cdot)$ is the cumulative distribution function (CDF) of $f(\cdot)$, and z_{min} and z_{max} are the minimum and the maximum RSS that the antenna can sense, respectively. Let

$$G(z) = \int_{z_{min}}^{z} g(z)dz \tag{4.4}$$

be the CDF of the truncated normal distribution.

Since Z^k can only take integer values, we obtain the antenna model as follows,

$$P(Z^k = z|\mathbf{x}_m^k) = G(z + 0.5) - G(z - 0.5). \tag{4.5}$$

2. CSMA Model

One critical part of the sensing problem is to estimate how many radio sources are in the network. Here we utilize the CSMA model to estimate the potential number of sources.

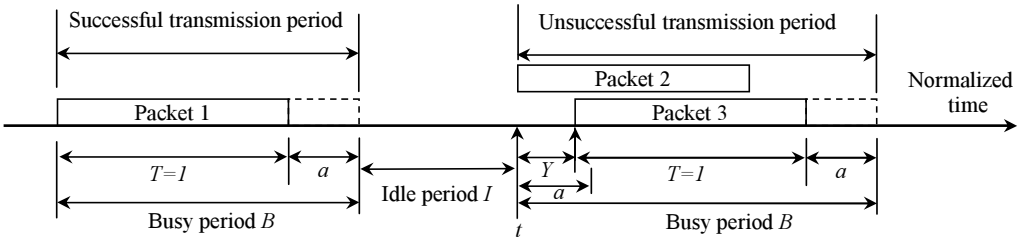


Fig. 14.: CSMA transmission period analysis.

Fig. 14 illustrates the timing of a CSMA protocol. The time axis is alternatively divided into busy and idle periods. In the figure, $a \ll 1$ denotes the propagation delay, t is the starting time of a busy period, and $t+Y$ is the time that the last packet arrives between t and $t+a$, $0 < Y \leq a$. B, I, and U are the durations of the busy period, the idle period, and the time during a cycle that the channel is used without conflicts, respectively. Each busy period is also termed as a transmission period, which is further classified as a successful transmission period or an unsuccessful transmission

period.

Without loss of generality, we set packet length $T = 1$ in Fig. 14. A packet takes additional time a to propagate, and $a \ll 1$. Therefore, a successful transmission takes time $(1 + a)$. a is mainly determined by how fast the circuitry can recognize the transmission. If a radio source transmits packet 2 at time t, then the duration between t and $t + a$ is a "vulnerable" period because other radio sources cannot sense its transmission and may initiate another transmission (Packet 3), which would lead to a collision.

If each radio source transmits according to an independent Poisson process with the same packet generation rate λ, the aggregated transmission rate S is given by $S = l\lambda$. Due to retransmission, the actual packet arrival rate G, called offered traffic rate, is larger than S. By the aggregation of several Poisson signal sources, S is also a Poisson process. G can also be approximated with a Poisson process. The offered traffic rate G is the sum of the source traffic rate S and the retransmission traffic rate R, thus, $G = S + R$.

Define the busy collision probability P_{pc} as the conditional probability of a collision given the channel is busy. Then

$$P_{pc} = 1 - e^{-aG} \tag{4.6}$$

by the approximation that G is Poisson. Since the robot can listen to all traffic, G and P_{pc} can be observed over time. Hence the unknown networked parameter a can be estimated using (4.6). Hence, we treat a as known in the rest of the paper. Upon each collision, there are two retransmissions scheduled $R = 2GP_{pc}$. Therefore,

$$G = S + 2GP_{pc} = S. \tag{4.7}$$

With G and P_{pc} observed, we can obtain S using (4.7). If we know λ, then we can

obtain $l = S/\lambda$. However, this would not work for the most general cases because 1) λ is usually unknown and 2) λ might not be the same across different radio sources. To handle the problem, we can envision that each radio source can be further divided into multiple collocated sub radio sources with each sub radio source shares the same transmission rate $\lambda' << \lambda_{\min}$, where $\lambda_{\min}$ is the smallest transmission rate of the original radio sources.

Hence we can still apply the condition that each radio source has the same transmission rate of λ'. The number l will be much bigger than the actual l. However, this is not a concern because we can always combine collocated sources after they are localized. For this reason, we assume each radio source shares the same transmission rate in the rest of the paper.

3. Particle Filters

We now know that there are l radio sources. For each radio source m, we use a particle filter to track its spacial distribution $p(\mathbf{x}_m^k|Z^k)$. This is an instance of the Bayes filtering problem which can be computed using a two-phase recursive approach:

- Prediction Phase:

$$p(\mathbf{x}_m^k|Z^{k-1}) = \int p(\mathbf{x}_m^k|\mathbf{x}_m^{k-1}, \mathbf{u}^k)p(\mathbf{x}_m^{k-1}|Z^{k-1})d\mathbf{x}_m^{k-1}. \tag{4.8}$$

  Since positions of radio sources are static, state $\mathbf{x}_m^k$ is independent of the deterministic robot motion $\mathbf{u}^k$. Therefore, the prediction phase in (4.8) is trivial,

$$p(\mathbf{x}_m^k|Z^{k-1}) = p(\mathbf{x}_m^{k-1}|Z^{k-1}). \tag{4.9}$$

- Update Phase:

$$p(\mathbf{x}_m^k|Z^k) = \eta p(Z^k|\mathbf{x}_m^k)p(\mathbf{x}_m^k|Z^{k-1})$$
$$= \eta p(Z^k|\mathbf{x}_m^k)p(\mathbf{x}_m^{k-1}|Z^{k-1}) \quad (4.10)$$

where η is a normalizing factor.

The particle filter represents $p(\mathbf{x}_m^k|Z^k)$ by a set of particles $\mathbf{s}_m^k$. Recall that $\mathbf{s}_m^k = \{w_{m,i}^k, \vec{x}_{m,i}^k | i = 1, ..., n\}$ where n is the total number of particles, $w_{m,i}^k$ is the assigned weight for the particle, and $\vec{x}_{m,i}^k = [x_{m,i}^k, y_{m,i}^k]^T \in \mathbb{R}^2$ is the potential radio source location.

The update phase in the particle filters is performed in two stages: *importance sampling* and *resampling*.

a. Importance Sampling

The importance sampling weights each of the samples

$$w_{m,i}^k = w_{m,i}^{k-1} p(Z^k|\vec{x}_{m,i}^k) \quad (4.11)$$

by the sensor model $p(Z^k|\vec{x}_{m,i}^k)$ that can be computed using (4.5). Each particle in $\mathbf{s}_m^k$ is randomly drawn from $\mathbf{s}_m^{k-1}$ proportional to the updated weight $w_{m,i}^k$. The importance sampling step reduces the number of low weighted particles and increases the number of high weighted particles.

b. Resampling

After a few iterations of the importance sampling, the number of survived particles shrinks and ultimately becomes zero, which causes the degeneracy problem. The

problem can be solved by adding more particles into $\mathbf{s}_m^k$ by resampling when the effective number of particles is below an effective threshold number. Let n_{eff} denote the effective number, which is computed based on weights,

$$n_{\text{eff}} = \frac{1}{\sum_{i=1}^{n}(w_{m,i}^k)^2} \tag{4.12}$$

according to [76]. We define n_t as the threshold that is determined by the experiments. If $n_{\text{eff}} < n_t$, we perform resampling.

Resampling also introduces the problem of loss of diversity among particles. This is because samples are drawn from a discrete particle set rather than from a continuous distribution. In order to solve this problem, it is necessary to modify the resampling process by introducing Gaussian random noise into the resampled particles. Let $N(\vec{\mu}_r, \Sigma_r)$ denote the two dimensional Gaussian distribution where $\vec{\mu}_r$ and Σ_r are the mean vector and the covariance matrix, respectively, where $\vec{\mu}_r = \vec{x}_{m,i}^{k-1}$ and Σ_r is a tunable diagonal matrix determined by experiments. Therefore, particles in $\mathbf{s}_m^k$ are obtained by resampling from $\{w_{m,i}^k, N(\vec{x}_{m,i}^{k-1}, \Sigma_r) | i = 1, ..., n\}$.

Another problem of resampling is that there could be no particle in vicinity of the correct state. This is known as the *particle deprivation problem*. To address the problem, we add a 5% randomly generated particles into $\mathbf{s}_m^k$ with an initial weight of $1/n$ each.

4. Data Association

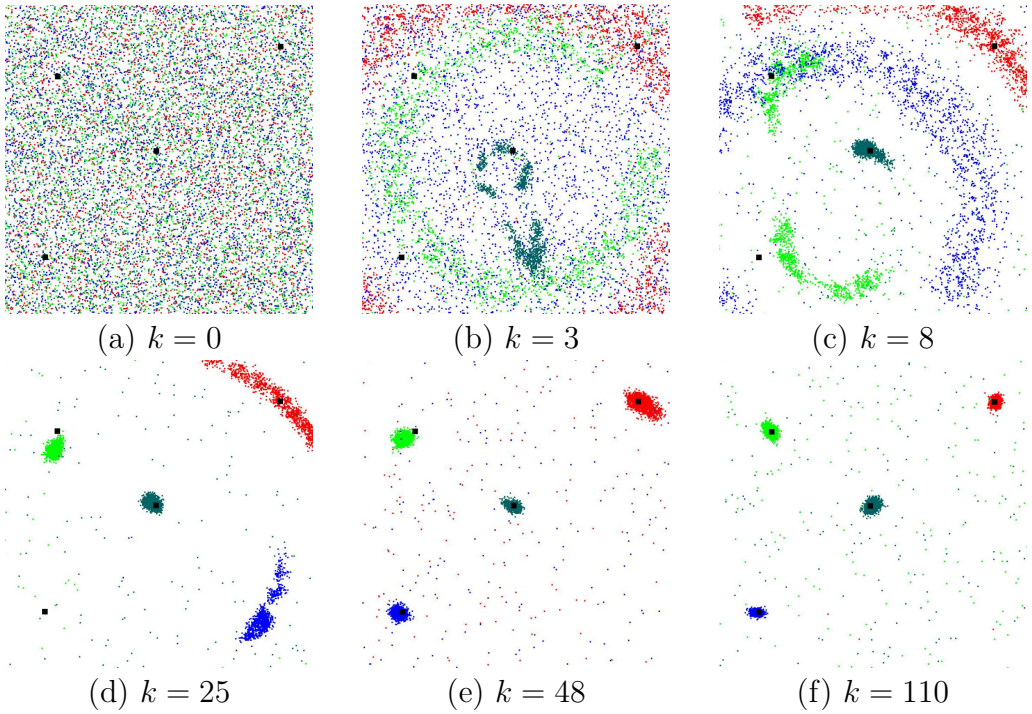
For l radio sources, there are l particle filters. It is important to determine which particle filter to be updated once a RSS is perceived. This is a data association problem. We use maximum a posteriori probability (MAP) estimation to address the problem. Let $\hat{p}(\mathbf{x}_m^k | Z^k)$ be the posterior probability estimation of the m-th particle

set,

$$\hat{p}(\mathbf{x}_m^k|Z^k) = \frac{\sum_{i=1}^{n} w_{m,i}^{k-1} p(Z^k|\vec{x}_{m,i}^{k-1})}{\sum_{j=1}^{l}\sum_{i=1}^{n} w_{j,i}^{k-1} p(Z^k|\vec{x}_{j,i}^{k-1})}. \tag{4.13}$$

Let m^* be the index for the selected radio source, which is chosen by maximizing $\hat{p}(\mathbf{x}_m^k|Z^k)$,

$$m^* = \arg\max_{m\in\mathbf{M}} \hat{p}(\mathbf{x}_m^k|Z^k). \tag{4.14}$$



(a) $k = 0$ (b) $k = 3$ (c) $k = 8$

(d) $k = 25$ (e) $k = 48$ (f) $k = 110$

Fig. 15.: Sample results of particle distribution with respect to actual radio source location over time k. There are four radio sources represented by black dots. The smaller color dots indicate each individual particle. Four different colors represent results of four particle filters. The robot performs random walk in this example.

5. Stopping Time and Localization Criterion

With the MAP approach, we can selectively update an individual particle filter. Fig. 15 illustrates the results of the particle distribution with respect to the actual

radio source location over time k. It is clear that the majority of particles converge to the vicinity of the radio source location. As a Monte Carlo method, it is necessary to determine a stopping time that detects convergency trend of the particles as a function of each individual particle set $\mathbf{s}_m^k$.

Since particles are located in the 2D space, the spatial distribution of particles in $\mathbf{s}_m^k$ are can be described by a mean vector $\vec{\mu}_m$ and a covariance matrix Σ_m. Hence we have,

$$\vec{\mu}_m^k = \sum_{i=1}^{n} w_{m,i}^k \vec{x}_{m,i}^k, \tag{4.15}$$

$$\Sigma_m = \frac{\sum_{i=1}^{n} w_{m,i}^k [(\vec{x}_{m,i}^k - \vec{\mu}_m^k)(\vec{x}_{m,i}^k - \vec{\mu}_m^k)^T]}{1 - \sum_{i=1}^{n} (w_{m,i}^k)^2}. \tag{4.16}$$

Define λ_m and $\mathbf{V}_m$ as the maximum eigenvalue and the corresponding eigenvector of Σ_m, respectively. According to principle component analysis (PCA), we know that the maximum variance of the particle distribution in the 2D space can be measured by its largest eigenvalue λ_m. As the particle set converges to the vicinity of the radio source, λ_m should decrease. Define ϵ as the threshold for λ_m. We define that the m-th radio source is located and we can stop the corresponding particle filter computation if

$$\lambda_m \leq \epsilon. \tag{4.17}$$

6. Algorithm

The computation of the sensing model can be summarized in Algorithm 1. It is clear that each iteration of Particle Filter-based Sensing Algorithm (PFSA) runs in $O(n)$ time for n particles.

Algorithm 1: Particle Filter-based Sensing Algorithm

input : $\mathbf{Z}^k$

output: $\mathbf{s}_m^k$

begin

Update G	$O(1)$
Estimate S and l according to (4.7)	$O(1)$
Compute $\hat{p}(\mathbf{x}_m^k \mid Z^k)$ using (4.13)	$O(n)$
Find m^* using (4.14)	$O(1)$
Compute $w_{m^*,i}^k$, $i \in \mathbf{I}$, using (4.11)	$O(n)$
Normalize $w_{m^*,i}^k$, $i \in \mathbf{I}$	$O(n)$
$n_c = 0$; $\mathbf{s}_{m^*}^k = \emptyset$	$O(1)$
for $i = 1$ **to** n **do**	$O(n)$
Draw i from $\mathbf{s}_{m^*}^{k-1}$ with probability $\propto w_{m^*,i}^k$	$O(1)$
if $\{w_{m^*,i}^k, \vec{x}_{m^*,i}^{k-1}\} \notin \boldsymbol{s}_{m^*}^k$ **then**	
$\mathbf{s}_{m^*}^k = \mathbf{s}_{m^*}^k \cup \{w_{m_c,i}^k, \vec{x}_{m_c,i}^{k-1}\}$	$O(1)$
$n_c = n_c + 1$	$O(1)$
$n = n_c$	$O(1)$
Compute n_{eff} using (4.12)	$O(1)$
Compute λ_{m^*} using PCA	$O(1)$
if $\lambda_{m^*} \leq \epsilon$ **then**	
radio source m^* is localized.	$O(1)$

E. Motion Planning Problem

As illustrated in Fig. 15, the particle sets track the spatial distribution of radio sources. However, the results shown in Fig. 15 are based on a robot performing a random walk,

which is not necessarily the best choice for robot motion planning. We need to develop an effective robot motion planner to ensure $\mathbf{s}_m^k$ converges.

We propose a two step approach. First, the robot chooses a targeted radio source m_t to investigate. Then the robot determines its configuration that best ensures the convergence of $\mathbf{s}_{m_t}^k$.

1. Choosing a Target

The process of choosing a target largely depends on how well each particle set converges and the traveling distance of the robot. For m-th particle set, recall that a smaller λ_m means radio source m is closer to be localized. Hence, the robot can localize the target without spending too much time. On the other hand, we would like the robot to travel the minimum distance to save energy. We define the following function to describe the tradeoff between the convergence status and the traveling distance,

$$\omega_m = \alpha\lambda_m + (1-\alpha)d_{\mu_m}, \tag{4.18}$$

where $0 \leq \alpha \leq 1$ is the weighting factor between convergence and distance, and d_{μ_m} is the distance between the robot's current position and the estimated position of m-th radio source $\vec{\mu}_m^k$.

A radio source with a small ω_m would be a desirable target for the robot. However, if we use this metric, the robot might stick with a prominent target and fail to explore other targets. To avoid this, we define a history weighting function,

$$h(m_c) = \begin{cases} \infty & \text{if } \tau_{m_c} > \tau_{max} \\ 1 & \text{otherwise} \end{cases} \tag{4.19}$$

where m_c is the current target, τ_{m_c} is the elapsed time that the robot has been with

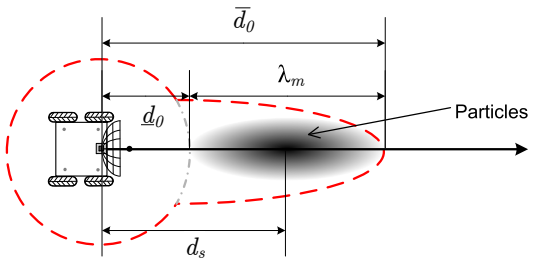


Fig. 16.: Sample robot configuration for a particle set. The gray ellipsoid region represents particle distribution. The dashed red line represents the directivity of the antenna (i.e. function $s(\cdot)$ in (4.2)).

the current target, and τ_{max} is the time threshold for the maximum investigation duration. At each step, τ_{m_c} is updated as follows,

$$\tau_{m_c} = \begin{cases} \tau_{m_c} + 1 & \text{if } m_c \text{ has not change} \\ 0 & \text{otherwise.} \end{cases} \tag{4.20}$$

Therefore, the robot is forced to investigate other targets once τ_{max} is reached. τ_{max} can be obtained using the transmission rate λ and a probability threshold p_m. The probability that the targeted radio source does not transmit any signal during τ_{max} is $1 - e^{-\lambda\tau_{max}}$. If we want the probability to be less than p_m, we can choose

$$\tau_{max} = -\frac{1}{\lambda}\ln(1 - p_m). \tag{4.21}$$

Combining convergence, traveling distance, and history, we choose the targeted radio source m_t that minimizes the following

$$m_t = \arg\min_{m \in \mathbf{M}} h(m_c)\omega_m. \tag{4.22}$$

2. Robot Configuration

Once a target radio source m is identified, we need to identify a corresponding robot configuration that can accelerate the convergence of the particle set $\mathbf{s}_m^k$. As illustrated in Fig. 16, an intuitive choice is to align the most sensitive reception region of the directional antenna with the particle set. In this way, the robot does not need to travel too close to the radio source, the robot can reduce its travel distance, and be energy efficient.

To ensure a good alignment between the antenna and the particle ellipsoid describing regions with a high concentration of particles, it is necessary to align the zero bearing angle of the antenna with the long axis of the particle ellipsoid. Recall that $\mathbf{V}_m$ represents the eigenvector that corresponds to the maximum eigenvalue of matrix Σ_m. Let $v_{m,x}$ and $v_{m,y}$ be the x- and y- components of $\mathbf{V}_m$, respectively. We know that the long axis of the ellipsoid is determined by $\mathbf{V}_m$ according to PCA. Hence, the orientation of the robot/antenna is,

$$\theta = \text{atan2}(v_{m,x}, v_{m,y}). \tag{4.23}$$

The remaining parameter is the distance between the robot and the center of the particles. As illustrated in Fig. 16, the distance is defined as d_s . If d_s is obtained, the robot position $[x, y]$ is obtained straightforwardly,

$$\begin{aligned} x &= \vec{\mu}_{m,x} - d_s \cos(\theta), \\ y &= \vec{\mu}_{m,y} - d_s \sin(\theta). \end{aligned} \tag{4.24}$$

where $\vec{\mu}_{m,x}$ and $\vec{\mu}_{m,y}$ represent the x- and y- components of $\vec{\mu}_m^k$, respectively. $\vec{\mu}_m^k$ is the center the particle set $\mathbf{s}_m^k$ according to (4.15). Therefore, we need to compute d_s. From PCA, we know that the ellipsoid in Fig. 16 is the approximation of the particle

distribution. As illustrated in Fig. 16, we define $\overline{d}_0$ and $\underline{d}_0$ as the interception points of the outer and inner boundaries of the main reception area with the zero bearing axis, respectively. The boundary functions are described in (4.2). We would like to fit the long axis of the ellipsoid in between $\overline{d}_0$ and $\underline{d}_0$,

$$\lambda_m = \overline{d}_0 - \underline{d}_0. \tag{4.25}$$

Since λ_m is known, this allows us to find the expected signal strength using (4.1),

$$E(Z) = 10[\log_{10} C - \beta \log_{10} \lambda_m + \beta \log_{10}(1 - 10^{\frac{\log_{10} \cos^2 80^\circ}{\beta}})]. \tag{4.26}$$

The expected signal strength can help us to compute $\overline{d}_0$ and $\underline{d}_0$ and obtain

$$d_s = \underline{d}_0 + \frac{1}{2}\lambda_m. \tag{4.27}$$

Therefore, the robot configuration $[x, y, \theta]^T$ is found. As more transmission are intercepted, the particles converge and λ_m decreases. Consequently, the robot adaptively moves close to the radio source to increase localization accuracy. Hence we name this approach as Greedy Adaptive Motion Planning (GAMP).

3. Algorithm

We summarize our GAMP algorithm in Algorithm 2. It is clear that the algorithm runs in $O(nl)$ time for n particles and l radio sources.

Algorithm 2: GAMP Algorithm

input : $\mathbf{s}_m^k$

output : $[x, y, \theta]^T$

begin

for $m = 1$ **to** l **do**	$O(l)$
Compute $\vec{\mu}_m$ and Σ_m using (4.15) and (4.16)	$O(n)$
Perform PCA on Σ_m and obtain λ_m	$O(1)$
end	
Update τ_{m_c} using (4.20)	$O(1)$
Compute τ_{max} using (4.21)	$O(1)$
Compute ω_m using (4.18)	$O(1)$
Find m_t according to (4.22)	$O(1)$
Compute θ using (4.23)	$O(1)$
Compute $E(Z)$ using (4.26)	$O(1)$
Compute $\overline{d}_0$ and $\underline{d}_0$ using (4.1)	$O(1)$
Compute d_s using (4.27)	$O(1)$
Compute $[x, y, \theta]$ using (4.24)	$O(1)$

end

F. Experiments

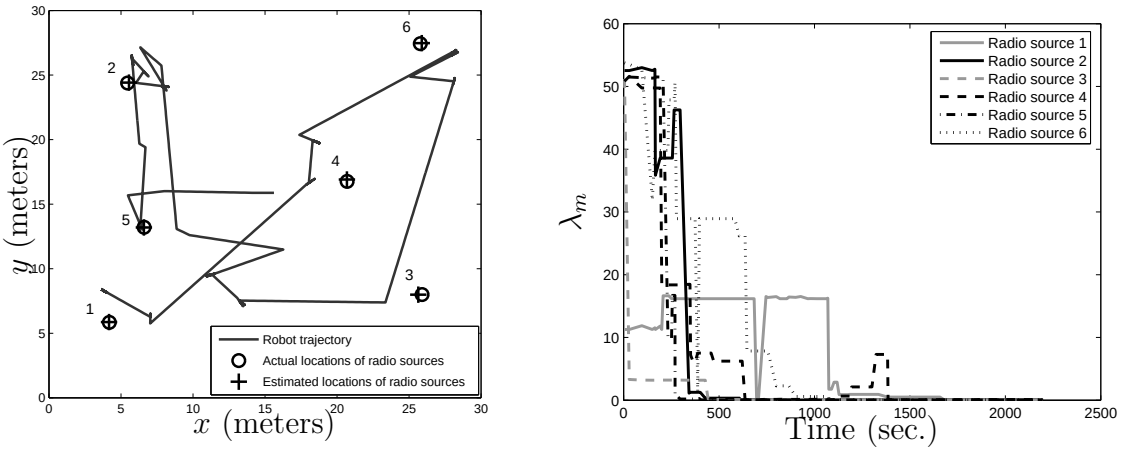
We have implemented the algorithms and the simulation platform using Microsoft Visual C++ .NET 2005 with OpenGL on a PC Desktop with an Intel 2.13GHz Core 2 Duo CPU and 2GB RAM. The machine runs Microsoft Windows XP. To use the actual radiation pattern to drive our simulation, we first calibrate the directional antenna. The calibration is conducted at 328 configurations using 6560 readings. The calibration establishes the parameters in (4.1).

For the unknown network, each radio source generates radio transmission signals according to an independently and identically distributed Poisson process with a rate of 0.01 packets per second. The packet length is 0.01 seconds. The propagation delay a is 3% of the packet length. The radio sources are located in a square field with a side length of 30 meters.

For the particle filter, we set the maximum number of particles for each radio source $n_{\max} = 3000$, the threshold for the effective number of particles $n_t = 1000$, and the covariance matrix for adding the Gaussian noise $\Sigma_r = \begin{pmatrix} \frac{1}{5} & 0 \\ 0 & \frac{1}{5} \end{pmatrix}$. For localization stopping condition, we set $\epsilon = 0.05$. For the motion planning, we set robot speed at 0.25 m/sec., weighting factor $\alpha = 0.9$, and the probability threshold for transmission $p_m = 0.02$. Those parameters are set based on the best performance derived from multiple trials in experiments.

1. A Sample Case

Fig. 17 illustrates the robot trajectory and the convergence trends for a sample case with six radio sources. The initial position of the robot is the center of the field. As we can see from Fig. 17a, the robot gradually approaches each radio source. At the end of experiments, the estimated locations of radio sources are conformal to the actual locations of the radio sources. The localization process is successful. Fig. 17b illustrates how λ_m for each radio source converges over time. All λ_m's successfully converge. What is worth noting is the relationship between the robot position and the convergence trend. If we take a close look at radio source 1, we can find that it converges last because it is the last radio source the robot approaches. We consider the convergence speed satisfying because each radio source only transmits at a mean rate of 1 packet per 100 seconds.



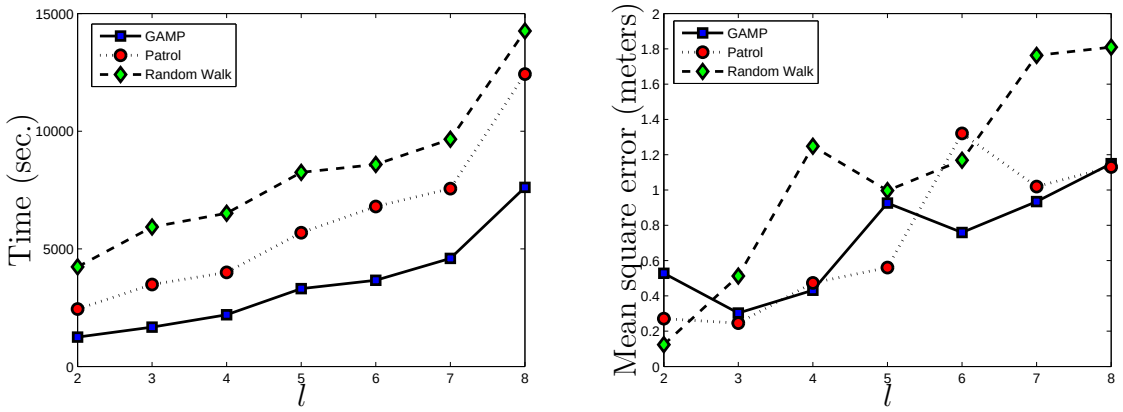
(a) Trajectory (b) Convergence

Fig. 17.: The resulting robot trajectory and the convergence trends for a sample case with six radio sources.

For the sample case, the PFSA runs in 113 milliseconds and GAMP algorithm runs in 0.6 milliseconds. This is not surprising because they are linear algorithms. Since computation speed is not a concern here, we skip speed tests in the rest of experiments.

2. Comparison with Two Heuristic Approaches

We also compare our GAMP to two heuristic approaches, namely, a random walk and a fixed-route patrol. The fixed-route patrol traverses the field using a pre-defined route. We increase the total radio source number l from 2 to 8 to observe the performance of each method. For each trial, we randomly generate radio source locations and test all three methods. We repeat 20 trials for each case and compute the average time required for localizing all radio sources and the mean square error between estimated radio source locations and the actual source locations. Comparison results are shown in Fig. 18. All algorithms are able to localize radio sources. As illustrated



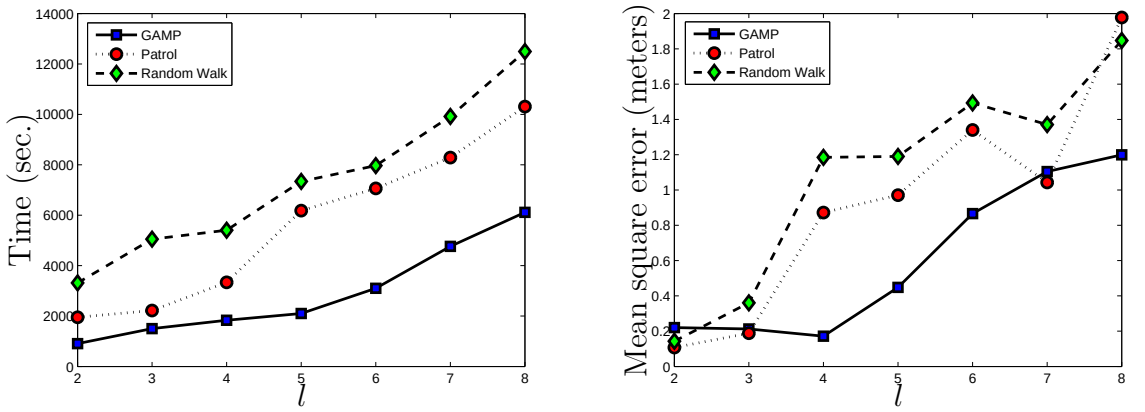
(a) Time comparison (b) Localization accuracy comparison

Fig. 18.: Localization performance comparison among the GAMP, a random walk, and a fixed route patrol.

Fig. 18a, GAMP is consistently faster than the two counterparts. As for the localization accuracy, Fig. 18b shows that all methods are similar in localization accuracy. The accuracy decreases as number of radio sources increases. We conjecture that this is due to the RSS resolution of the antenna and the randomness in RSS make all methods unable to distinguish the radio sources that are close to each other. Hence, the data association step in Section 4 might associate the radio source with a wrong particle filter. Hence the accuracy of localization decreases.

3. Robustness Tests

Our localization method is derived under a set of restrictive assumptions. In this part, we are interested in testing the system performance when relaxing some assumptions. In other words, we would like to know the robustness of the proposed method. More specifically, we focus on the most restrictive assumptions, which are Poisson arrival processes, circle radiation patterns, and evenly distributed traffic among radio sources.

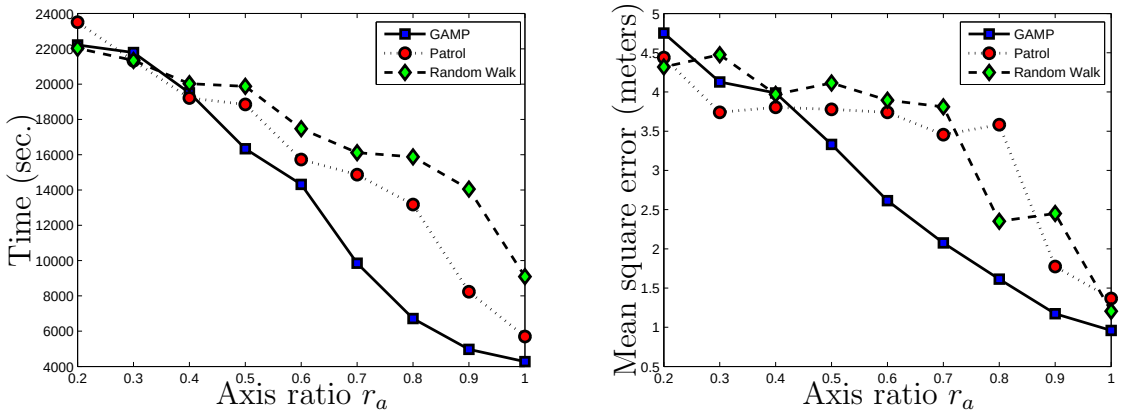


(a) Time comparison (b) Localization accuracy comparison

Fig. 19.: Localization performance comparison under Gaussian inter-arrival time.

The first test of robustness is to relax the assumption that the packets are generated according to Poisson processes. In reality, if a particular routing mechanism is used, then the packet generation processes could deviate significantly from Poisson processes. We simulate those traffic patterns by using Gaussian inter-arrival time. The Gaussian distribution has a mean of 0.01 second and a variance of 100. Other parameters are the same as the sample case test in Fig. 17. Fig. 19a shows that our method is still consistently faster than its two counterparts. For localization accuracy, our method is slightly better. This indicates that our method is not limited to the Poisson arrival process.

The second test of robustness concerns localization performance when the radiation pattern of the radio sources are non-circular. Due to different surface conditions, materials, and environment influence, the radiation pattern of a wireless sensor node is not necessarily round. To characterize this problem, we use an ellipse radiation pattern to approximate the real radiation pattern. To quantify the deviation from the circular radiation pattern, we define axis ratio r_a as the ratio between the minor

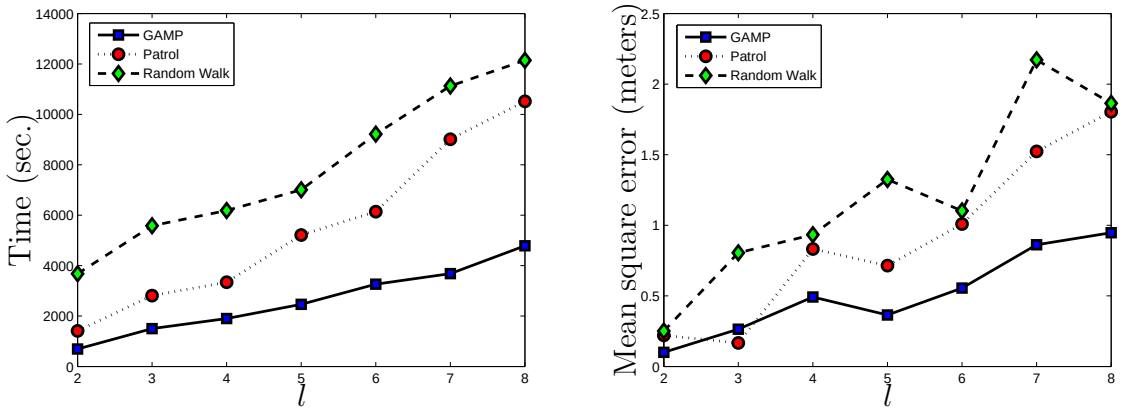


(a) Time comparison (b) Localization accuracy comparison

Fig. 20.: Localization performance comparison for a case with 6 radio sources and non circular radiation pattern.

axis and the major axis of the ellipse. If $r_a = 1$, the radiation pattern is perfectly circular. We vary the ratio from 0.2 to 1. We use a 6-radio source setup in the experiment and 20 random trials for each axis ratio. To avoid the possibility of failure to converge, we set the maximum running time of the simulation as 30000 sec. Fig. 20 illustrates the results. Fig. 20a shows that all three methods are very slow when r_a is small and become fast when the axis ratio increases. When r_a is small, the ellipse is long and narrow. Hence the antenna model cannot provide a reasonably accurate prediction of the location of the radio sources. Our GAMP method become faster than the other two when $r_a > 0.4$. Similar results in localization accuracy are shown in Fig. 20b. These results suggest that our GAMP method is more robust to non-circular radiation pattern than the other two, which is desirable.

The third test of the robustness focuses on the scenario where the traffic might be unevenly distributed in a sensor network. Due to the popularity of clustering techniques in routing, certain nodes (i.e., cluster heads/routers) have much higher



(a) Time comparison (b) Localization accuracy comparison

Fig. 21.: Localization performance comparison with uneven transmission rate.

traffic than other nodes in the network. In this test, we set one radio source to transmit at 0.05 packets per second and other radio sources transmit at 0.01 packets per second. The rest of setup is the same as those in Fig. 18. Fig. 21 illustrates the time and accuracy comparison results. Once again, our GAMP method is better than the other counterparts when the transmission rate is uneven.

All of the tests show that our localization method is more robust to the violation of assumptions than the fixed route patrol and the random walk.

G. Conclusions

We reported how to use a single mobile robot equipped with a directional antenna to localize unknown networked radio sources. We proposed a particle filter-based localization approach that combines a CSMA model and a directional antenna model. We also proposed a motion planning algorithm based on the particle distribution. The sensing algorithm runs in $O(n)$ time for n particles and the motion planning algorithm runs in $O(nl)$ time for l radio sources and n particles. We have implemented the

algorithm and tested it using a real data-driven simulation platform. The results show that the algorithm is capable of localizing unknown networked radio sources. The experiment results shown that the proposed localization method is faster, more accurate, and more robust than the two other heuristic methods.

Chapter V relaxes the dependency of the CSMA protocol and develops protocol independent SRMT localization scheme.

CHAPTER V

PROTOCOL INDEPENDENT SRMT LOCALIZATION*

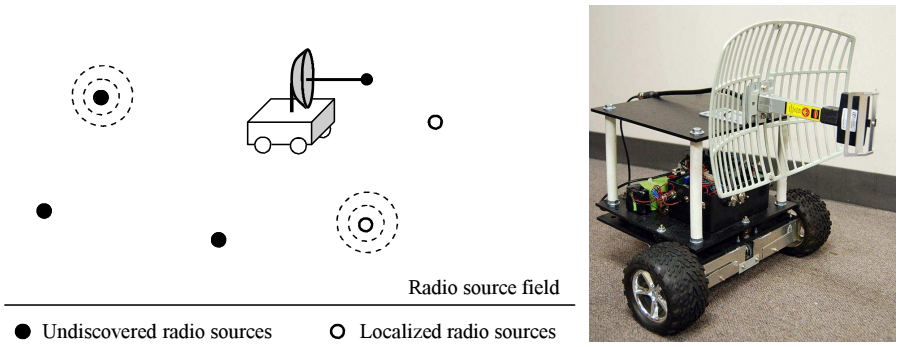
A. Introduction

The CSMA SRMT localization scheme suffers from scalability because the number of particle sets increase as the number of sensor nodes increase. Besides, all of the network MAC protocols do not necessarily follow the CSMA model, thereby failing this approach. It drives us to develop the protocol independent SRMT localization scheme. First, we model the radio source behaviors using a novel spatiotemporal probability occupancy grid (SPOG) that captures transient characteristics of radio transmissions and tracks their posterior probability distributions. Based on SPOG, We propose a Monte Carlo motion planning algorithm that enables the robot to efficiently traverse high probability regions to accelerate the convergence of the posterior probability distributions of radio sources. We have implemented the algorithms and extensively tested them in comparison to a random walk and a fixed-route patrol mechanism. In experiments, our algorithms have shown consistently superior performance over its the two heuristics.

B. System Design

Fig. 23a illustrates system architecture. From the robot perspective, the input is the RSSs from the antenna with the corresponding antenna positions and orientations. The output of the system is the planned trajectory for the robot to execute in the

*Reprinted with permission from "Simultaneous Localization of Multiple Unknown and Transient Radio Sources Using a Mobile Robot" by Dezhen Song, Chang Young Kim, and Jingang Yi, 2012, IEEE Trans. on Robotics (T-RO), vol. 28, no. 3, pp. 1-13, Copyright 2012 by the IEEE.



(a) Schematics. (b) The robot.

Fig. 22.: Schematics of deploying a single mobile robot to localize unknown transient radio sources. The radio sources with dashed circles indicate that they are sending radio signals.

following period. The entire system is built around the SPOG, which tracks each cell's probability of containing a radio source and its transmission rate.

On the one hand, the system updates the SPOG whenever a radio transmission is detected by the antenna. The antenna model outputs the posterior probability distribution of the signal source as the inputs to the SPOG. This update process is described by a continuous time system. We use t to denote the continuous time throughout the paper. On the other hand, the robot plans its motion periodically with period index $k \in \mathbb{N}$. We define the period length as τ_0, which is carefully chosen to ensure the robot has enough time to execute the planned trajectory. At the beginning of each period, the robot plans its trajectory based on the current SPOG.

Fig. 23b illustrates the relationship between the continuous time system and the discrete time system. Let $t^k \in \mathbb{R}$ be the exact continuous time at the moment of the discrete time k. We define the k-th period as the time interval between t^{k-1} and t^k. Hence $t^k - t^{k-1} = \tau_0$ for $k > 1$. We also define $t^k_j \in \mathbb{R}$ as the exact continuous time when the j-th radio transmission occurs in the k-th period: $t^{k-1} \leq t^k_j < t^k$. j is set

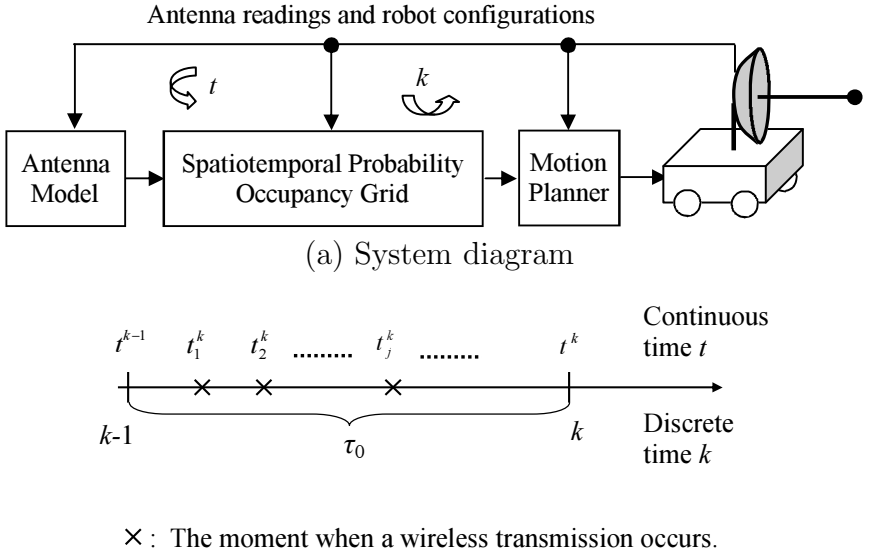


Fig. 23.: An illustration of system diagram and timing.

to zero at the beginning of each period.

C. Problem Definition

To setup the localization problem, we have the following. 1) Both the robot and radio sources are located in a free 2D Euclidean space. 2) The network traffic is light and each transmission is short, which are the typical characteristics of a low power sensor network. 3) The directional antenna on the robot has high sensitivity and can listen to all traffic because the robot has space and power advantage over sensor nodes. 4) The radio sources are static nodes. 5) Radio transmissions have the same power level. This assumption can be relaxed if the robot is equipped with an orthogonal antenna pair, which can provide directional information regardless of the transmission power. 6) The radiation pattern of radio sources is circular because most miniature wireless sensors are equipped with omni-directional antennas. Due to

the transient transmission and signal anonymity, the robot cannot simply triangulate the signal source. Since only one robot is considered, the single perspective makes it more difficult than cases with multiple robots or receivers.

1. Spatiotemporal Probability Occupancy Grid

We introduce SPOG to track the posterior spatiotemporal distributions of radio sources. To define the SPOG, we partition the entire field into equally-sized square cells using a grid. Let us define cell index set $I := \{1, ..., n\}$, where n is the total number of cells. Define $i \in I$ as a cell index variable. The size of each cell is determined by the RSS resolution of the antenna. Inside each cell, we approximate radio source locations using cell center locations. Define C_i as the event that cell i contains at least one radio source and $P(C_i)$ as the probability that event C_i occurs.

At t_j^k, a transmission occurs. We define C_i^1 as the event that cell i is the active radio source at time t_j^k. Define C_i^0 as the event that cell i is inactive at time t_j^k. Hence

$$P(C_i^0) + P(C_i^1) = 1 \text{ and } \sum_{i \in I} P(C_i^1) = 1 \tag{5.1}$$

because there is only one active transmission when the transmission is detected. We ignore the collision case because we read the RSS as soon as the transmission is initiated. The probability of two or more transmissions that are initiated at the exact same moment is negligible in a light traffic network. C_i^1 is determined by the relative radio transmission rate and is the temporal part of the SPOG. Unlike a regular occupancy grid, the SPOG is unique because each cell is described by two types of correlated random events: the spatial event C_i and the temporal events C_i^0 and C_i^1.

2. Problem Formulation

Fig. 23a suggests that the overall localization problem can be divided into two sub problems: a sensing problem and a motion planning problem. Let random variable $Z_j^k \in [1, 255] \cap \mathbb{N}$ be the corresponding RSS at time t_j^k. Note that the RSSs are from a receiver with a resolution of eight bits. Define $\mathbf{Z}(Z_j^k)$ as the set of all RSSs sensed from the beginning of the localization process to the moment when Z_j^k is sensed. We also define set $\mathbf{Z}^-(Z_j^k) := \mathbf{Z}(Z_j^k) - \{Z_j^k\}$, which is the set of all RSSs from the beginning of the localization process to the moment right before Z_j^k is sensed. Define $P(C_i|\mathbf{Z}(Z_j^k))$ as the conditional probability that cell i contains at least one radio source given the RSS set $\mathbf{Z}(Z_j^k)$. Similarly, we define the $P(C_i|\mathbf{Z}^-(Z_j^k))$, $P(C_i^1|\mathbf{Z}(Z_j^k))$, and $P(C_i^1|\mathbf{Z}^-(Z_j^k))$. The sensing problem updates the SPOG when a new transmission is detected,

Problem 5 (Sensing Problem)**.** *Given the current sensed RSS Z_j^k, previous RSS set $\mathbf{Z}^-(Z_j^k)$, $P(C_i|\mathbf{Z}^-(Z_j^k))$, $P(C_i^1|\mathbf{Z}^-(Z_j^k))$, and the corresponding robot configurations, compute $P(C_i|\mathbf{Z}(Z_j^k))$ and $P(C_i^1|\mathbf{Z}(Z_j^k))$ for each cell i.*

At the beginning of each period k, we plan the robot trajectory for the period. Let us define the robot position and orientation as $\mathbf{r}(t) = [x(t), y(t), \theta(t)]^T \in \mathbb{R}^2 \times S$, where $S = (-\pi, \pi]$ is the orientation angle set. Since the antenna is fixed on the robot and points to the robot forwarding direction, $\theta(t)$ is also the antenna orientation. Define $j_{\max}$ as the index for the last transmission sensed in period k. Therefore, we can define the Monte Carlo motion planning problem for time k (or t^k) as,

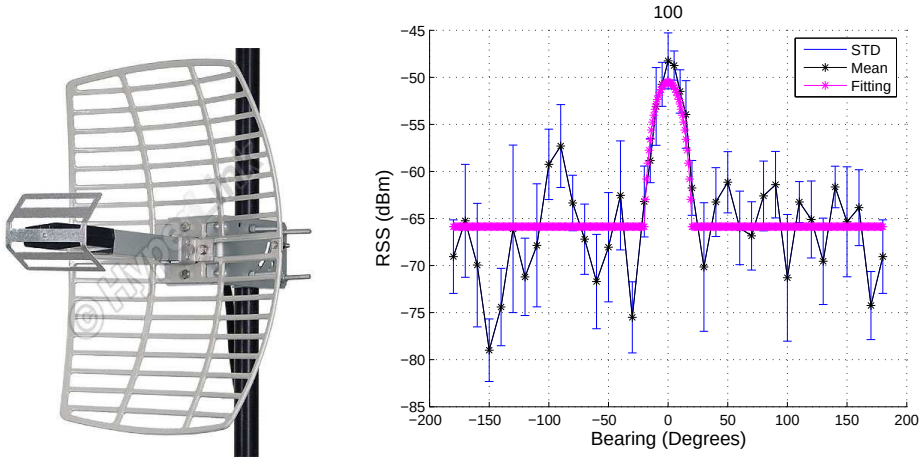
Problem 6 (Radio Source Localization Motion Planning)**.** *Given the current SPOG, which are sets $\{P(C_i|\mathbf{Z}(Z_{j_{\max}}^k))|i \in I\}$ and $\{P(C_i^1|\mathbf{Z}(Z_{j_{\max}}^k))|i \in I\}$, plan robot trajectory $\{\mathbf{r}(t)|t^k \le t < t^{k+1}\}$ that enables the robot to quickly localize radio sources.*

D. Modeling

1. Sensing Problem

We address the sensing problem first. The sensing problem actually has two components: an antenna model and an SPOG update process.

a. Antenna Model



(a) Antenna photo (b) Calibrated radiation pattern

Fig. 24.: HyperGain HG2415G parabolic directional antenna properties.

Figure 24 illustrates the antenna in our system. Bearing and distance are the two most important variables in an antenna model [77]. Let $(x_j^k, y_j^k, \theta_j^k)$ be the robot configuration when the j-th radio transmission in the k-th period is sensed. Let (x_i, y_i) be the cell center location. Define $d_{ij}^k = \sqrt{(x_j^k - x_i)^2 + (y_j^k - y_i)^2}$ as the distance from robot to the center of the cell. Let $\phi_{ij}^k = \text{atan2}(x_j^k - x_i, y_j^k - y_i) - \theta_j^k$ be the bearing of the cell with respect to the robot. Assume the active radio source is located in cell i, the expected RSS s_i of the directional antenna is approximated as $s_i = C \cdot (d_{ij}^k)^{-\beta} s(\phi_{ij}^k)$, where C is a constant depending on radio transmission power and $(d_{ij}^k)^{-\beta}$ is the signal

decay function. The directivity of the antenna is captured by the term $s(\phi_{ij}^k)$, which describes the radiation pattern of the antenna. We obtain $C = 1.77$ and the decay factor $\beta = 2.65$ for our antenna from calibration.

Since our receiver uses dBm as RSS unit, we have to take a $10 \log 10$ with respect to s_i,

$$z_0 = 10\big(\log_{10} C - \beta \log_{10} d_{ij}^k + \log_{10} s(\phi_{ij}^k)\big), \tag{5.2}$$

where z_0 is the expected RSS in units of dBm. From the antenna theory and the results from antenna calibration, we perform curve-fitting to obtain the radiation pattern function as illustrated Fig. 24b,

$$s(\phi_{ij}^k) = \begin{cases} \cos^2(4\phi_{ij}^k) & \text{if } -20^\circ \le \phi_{ij}^k \le 20^\circ, \\ \cos^2(80^\circ) & \text{otherwise.} \end{cases} \tag{5.3}$$

Note that the peak at the zero bearing in Fig. 24b is about 15 dBm higher than the average of non-peak regions. Although the data in Fig. 24b is obtained from the antenna calibration, the result conforms to antenna specifications well.

Eqs. (5.2) and (5.3) describe the expected RSS given that the radio transmission is from cell i. However, the sensed RSS is not a constant but a random variable due to the uncertainties in radio transmissions. Define Z_j^k as the sensed RSS. Therefore, the mean value of Z_j^k is z_0. From the antenna calibration, we know that Z_j^k conforms to the truncated normal distribution with a density function of $g(z) = \frac{\frac{1}{\sigma} f(\frac{z-z_0}{\sigma})}{F(\frac{z_{\max}-z_0}{\sigma}) - F(\frac{z_{\min}-z_0}{\sigma})}$, where the value of σ is 3.3 that is obtained from the antenna calibration, z is the sensed RSS, $f(\cdot)$ is the probability density function (PDF) of a normal distribution with zero mean and unit variance, $F(\cdot)$ is the cumulative distribution function (CDF) of $f(\cdot)$, and $z_{\min}$ and $z_{\max}$ are the minimum and the maximum RSS that the antenna can sense, respectively. Let $G(z) = \int_{z_{\min}}^{z} g(z)dz$ be the CDF of the truncated normal distribution.

Define $P(Z_j^k = z|C_i^1)$ as the conditional probability that the sensed signal strength is an integer z given cell i contains at least an active radio source. Since Z_j^k can only take integer values, $P(Z_j^k = z|C_i^1)$ actually is the overall antenna model,

$$P(Z_j^k = z|C_i^1) = G(z + 0.5) - G(z - 0.5). \tag{5.4}$$

b. Updating Probability Occupancy Grid

When a radio transmission with an RSS of z is sensed, we are interested in $P(C_i|Z_j^k = z)$, which is the conditional probability that cell i contains at least one radio source given the sensed RSS is z. According to (5.1), we have

$$P(C_i|Z_j^k = z) = P(C_i, C_i^1|Z_j^k = z) + P(C_i, C_i^0|Z_j^k = z).$$

Since event C_i^1 implies event C_i, the joint event (C_i, C_i^1) is the same as C_i^1. Hence,

$$P(C_i|Z_j^k = z) = P(C_i^1|Z_j^k = z) + P(C_i, C_i^0|Z_j^k = z). \tag{5.5}$$

According to Bayes' theorem,

$$P(C_i^1|Z_j^k = z) = \frac{P(Z_j^k = z|C_i^1)P(C_i^1)}{\sum_{i\in I} P(Z_j^k = z|C_i^1)P(C_i^1)}. \tag{5.6}$$

The second term $P(C_i, C_i^0|Z_j^k = z)$ in (5.5) is the joint conditional probability that there is at least one radio source in cell i and none of the radio sources in cell i transmits given the sensed RSS is z. Joint event (C_i, C_i^0) implies the following information:

- Since cell i is not transmitting, condition $Z_j^k = z$ cannot provide additional information for event C_i, which implies $P(C_i|Z_j^k = z) = P(C_i)$.
- There must be one active cell s, $s \in I$ and $s \neq i$.

- Joint conditional event $(C_i, C_i^0 | Z_j^k = z)$ is equivalent to the union of the collection of events $\{(C_i, C_s^1 | Z_j^k = z), s \neq i, s \in I\}$ because of no collision.
- Events C_i and C_s^1 are independent.

Therefore, we can obtain,

$$P(C_i, C_i^0 | Z_j^k = z) = P(C_i) \sum_{s \neq i, s \in I} P(C_s^1 | Z_j^k = z) \tag{5.7}$$

Note that $P(C_s^1 | Z_j^k = z)$ can be computed using (5.6). Plugging (5.6) and (5.7) into (5.5), we get,

$$P(C_i | Z_j^k = z) = \frac{\left(\begin{array}{l} P(Z_j^k = z | C_i^1) P(C_i^1) + \\ \qquad P(C_i) \sum_{s \neq i, s \in I} P(Z_j^k = z | C_s^1) P(C_s^1) \end{array} \right)}{\sum_{i \in I} P(Z_j^k = z | C_i^1) P(C_i^1)} \tag{5.8}$$

Unfortunately, (5.6) and (5.8) cannot be directly used in the system because $P(C_i)$ and $P(C_i^1)$ are not available. We have to rely on the conditional versions of $P(C_i)$ and $P(C_i^1)$ that build on the observation $\mathbf{Z}^-(Z_j^k)$. We can derive the following from (5.6) by adding $\mathbf{Z}^-(Z_j^k)$ as the condition,

$$P(C_i^1 | \{Z_j^k = z\} \cup \mathbf{Z}^-(Z_j^k)) = \frac{P(Z_j^k = z | C_i^1, \mathbf{Z}^-(Z_j^k)) P(C_i^1 | \mathbf{Z}^-(Z_j^k))}{\sum_{i \in I} P(Z_j^k = z | C_i^1, \mathbf{Z}^-(Z_j^k)) P(C_i^1 | \mathbf{Z}^-(Z_j^k))}. \tag{5.9}$$

Since the conditional event $Z_j^k = z$ is independent of the previous RSSs $\mathbf{Z}^-(Z_j^k)$ given C_i^1, we know $P(Z_j^k = z | C_i^1, \mathbf{Z}^-(Z_j^k)) = P(Z_j^k = z | C_i^1)$. According to the definition, $\{Z_j^k = z\} \cup \mathbf{Z}^-(Z_j^k) = \mathbf{Z}(Z_j^k)$. Eq. (5.9) can be rewritten as,

$$P(C_i^1 | \mathbf{Z}(Z_j^k)) = \frac{P(Z_j^k = z | C_i^1) P(C_i^1 | \mathbf{Z}^-(Z_j^k))}{\sum_{i \in I} P(Z_j^k = z | C_i^1) P(C_i^1 | \mathbf{Z}^-(Z_j^k))}. \tag{5.10}$$

Similarly, from (5.8), we can derive the following,

$$P(C_i|\mathbf{Z}(Z_j^k)) = \frac{\left(\begin{array}{l} P(Z_j^k = z|C_i^1)P(C_i^1|\mathbf{Z}^-(Z_j^k))+ \\ \quad P(C_i|\mathbf{Z}^-(Z_j^k))\times \\ \quad\quad \sum_{s\neq i, s\in I} P(Z_j^k = z|C_s^1)P(C_s^1|\mathbf{Z}^-(Z_j^k)) \end{array}\right)}{\sum_{i\in I} P(Z_j^k = z|C_i^1)P(C_i^1|\mathbf{Z}^-(Z_j^k))} \tag{5.11}$$

Eqs. (7.2) and (7.1) provide a recursive formulation for updating SPOG when a new radio transmission is sensed.

Eqs. (7.2) and (7.1) suggest that the update of the SPOG largely depends the antenna model $P(Z_j^k = z|C_i^1)$, which actually is a function of robot configurations. Since we threshold $P(C_i|\mathbf{Z}(Z_j^k))$ to determine if cell i contains at least a radio source, the convergence rate of the SPOG determines localization speed and accuracy. Hence, the convergence of the SPOG and the corresponding convergence speed really depend on the robot motion planning.

2. Robot Motion Planner

The intuition is to accelerate the rate that $P(C_i|\mathbf{Z}(Z_j^k)) \to 1$ for cells that contains radio sources with high probabilities through effective robot motions. Eq. (7.1) suggests that the update process contains two parts: $P(C_i|\mathbf{Z}(Z_j^k)) = P(C_i^1|\mathbf{Z}(Z_j^k)) + P(C_i, C_i^0|\mathbf{Z}(Z_j^k))$, where

$$P(C_i, C_i^0|\mathbf{Z}(Z_j^k)) = \frac{P(C_i|\mathbf{Z}^-(Z_j^k))\sum_{s\neq i, s\in I} P(Z_j^k = z|C_s^1)P(C_s^1|\mathbf{Z}^-(Z_j^k))}{\sum_{i\in I} P(Z_j^k = z|C_i^1)P(C_i^1|\mathbf{Z}^-(Z_j^k))}. \tag{5.12}$$

Since joint event (C_i, C_i^0) offers no more information regarding C_i, we ignore this part. Therefore, to increase the value of $P(C_i|\mathbf{Z}(Z_j^k))$, we want to increase $P(C_i^1|\mathbf{Z}(Z_j^k))$ as

much as possible. According to (7.2), this means

$$\max_{z} P(C_i^1|\mathbf{Z}^-(Z_j^k)). \tag{5.13}$$

We omit the process of deriving the optimal solution for (5.13) for brevity. Eq. (5.13) achieves its maximum when z is at its maximum. This means that the robot has to place its antenna's most sensitive region over the cell that has a high probability of containing radio sources.

Eqs. (5.2) and (5.3) suggest that the most sensitive region is located at zero bearing angle and at the nearest distance. Combining this, it is clear that the principle of the motion planning is to place the robot into the cells with the high $P(C_i|\mathbf{Z}(Z_j^k))$ values and force the robot to face these cells as much as possible. This principle inspires us to develop a Ridge Walking Algorithm (RWA) for the robot motion planning.

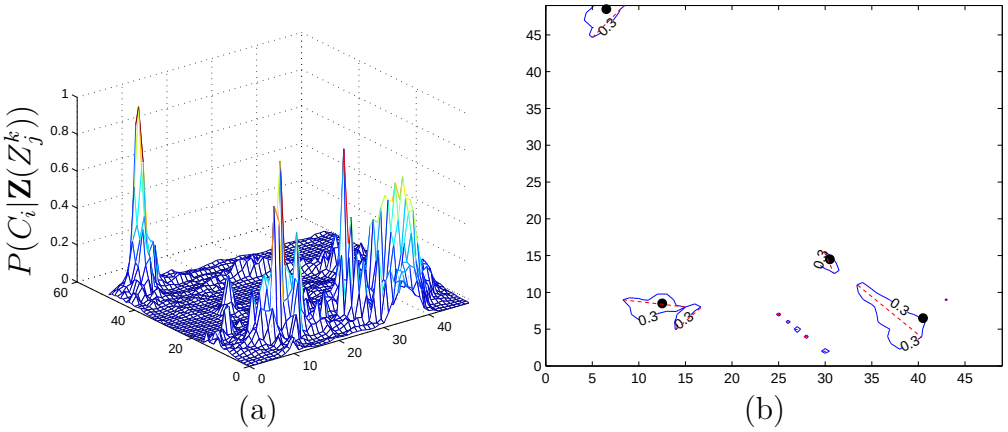


Fig. 25.: (a) An example of $P(C_i|\mathbf{Z}(Z_j^k))$ distribution, (b) Radio source locations, a sample level set $L(0.3)$, and ridges over a 50×50 grid for the case. The radio source locations are shown in black dots. Level set is bounded inside the blue solid lines. The red dashed lines are the corresponding ridges for the level set components.

Fig. 25a illustrates an example of the distribution of $P(C_i|\mathbf{Z}(Z_j^k))$ over a $50 \times$

50 grid. The actual radio source positions are shown as black dots in Fig. 25b. $P(C_i|\mathbf{Z}(Z_j^k))$ value is much larger in the area adjacent to radio sources than that of other areas. To study the spatial distribution of $P(C_i|\mathbf{Z}(Z_j^k))$, we introduce level set $L(p)$, $p \in (0, 1]$ as follows,

$$L(p) = \{i|P(C_i|\mathbf{Z}(Z_j^k)) \geq p, i \in I\}. \tag{5.14}$$

Let us envision that a plane parallel to the ground plane intersects the mountain-like $P(C_i|\mathbf{Z}(Z_j^k))$ distribution at height p in Fig. 25a. The intersection generates $L(p)$ which contains all cells with $P(C_i|\mathbf{Z}(Z_j^k))$ above the plane. Fig. 25b illustrates the level set $L(0.3)$ for the example in Fig. 25a.

Fig. 25b also shows that $L(p)$ usually consists of several disconnected components. Define $l_{\max}$ as the total number of the disconnected components and L_l as the l-th component, $l = 1, ..., l_{\max}$. Therefore, $L(p) = L_1 \cup L_2 \cup ... \cup L_{l_{\max}}$, and $L_l \cap L_m = \emptyset$, where $m \neq l$ and $m = 1, 2, .., l_{\max}$. For the l-th component, we define its ridge R_l as the line segment defined by points (x', y') and (x'', y'') on L_l,

$$\begin{aligned} R_l = \{(x, y)|x = (1 - \alpha)x' + \alpha x'', \\ y = (1 - \alpha)y' + \alpha y'', \alpha \in [0, 1]\}, \end{aligned} \tag{5.15}$$

where points (x', y') and (x'', y'') are the two points on L_l such that the distance between (x', y') and (x'', y'') is the maximum.

If the robot walks on the ridge, the probability that the robot is close to a potential radio source is very high. Due to the walking direction, the antenna is always pointed along the ridge, which ensures the most sensitive reception region of the antenna to overlap with the l-th component. In the RWA algorithm, there are two types of robot motion: on-ridge movements and off-ridge movements. Since the on-

ridge movement is the effective movement for the localization purpose, it is desirable for the robot to allocate its time to on-ridge movements as much as possible. The off-ridge movement refers to the travel in-between ridges for the robot. Since we have a fixed time period, we set the robot to travel at its fastest speed along the shortest path for off-ridge movements to save time for on-ridge movements.

Since each ridge is usually short, we can approximate each ridge as a vertex. We define edges as the line segments connecting different vertices on the 2D plane. With a vertex set V, an edge set E and a graph $G(V, E)$, to find the shortest path for the off-ridge movement is an instance of the traveling salesman problem (TSP) problem. Although the decision version of the planar TSP problem is NP-complete, we can use the 3-opt heuristics to solve it [78]. If a better approximation result is needed, we can use other approximation algorithms[79]. Those algorithms give us a close to the shortest off-ridge movement trajectory. Define $v_{\max}$ as the maximum velocity that the robot can travel. The time available for on-ridge movements t_{ON} is,

$$t_{\text{ON}} = \tau_0 - d_{\text{OFF}}/v_{\max}, \tag{5.16}$$

where d_{OFF} is the total length of off-ridge edges. We allocate t_{ON} to each ridge proportional to the probability that the corresponding component contains a radio source. For component l, we define the time the robot spend on the ridge R_l as τ_l. Therefore,

$$\tau_l = \frac{\sum_{i \in L_l} P(C_i|\mathbf{Z}(Z_j^k))}{\sum_{i \in L(p)} P(C_i|\mathbf{Z}(Z_j^k))} t_{\text{ON}}. \tag{5.17}$$

With τ_l and the length of each ridge, it is trivial to find the robot velocity for the ridge.

E. Algorithms

To summarize our analysis, we present two algorithms including an SPOG update algorithm and the RWA. Corresponding to the sensing problem in Section 2, the SPOG update algorithm runs when a radio signal is detected. Define set $\mathbb{C}^*$ as the set of cells that contain radio sources with initial value $\mathbb{C}^* = \emptyset$. Define p_t as the probability threshold for finding the radio source. The robot reports the cells that satisfy $P(C_i|\mathbf{Z}(Z_j^k)) > p_t$ as the cells that contain at least one radio source. Recall

Algorithm 3: SPOG Update Algorithm

```
input  : the received RF signal strength Z_j^k = z
output: P(C_i|Z(Z_j^k)), P(C_i^1|Z(Z_j^k)), i ∈ I, and C*
for i ∈ I do                                                    O(n)
    Compute distance d_ij^k and φ_ij^k                          O(1)
    Compute radiation pattern s(φ_ij^k) using (5.3)             O(1)
    Compute z_0 using (5.2)                                     O(1)
    Compute g(z) and G(z)                                       O(1)
    Compute P(Z_j^k = z|C_i^1) using (5.4)                      O(1)
end
for i ∈ I do                                                    O(n)
    Compute P(C_i^1|Z(Z_j^k)) using (7.2)                       O(n)
    Compute P(C_i|Z(Z_j^k)) using (7.1)                         O(n)
    if P(C_i|Z(Z_j^k)) > p_t and i ∉ C* then
        C* = C* ∪ {i}                                           O(1)
    end
end
```

that n is the total number of cells. It is clear that the SPOG update algorithm runs $O(n^2)$. The initial value settings are $P(C_i|\mathbf{Z}(Z_0^0)) = 0$ and $P(C_i^1|\mathbf{Z}(Z_0^0)) = 1/n$.

The RWA algorithm runs every τ_0 time. As illustrated in Algorithm 4, the robot performs random walking until set $L(p) \neq \emptyset$ at the initialization stage. Then the robot switches into the normal ridge walking mode. The robot stops when no additional radio source has been found in $k_{\max}$ consecutive periods where $k_{\max}$ is a preset iteration number. Algorithm 4 uses exhaustive search to find the exact TSP

tour. The overall complexity is $O(n + (l_{\max} - 1)!)$. Although the 3-opt heuristic can accelerate the computation of the TSP, it cannot change the worst case complexity.

Algorithm 4: Ridge Walking Algorithm

input : $P(C_i\|\mathbf{Z}(Z_j^k))$, $P(C_i^1\|\mathbf{Z}(Z_j^k))$, $i \in I$	
output: Robot motion $\{\mathbf{r}(t)\|t^k \le t < t^{k+1}\}$ and $\mathbb{C}^*$	
Compute $L(p)$	$O(n)$
if $L(p) = \emptyset$ **then**	
$\{\mathbf{r}(t)\|t^k \le t < t^{k+1}\}$ = random walk	$O(1)$
end	
else	
Find all disconnected components in $L(p)$	$O(n)$
Compute R_l for each L_l	$O(n)$
Construct graph G and solve TSP	$O((l_{\max} - 1)!)$
Compute d_{OFF}	$O(l_{\max})$
Compute t_{ON} using (5.16)	$O(1)$
Compute τ_l for each ridge using (5.17)	$O(1)$
Output robot motion $\{\mathbf{r}(t)\|t^k \le t < t^{k+1}\}$	$O(1)$
end	

F. Experiments

We have implemented the algorithms and the simulation platform using Microsoft Visual C++ .NET 2005 with OpenGL on a PC Desktop with an Intel 2.13GHz Core 2 Duo CPU, 2GB RAM, and Windows XP. The algorithms are tested in the simulation. The radio sources are XBeeT with ZigBeeT/802.15.4 OEM radio frequency Modules by MaxStream, Inc. The antenna is calibrated first with the radio sources. The calibration is conducted at 328 configurations and 6560 readings have been collected. We use the data from the real hardware to drive the simulation experiments below.

The grid is a square with 50×50 cells. Each grid cell has a size of 5.08×5.08 cm^2. Each radio source generates radio transmission signals according to an independently and identically distributed Poisson process with a rate of $\lambda = 0.012$ packets per

second. The threshold $p_t = 0.8$ and the level set parameter $p = \frac{6}{n}\sum_i P(C_i|\mathbf{Z}(Z_j^k))$, where the constant 6 is determined by many experimental trials. During each trial of the simulation, we randomly generate radio source locations in the 50×50 grid.

Fig. 26a illustrates how $P(C_i|\mathbf{Z}(Z_j^k))$ converges at the radio source for a sample case with six radio sources. The location of the six radio sources is shown in Fig. 25b. It is clear that $P(C_i|\mathbf{Z}(Z_j^k))$ grows monotonically toward 1. This is what we expect to see: $P(C_i|\mathbf{Z}(Z_j^k)) \to 1$ for cells contains radio sources.

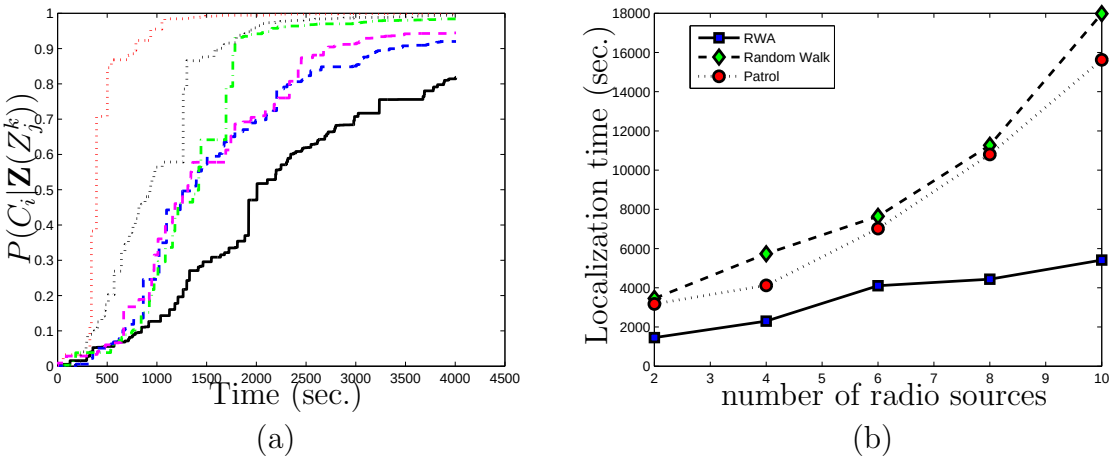


Fig. 26.: (a) Convergence of $P(C_i|\mathbf{Z}(Z_j^k))$ at radio source locations for a six-radio source case. (b) Localization performance comparison among the RWA, the random walk, and the fixed-route patrol.

We also compare our algorithms to a random walk and a fixed-route patrol. The random walk is chosen because it is considered as the most conservative approach which covers the entire field in long run. The fixed-route patrol traverses the field using a pre-defined route. It is considered as energy efficient but might not treat each cell equally due to the route selection. We increase the radio source number from 2 to 10 to observe the performance of each method. For each trial, we test all

three methods. We repeat for 10 trials for each radio source number and compute the average time required for localizing all radio sources. Fig. 26b illustrates comparison results. It is clear that the RWA significantly outperforms the two heuristics. It is also surprising that the fixed route patrol is no much better than the random walk. However, the result can be explained that the robot motion for the two heuristics does not consider sensor location distribution and hence cannot achieve good performance.

G. Conclusions

We report our system and algorithm developments that enable a mobile robot equipped with a directional antenna to localize unknown transient radio sources. We modeled the radio transmission activities using an SPOG and proposed an SPOG update algorithm and an RWA algorithm for robot motion planning. We tested the algorithm using simulation with the data from the real hardware. In the experiment, we compared our algorithms with a random walk and a fixed-route patrol heuristics. Our algorithms showed a consistently superior performance over the two heuristics.

Chapter VI extends the multiple targets localization problem using multiple robots.

CHAPTER VI

CENTRALIZED MRMT LOCALIZATION

A. Introduction

Imagine that a team of mobile robots is searching for a sensor network deployed by enemies (see Fig. 27). The transmission power of the radio targets is unknown and may vary from time to time. In this case, using a single robot is not visible to resolve the multiple targets localization problem due to unknown transmission power and distance. This leads to cooperation of multiple robots. A new method is needed for this multiple targets localization problem that is coupled with issues in signal correspondence, variable source transmission power, and robot sensing range limit. For simplicity, we assume that each robot has the unlimited communication range so that all robots are coordinated using a centralized control.

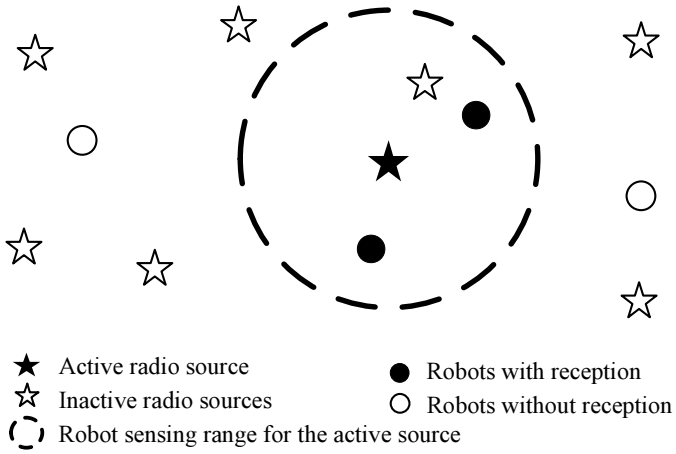


Fig. 27.: An example of a localization scenario.

When signal sources are not cooperative, RSS readings are the primary information for localization because RSS attenuates over distance. Since signal transmission power at the source is not available, ratios between RSS readings from dislocated lis-

teners have been proven to be effective. This approach focuses on localizing a single source at a time and hence are not concerned with the signal correspondence issue in their setup. The existing work can be viewed as the best case scenarios for our problem, which inspires our sensing model development.

B. Problem Definition

1. Problem Scenario

Both robots and radio sources reside in a 2D Euclidean space. To formulate the localization problem, we make the following assumptions:

1. Each robot is equipped with an omni-directional antenna with a limited sensing range.

2. All robots are coordinated using a centralized control.

3. The unknown network traffic is light and each target radio transmission is short, which are the typical characteristics of a low power sensor network.

4. Transmission powers of radio sources are unknown to the robots and may change from time to time. However, locations of radio sources do not change.

2. Spatiotemporal Probability Occupancy Grid

To infer the transmitter locations and transmission rates based on perceived signals, we use a Bayesian framework to keep track of the knowledge of unknown radio sources. Here we extend the SPOG proposed in Chapter V. SPOG partitions the searching region into small and equal-sized grid cells. Define $i \in \mathcal{N}_i$ as the cell index variable where $\mathcal{N}_i := \{1, ..., n\}$ is the grid cell index set and n is the total number of cells. SPOG tracks two types of probabilistic events: C_i represents the event that cell i

contains a radio source and C_i^1 represents the event that cell i is the active source when a transmission is detected. Define $P(C)$ as the probability for event C. $P(C_i)$ and $P(C_i^1)$ characterize spatiotemporal behaviors of transient radio sources. Note that we ignore collision cases because robots sense the radio signal strength (RSS) as soon as the transmission is initiated and the probability of two or more transmissions initiated at the exact same moment is negligible in a light traffic network.

Let $l \in \mathcal{M} := \{1, ..., m\}$ be the robot index variable where m is the total number of robots and $\mathcal{M}$ is the robot index set. Discrete time k refers to each moment when a transmission is detected by robots. et the discrete random variable $\tilde{Z}_l^k \in [1, 255] \cap \mathbb{N}$ be the sensed RSS reading (from an 8-bit receiver) of the l-th robot at time k. Define $\tilde{\mathbf{Z}}^k = [\tilde{Z}_1^k, ..., \tilde{Z}_m^k]^T$ as a discrete random vector of all the sensed RSS readings at time k and let $\tilde{\mathbf{z}}^k := [\tilde{z}_1^k, ..., \tilde{z}_m^k]^T$ be corresponding values. As a convention, we use lower cases of random variables or vectors to denote their values.

At time k, event $\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k$ is perceived by robots. The posterior probability $P(C_i|\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k)$ over the grid needs to be updated. According to Chapter V, this is actually a nested multivariate Bayesian process,

$$P(C_i|\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k) = \frac{\left(\begin{array}{l} P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)P(C_i^1) + \\ \qquad P(C_i) \sum_{s \neq i, s \in I} P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_s^1)P(C_s^1) \end{array} \right)}{\sum_{i \in I} P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)P(C_i^1)}, \tag{6.1}$$

$$P(C_i^1|\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k) = \frac{P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)P(C_i^1)}{\sum_{i \in I} P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)P(C_i^1)}, \tag{6.2}$$

where $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)$ is the sensing model. Eqs. (7.1) and (7.2) can be easily modified to an incremental conditional format for recursive update in Chapter V. As more RSS readings enter the system over time, $P(C_i|\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k)$ converges and allows robots to

localize each radio source.

3. Problem Formulation

To utilize the Bayesian framework, we need to derive a sensing model first:

Definition 3 (Sensing Problem)**.** *Derive* $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k | C_i^1)$ *for present time* k *when a new RSS reading is received.*

Once $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k | C_i^1)$ is obtained, we can use (7.1) and (7.2) to compute posterior sensor location distribution $P(C_i | \tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k)$, which leads to robot trajectory planning,

Definition 4 (Planning Problem)**.** *Given the updated* $P(C_i | \tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k)$*, plan trajectories for each robot at the beginning of each planning period.*

We start with the sensing problem first in Section C.

C. Sensing Model

The sensor model $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k | C_i^1)$ is very complex. It is a joint conditional distribution of an m-dimensional random vector. To derive the conditional probability, we model the signal transmission uncertainty, derive pairwise sensing model based on signal strength ratio to remove the dependence on source transmission power, and propose a sensing fusion scheme to aggregate the output of all pairs to obtain the high order model $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k | C_i^1)$. For simplicity, the time superscript k is dropped in this section by assuming that all values correspond to present time k. Thus, $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k | C_i^1)$ becomes $P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}} | C_i^1)$.

1. Signal Propagation Model

For a robot equipped with an omni-directional antenna, the distance to the active radio source and source transmission power largely determine the perceived RSS.

Assume the active radio source is located at the center of cell i. Let $\mathbf{x}_i = [x_i, y_i]^T$ and $\mathbf{x}_l = [x_l, y_l]^T$ be the center location of cell i and the location of robot l, respectively, when the transmission is sensed. Define $d_{li} = \| \mathbf{x}_l - \mathbf{x}_i \|$ as the Euclidean distance between $\mathbf{x}_l$ and $\mathbf{x}_i$. Following the signal propagation model [80], the expected RSS of robot l is denoted as ψ_l and measured in units of dBm:

$$\psi_l = w_i - 10\beta \log_{10}(d_{li}), \tag{6.3}$$

where source power level w_i is unknown and β is the signal decay factor.

An RSS level is not a constant but a continuous random variable due to uncertainties in transmissions. Assume the robot radio listener has an infinite resolution, its perceived RSS would be a continuous random variable Z_l for robot l. Moreover, robots can only detect the transmission signal if active radio source is located in their sensing ranges, each of which is determined by an RSS threshold denoted by ζ. To characterize sensing range limit and background noises in sensing, we have

$$Z_l = \mu_l + \omega_l, \text{ where } \mu_l = \begin{cases} \psi_l, & \text{if } \tilde{z}_l > \zeta \\ \zeta, & \text{otherwise,} \end{cases} \tag{6.4}$$

where ω_l follows the independent and identically distributed (*i.i.d.*) Gaussian with zero mean and a variance of σ^2. Note that β in (6.3) and σ^2 can be obtained by calibration. Therefore, the probability density function (PDF) of $Z_l|C_i^1$ is $f_{Z_l|C_i^1}(z_l) = Bel(\mu_p, \sigma^2)$, where $Bel(\mu_p, \sigma^2)$ is the Gaussian PDF. As a convention, the subscript of $f(\cdot)$ is the corresponding random variable of the PDF function.

Actually, the sensed RSS reading $\tilde{Z}_l$ is an integer due to receiver hardware limit. As a convention, we use $\tilde{a}$ to indicate the integer value of continuous variable a. Define

$\mathcal{I}_l$ as an RSS interval,

$$\mathcal{I}_l = (\tilde{z}_l - 0.5, \tilde{z}_l + 0.5] \subset \mathbb{R}. \tag{6.5}$$

Thus, we have the relation between $\tilde{Z}_l$ and Z_l given C_i^1,

$$P(\tilde{Z}_l = \tilde{z}_l | C_i^1) = P(Z_l \in \mathcal{I}_l | C_i^1) = \int_{z_l \in \mathcal{I}_l} f_{Z_l|C_i^1}(z_l) dz_l. \tag{6.6}$$

This is actually the sensing model when there is only one robot. Since this model relies on unknown source power level w_i, it is not a viable sensing model, but provides a foundation for the next step.

2. Transmission Power Independent Pairwise Sensing

For a robot pair (p, q), $p \neq q$, recall the possible RSS readings form sets $\mathcal{I}_p$ and $\mathcal{I}_q$ as defined in (6.5), respectively. According to our convention, $P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q | C_i^1)$ is a pairwise conditional probability given C_i^1. We are now ready to show that $P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q | C_i^1)$ can be obtained from its RSS ratio regardless of source transmission power levels.

Define $Z_{p-q} := Z_p - Z_q$ and let $\mathcal{I}_{p-q} = (\tilde{z}_p - \tilde{z}_q - 1, \tilde{z}_p - \tilde{z}_q + 1] \subset \mathbb{R}$ be the interval of Z_{p-q} values. $P(Z_{p-q} \in \mathcal{I}_{p-q} | C_i^1)$ denotes the probability of pairwise difference given C_i^1. We have the following Lemma with its proof in Appendix G.

Lemma 2.

$$P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q | C_i^1) = \frac{1}{\eta_{pq}} P(Z_{p-q} \in \mathcal{I}_{p-q} | C_i^1), \tag{6.7}$$

where η_{pq} is the normalizing factor.

It is worth noting, since the RSS readings are in log scale, the difference between the two readings Z_{p-q} actually means a RSS ratio which does not depend on source

transmission power levels. Computing $P(Z_{p-q} \in \mathcal{I}_{p-q}|C_i^1)$ is nontrivial because some of robots may not have readings due to limited sensing ranges. Based on (6.4), the robot index set $\mathcal{M}$ is partitioned into two disjoint sets $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_0$ which correspond to the sets of robots with and without reception, respectively. As a result, we have three types of pairs: no detection for either robot, single detection, and dual detection. Define $\mathcal{E}$ as the set for all possible pairs which consists of three disjoint subsets $\mathcal{E} = \mathcal{E}_{11} \cup \mathcal{E}_{10} \cup \mathcal{E}_{00}$ where

$$\begin{aligned}
\mathcal{E}_{11} &= \{(p,q)|p<q, p \in \mathcal{M}_1, q \in \mathcal{M}_1\}, \\
\mathcal{E}_{10} &= \{(p,q)|p \in \mathcal{M}_1, q \in \mathcal{M}_0\}, \\
\mathcal{E}_{00} &= \{(p,q)|p<q, p \in \mathcal{M}_0, q \in \mathcal{M}_0\}.
\end{aligned} \tag{6.8}$$

Define Z_{p-q}^{11}, Z_{p-q}^{10} and Z_{p-q}^{00} as the sensor readings of the robot pair (p,q) corresponding to components of $\mathcal{E}_{11}$, $\mathcal{E}_{10}$ and $\mathcal{E}_{00}$, respectively. Z_{p-q} in (6.7) will be one of these three types. Note that $P(Z_{p-q}^{00} \in \mathcal{I}_{p-q}|C_i^1)$ is a constant because it provides no information due to no reception. We now focus on deriving $P(Z_{p-q}^{11} \in \mathcal{I}_{p-q}|C_i^1)$ and $P(Z_{p-q}^{10} \in \mathcal{I}_{p-q}|C_i^1)$.

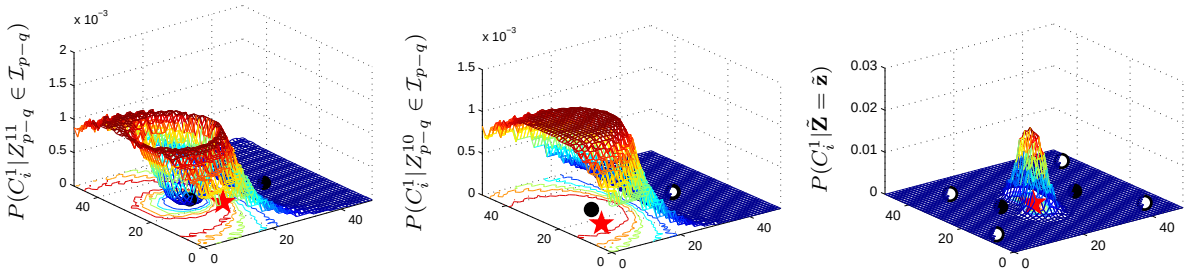
Let us compute $P(Z_{p-q}^{11} \in \mathcal{I}_{p-q}|C_i^1)$ first. From (6.3) and (6.4), the mean value $(\mu_p - \mu_q)$ of Z_{p-q}^{11} becomes

$$\mu_p - \mu_q = \psi_p - \psi_q = 10\beta \log_{10} \frac{d_{qi}}{d_{pi}}, \tag{6.9}$$

and the PDF of $Z_{p-q}^{11}|C_i^1$ is

$$f_{Z_{p-q}^{11}|C_i^1}(z_{p-q}^{11}) = Bel\left(10\beta \log_{10} \frac{d_{qi}}{d_{pi}}, 2\sigma^2\right). \tag{6.10}$$

Thus, we have the following lemma.



(a) (b) (c)

Fig. 28.: Sample cases of posterior condition distributions of signal source location given that $P(C_i^1)$ initially uniform across cells (a) dual detection, (b) single detection, and (c) fusion of all pairs. The red star is the active radio source location. This is obtained using $P(Z_{p-q} \in \mathcal{I}_{p-q}|C_i^1)$ and the Bayesian framework in (7.2). The grid size is 50×50. Black and white dots represent robots with and without readings, respectively.

Lemma 3.

$$P(Z_{p-q}^{11} \in \mathcal{I}_{p-q}|C_i^1) = \int_{\tilde{z}_p-\tilde{z}_q-1}^{\tilde{z}_p-\tilde{z}_q+1} f_{Z_{p-q}^{11}|C_i^1}(z)dz$$
$$= \left[F_{Z_{p-q}^{11}|C_i^1}(\tilde{z}_p - \tilde{z}_q + 1) - F_{Z_{p-q}^{11}|C_i^1}(\tilde{z}_p - \tilde{z}_q - 1)\right], \tag{6.11}$$

where $F_{Z_{p-q}^{11}|C_i^1}(\cdot)$ *is the cumulative distribution function of* $f_{Z_{p-q}^{11}|C_i^1}(\cdot)$.

To facilitate the understanding of the dual detection case, Fig. 28(a) shows an example to illustrate the corresponding posterior probability $P(C_i^1|Z_{p-q}^{11} \in \mathcal{I}_{p-q})$.

For $P(Z_{p-q}^{10} \in \mathcal{I}_{p-q}|C_i^1)$, we have the following result.

Lemma 4.

$$P(Z_{p-q}^{10} \in \mathcal{I}_{p-q}|C_i^1) = \frac{1}{\eta^{10}}\big(1 - \int_{\tilde{z}_p-\tilde{z}_q}^{\tilde{z}_p-\tilde{z}_q+1} F_{Z_{p-q}^{11}|C_i^1}(z)dz\big), \tag{6.12}$$

where η^{10} *is the normalizing factor.*

The proof of Lemma 4 is in Appendix H. This result also does not depend on source transmission power. As an example, Fig. 28(b) illustrates the corresponding posterior probability $P(C_i^1|Z_{p-q}^{10} \in \mathcal{I}_{p-q})$.

3. Sensor Fusion of Multiple Pairs

Now we are ready to show that the m-dimensional joint conditional probability $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)$ can be reduced to a combination of pairwise conditional probabilities $P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q|C_i^1)$. We have the following lemma.

Lemma 5.

$$P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}|C_i^1) = \frac{1}{\eta} \prod_{(p,q)\in\mathcal{E}} P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q|C_i^1), \tag{6.13}$$

where η is the normalizing factor and remains the same for all p and q values.

Proof. According to (6.4), $\tilde{Z}_1, ..., \tilde{Z}_m$ are independent given C_i^1, and

$$P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}|C_i^1) = \prod_{l=1}^{m} P(\tilde{Z}_l = \tilde{z}_l|C_i^1). \tag{6.14}$$

The individual conditional probability $P(\tilde{Z}_l = \tilde{z}_l|C_i^1)$ can be paired up as:

$$\begin{aligned}
P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}|C_i^1) &= \prod_{l=1}^{m} P(Z_l \in \mathcal{I}_l|C_i^1) \\
&= \prod_{l=1}^{m} \left(\frac{P(Z_l \in \mathcal{I}_l|C_i^1)^{m-1}}{P(Z_l \in \mathcal{I}_l|C_i^1)^{m-2}} \right) \\
&= \frac{1}{\prod_{l=1}^{m} P(Z_l \in \mathcal{I}_l|C_i^1)^{m-2}} \\
&\times \prod_{p=1}^{m-1} \prod_{q=p+1}^{m} P(Z_p \in \mathcal{I}_p|C_i^1) P(Z_q \in \mathcal{I}_q|C_i^1) \\
&= \frac{1}{\eta} \prod_{(p,q)\in\mathcal{E}} P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q|C_i^1),
\end{aligned} \tag{6.15}$$

where $\eta = \prod_{l=1}^{m} P(Z_l \in \mathcal{I}_l|C_i^1)^{m-2}$ remains the same for all p and q values. □

Now, we can complete the sensor model $P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}|C_i^1)$. Combining Lemmas 2– 5, we have the following theorem.

Theorem 2. *The high dimension joint conditional probability sensing model* $P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}|C_i^1)$ *can be decomposed as a combination of pairwise conditional probabilities,*

$$P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}|C_i^1) = \frac{1}{\eta''} \prod_{(p,q)\in\mathcal{E}_{11}} \begin{pmatrix} F_{Z_{p-q}^{11}|C_i^1}(\tilde{z}_p - \tilde{z}_q + 1) \\ -F_{Z_{p-q}^{11}|C_i^1}(\tilde{z}_p - \tilde{z}_q - 1) \end{pmatrix} \times \prod_{(p,q)\in\mathcal{E}_{10}} \Big(1 - \int_{\tilde{z}_p - \tilde{z}_q}^{\tilde{z}_p - \tilde{z}_q + 1} F_{Z_{p-q}^{11}|C_i^1}(z)dz\Big), \tag{6.16}$$

where η'' *is the normalizing factor and remains the same for all* p *and* q *values.*

Proof. Combining Lemma 2 with Lemma 5, the sensing model becomes

$$\begin{aligned} P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}|C_i^1) &= \frac{1}{\eta} \prod_{(p,q)\in\mathcal{E}} \frac{1}{\eta_{pq}} P(Z_{p-q} \in \mathcal{I}_{p-q}|C_i^1) \\ &= \left(\frac{1}{\eta} \prod_{(p,q)\in\mathcal{E}} \frac{1}{\eta_{pq}}\right) \prod_{(p,q)\in\mathcal{E}} P(Z_{p-q} \in \mathcal{I}_{p-q}|C_i^1) \\ &= \frac{1}{\eta'} \prod_{(p,q)\in\mathcal{E}} P(Z_{p-q} \in \mathcal{I}_{p-q}|C_i^1), \end{aligned} \tag{6.17}$$

where $\eta' = \eta \prod_{(p,q)\in\mathcal{E}} \eta_{pq}$ is the normalizing factor and remains the same for all p and q values.

Applying (6.8) to (6.17) and combining Lemmas 3 and 4, the sensing model is

rewritten as

$$
\begin{aligned}
P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}|C_i^1) = \frac{1}{\eta^{'}} & \prod_{(p,q)\in\mathcal{E}_{11}} P(Z_{p-q}^{11} \in \mathcal{I}_{p-q}|C_i^1) \\
& \times \prod_{(p,q)\in\mathcal{E}_{10}} P(Z_{p-q}^{10} \in \mathcal{I}_{p-q}|C_i^1) \\
& \times \prod_{(p,q)\in\mathcal{E}_{00}} P(Z_{p-q}^{00} \in \mathcal{I}_{p-q}|C_i^1) \\
= \frac{1}{\eta^{''}} & \prod_{(p,q)\in\mathcal{E}_{11}} \begin{pmatrix} \mathrm{F}_{Z_{p-q}^{11}|C_i^1}(\tilde{z}_p - \tilde{z}_q + 1) \\ -\mathrm{F}_{Z_{p-q}^{11}|C_i^1}(\tilde{z}_p - \tilde{z}_q - 1) \end{pmatrix} \\
& \times \prod_{(p,q)\in\mathcal{E}_{10}} \Big(1 - \int_{\tilde{z}_p - \tilde{z}_q}^{\tilde{z}_p - \tilde{z}_q + 1} \mathrm{F}_{Z_{p-q}^{11}|C_i^1}(z)dz\Big), \qquad (6.18)
\end{aligned}
$$

where $\eta^{''} = \frac{\eta' \prod_{(p,q)\in\mathcal{E}_{10}} \eta^{10}}{\prod_{(p,q)\in\mathcal{E}_{00}} P(Z_{p-q}^{00} \in \mathcal{I}_{p-q}|C_i^1)}$ is the normalizing factor. □

Again, Fig. 28(c) illustrates the corresponding posterior probability $P(C_i^1|\tilde{\mathbf{Z}} = \tilde{\mathbf{z}})$, which is the fusion of all pairs. It is desirable that the adjacent regions of the red star have higher probabilities than that of other regions.

D. Robot Motion Planner

Theorem 2 summarizes how to compute $P(\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}|C_i^1)$. With the sensing model, the Bayesian framework in (7.2) can derive the posterior source location distributions $P(C_i|\tilde{\mathbf{Z}} = \tilde{\mathbf{z}})$. The next step is to develop a multi-robot motion planner that enables robots to quickly localize radio sources using the SPOG. We build on the ridge walking algorithm (RWA) in Chapter V. RWA has been designed for a single robot without sensing range limit to localize multiple radio sources. The experimental results have shown that it is an efficient framework. However, RWA is not designed for multiple robots and significant revisions are needed. Let us begin with a brief review of RWA.

RWA uses a probability threshold plane that intercepts $P(C_i|\tilde{\mathbf{Z}} = \tilde{\mathbf{z}})$ to generate

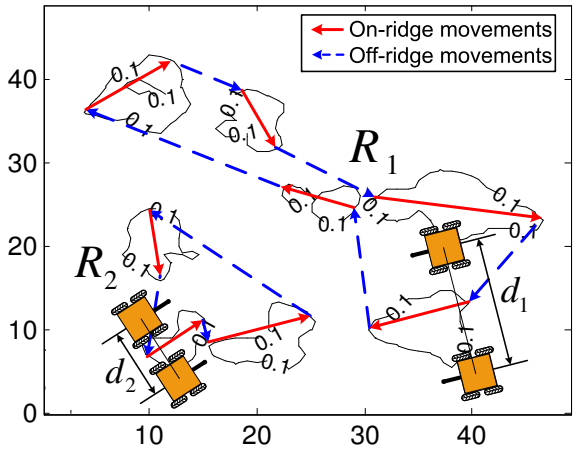


Fig. 29.: An illustration of level sets with probability threshold of 0.1, ridges, and Pairwise Ridge Walking Algorithm with two pairs of robots over a 50×50 grid.

level sets that enclose all cells with $P(C_i|\tilde{\mathbf{Z}} = \tilde{\mathbf{z}})$ no less than the threshold. The irregular closed curves in Fig. 32a are examples of level sets. Ridges are created by extracting the longest dimension of each isolated level set. The red line segments in Fig. 32a are ridges. In RWA, a 3-opt heuristics algorithm is employed to compute an Euclidean traveling salesperson (TSP) tour for the single robot that must include all ridges. The TSP tour is partitioned into on-ridge and off-ridge segments. For off-ridge segments, the robot moves at its fastest speed. For on-ridge segments, the robot spends the time proportional to the summation of posterior conditional probability $P(C_i|\tilde{\mathbf{Z}} = \tilde{\mathbf{z}})$ over the corresponding isolated level set on each ridge. This means that the robot spends more time in high probability regions, which increases the localization efficiency.

Since we have more than one robot, we need many sub tours instead of a single TSP tour. We pair up robots and treat a pair of robots as a super robot. Assuming m is an even number, we have $m/2$ super robots. Therefore, we need to partition the TSP tour into $m/2$ sub tours and assign each super robot to a sub tour. The

partition is based on k-means clustering algorithm [81] with $m/2$ as the cluster number to cluster ridge sets. For each cluster, we again use a 3-opt heuristics algorithm to find the TSP and the rest of RWA follows. Hence, we name this approach pairwise ridge walking algorithm (PRWA).

The remaining issue is how to determine the distance between each paired robots. Comparing Fig. 28(a) and (b), we notice that the dual detection case provides more information (less uncertainty) about radio source locations than the single detection case does. The spatial information contained in a distribution can be measured by the Shannon entropy in information theory. In order to choose the best distance d_u^* between the u-th pair, we formulate this problem by minimizing the Shannon entropy.

Define $\mathcal{S}_u$ as the set of cells in the isolated level set that correspond to the ridge cluster $\mathbf{R}_u$. Let cell $v \in \mathcal{S}_u$. Assume that the radio source $\mathbf{x}_v = [x_v, y_v]^T$ is located at the center of cell C_v by ignoring the minor intra-cell difference. Define $\hat{z}_{lv}^w$ as the mean RSS reading at robot l. We have,

$$\hat{z}_{lv}^w = w - 10\beta \log_{10}(d_{lv}), \tag{6.19}$$

where $w \in [w_{min}, w_{max}]$ is the unknown source transmission power which varies from w_{min} to w_{max}.

Define $\hat{\mathbf{Z}}_v^w = [\hat{Z}_{pv}^w, \hat{Z}_{qv}^w]^T$ as the RSS readings for the robot pair. Define $\mathbf{r}_u(t)$ as the center position of the robot pair at time t. We know $\mathbf{r}_u(t)$ because PRWA provides the trajectory for the super robot using the center position of the robot pair as the position on the trajectory. Denote $P(C_i|\hat{\mathbf{Z}}_v^w = \hat{\mathbf{z}}_v^w, \mathbf{r}_u(t), d_u)$ as the posterior probability that cell i contains a radio source given $\hat{\mathbf{z}}_v^w$, $\mathbf{r}_u(t)$ and d_u. Define $H(t, w, v, d_u)$ as the Shannon entropy over the probability distribution $P(C_i|\hat{\mathbf{Z}}_v^w = \hat{\mathbf{z}}_v^w, \mathbf{r}_u(t), d_u)$ given v, w

and d_u. $H(t, w, v, d_u)$ is given by

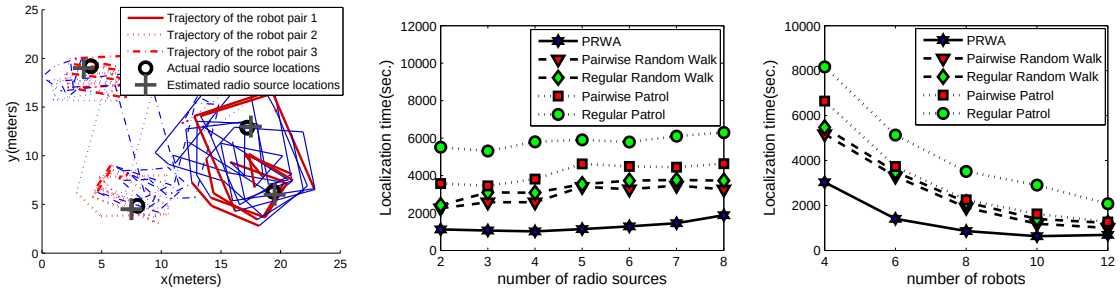
$$H(t, w, v, d_u) = -\sum_{i \in \mathcal{S}_u} \Big(\begin{matrix} P(C_i|\hat{\mathbf{Z}}_v^w = \hat{\mathbf{z}}_v^w, \mathbf{r}_u(t), d_u) \\ \times \ln P(C_i|\hat{\mathbf{Z}}_v^w = \hat{\mathbf{z}}_v^w, \mathbf{r}_u(t), d_u) \end{matrix} \Big) \tag{6.20}$$

where $P(C_i|\hat{\mathbf{Z}}_v^w = \hat{\mathbf{z}}_v^w, \mathbf{r}_u(t), d_u)$ is obtained from (7.1) and (7.2) after calculating the sensing model (6.16) with $\hat{\mathbf{z}}_v^w$. We choose the optimal d_u^* that minimizes the following Shannon entropy for the cluster region over the period τ_u when the robot is inside $\mathbf{R}_u$,

$$d_u^* = \arg\min_{d_u} \int_t^{t+\tau_u} \sum_{w=w_{min}}^{w_{max}} \sum_{v \in \mathcal{S}_u} H(t, w, v, d_u). \tag{6.21}$$

Note that here we assume that w is evenly distributed over integer values in $[w_{min}, w_{max}]$. In fact, we can estimate the more accurate distribution of w once more received signals become available to improve the model.

E. Experiments



(a) (b) (c)

Fig. 30.: Experiment results (a) Sample robot trajectories with three pairs of robots for a four-radio source case. The red and blue lines represent the on-ridge and off-ridge movements, respectively. (b) Localization time vs. number of radio sources. (c) Localization time vs. number of robots.

We have implemented the algorithms and the simulation platform using Microsoft Visual C++ .NET 2005 with OpenGL on a PC Desktop with an Intel 2.13GHz Core 2 Duo CPU, 2GB RAM, and Windows XP. The radio sources are XBee Pro with ZigBeeT/802.15.4 OEM radio frequency modules produced by Digi International Inc. The antenna is calibrated first with the radio sources. The calibration establishes the parameters in (6.3). We use the data from the real hardware to drive the simulation experiments below.

The grid is a square with 50×50 cells. Each grid cell has a size of 50.0×50.0 cm^2. Each radio source generates radio transmission signals according to an *i.i.d.* Poisson process with a rate of $\lambda = 0.05$ packets per second. We choose the probability convergence threshold as $p_t = 0.9$ which means if $P(C_i|\tilde{\mathbf{Z}} = \tilde{\mathbf{z}}) > 0.9$, the algorithm outputs the cell as a radio source location. During each trial of the simulation, we randomly generate radio source locations in the grid and randomly set their power levels as one of five power levels offered by XBee Pro nodes.

Fig. 30a illustrates the robot trajectories for a sample case with four radio sources using three pairs of robots. The initial positions of the robots are the center of the field. As shown in Fig. 30a, the on-ridge movements appear near radio sources. At the end of the experiments, the estimated locations of radio sources are within the error range of the actual locations. The localization process is successful.

We also compare the PRWA algorithm to four heuristics. Two of the four heuristics are based on random walk: a pairwise random walk and a regular random walk. In the pairwise random walk, robots are paired just as PRWA does. Each pair is treated as a super robot to perform a random walk together while all robots perform independent movements in the regular random walk. The remaining two heuristics are based on a fixed-route patrol: the robots patrol the field using a predefined route that covers the search region. Again, robots are either paired which results in a

pairwise patrol or non-paired which results in a regular patrol. Robot pairs in the pairwise patrol or individual robots in the regular patrol are distributed evenly along the route to increase coverage.

The experiment compares all five methods under different numbers of radio sources and robots. Figs. 30b and 30c illustrate experiment results. Each data point is an average of 100 independent trials. The results show that PRWA is consistently the fastest method under all comparisons. Also, the pairwise random walk and the pairwise patrol are consistently faster than the regular random walk and patrol, respectively. This is expected because paired robots are more efficient with their limited sensing ranges. Another interesting observation is that the two random walk-based methods are faster than the two fixed-route patrol methods. This is expected because random walk can bring robots together from time to time, which increases the number of effective pairs and hence listening efficiency. The fixed-route patrol methods emphasize coverage and spread robot pairs or individual robots apart along the route and hence cannot create many effective pairs, which decreases localization efficiency. The results in Fig. 30c also show that the difference between the five methods decreases as the number of robots increases. However, in reality, the number of robots is often constrained to where PRWA is superior.

F. Conclusions

We reported a new localization method that enables a team of mobile robots to localize multiple unknown transient radio sources. To cope with the challenges from signal correspondence, limited sensing ranges, and unknown transmission power, we paired up robots and developed a sensing model using the signal strength ratio from the paired robots. We formally proved that the sensed conditional joint posterior

probability of source locations for the $m-$robot team can be obtained by combining that of pairwise joint posterior probabilities. Moreover, we proposed a pairwise ridge walking algorithm (PRWA) to coordinate the robot pairs based on the clustering of high probability regions and the minimization of local Shannon entropy. We implemented the algorithm and tested it under hardware-driven simulation. Results show that PRWA-based localization consistently outperforms the other four heuristics in all settings tested.

The Unlimited communication range assumption resulting in the centralize localization is not realistic in the real world. Rather, the decentralized localization is natural characteristic of using multiple robots. Chapter VII addresses the decentralized localization problem.

G. Proof of Lemma 2

Proof. Recall that $Z_{p-q} = Z_p - Z_q$ and define $Z_{p+q} := Z_p + Z_q$. Denote $\mathbf{A} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$ as the linear transformation matrix. Therefore,

$$\begin{bmatrix} Z_{p-q} \\ Z_{p+q} \end{bmatrix} = \mathbf{A} \begin{bmatrix} Z_p \\ Z_q \end{bmatrix}. \tag{6.22}$$

Let us define R_A as the transformed integral region. According to [82], matrix $\mathbf{A}$ transforms the joint PDF of random variables Z_p and Z_q to the joint PDF of random variables Z_{p-q} and Z_{p+q},

$$f_{Z_{p-q}Z_{p+q}|C_i^1}(z_{p-q}, z_{p+q}) = \frac{1}{|\det \mathbf{A}|} f_{Z_pZ_q|C_i^1}(z_p, z_q) \tag{6.23}$$

where $|\det \mathbf{A}| = 2$. Hence $P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q | C_i^1)$ becomes

$$\begin{aligned} & P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q | C_i^1) \\ &= \int_{z_p \in \mathcal{I}_p} \int_{z_q \in \mathcal{I}_q} f_{Z_p Z_q | C_i^1}(z_p, z_q) dz_p dz_q \\ &= |\det \mathbf{A}| \iint_{R_A} f_{Z_{p-q} Z_{p+q} | C_i^1}(z_{p-q}, z_{p+q}) dz_{p+q} dz_{p-q}. \end{aligned} \tag{6.24}$$

Since Z_p and Z_q conform to *i.i.d.* normal distributions, Z_{p+q} and Z_{p-q} are also normal distributions. Moreover, Z_{p+q} and Z_{p-q} are independent because

$$\begin{aligned} Cov(Z_{p+q}, Z_{p-q}) &= Cov(Z_p + Z_q, Z_p - Z_q) \\ &= Var(Z_p) - Var(Z_q) = 0. \end{aligned}$$

Therefore, we know

$$\begin{aligned} f_{Z_{p-q}|C_i^1}(z_{p-q}) &= Bel\left(\mu_p - \mu_q, 2\sigma^2\right), \\ f_{Z_{p+q}|C_i^1}(z_{p+q}) &= Bel\left(\mu_p + \mu_q, 2\sigma^2\right), \\ f_{Z_{p-q} Z_{p+q}|C_i^1}(z_{p-q}, z_{p+q}) &= f_{Z_{p+q}|C_i^1}(z_{p+q}) f_{Z_{p-q}|C_i^1}(z_{p-q}). \end{aligned} \tag{6.25}$$

The integral over R_A in (6.24) can be calculated as follows,

$$\begin{aligned} & \iint_{R_A} f_{Z_{p-q} Z_{p+q}|C_i^1}(z_{p-q}, z_{p+q}) dz_{p+q} dz_{p-q} \\ &= \int_{\tilde{z}_p - \tilde{z}_q - 1}^{\tilde{z}_p - \tilde{z}_q + 1} \Bigg(f_{Z_{p-q}|C_i^1}(z_{p-q}) \\ &\times \int_{2\tilde{z}_q - 1 + z_{p-q}}^{2\tilde{z}_p + 1 - z_{p-q}} f_{Z_{p+q}|C_i^1}(z_{p+q}) dz_{p+q} \Bigg) dz_{p-q}. \end{aligned} \tag{6.26}$$

To simplify the above integral, let us define

$$g(z_{p-q}) = \int_{2\tilde{z}_q - 1 + z_{p-q}}^{2\tilde{z}_p + 1 - z_{p-q}} f_{Z_{p+q}|C_i^1}(z_{p+q}) dz_{p+q}. \tag{6.27}$$

Using the first mean value theorem for integration, we derive the relation between $P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q | C_i^1)$ and $P(Z_{p-q} \in \mathcal{I}_{p-q} | C_i^1)$ as,

$$\begin{aligned}
&P(Z_p \in \mathcal{I}_p, Z_q \in \mathcal{I}_q | C_i^1) \\
&= |\det \mathbf{A}| \cdot \int_{\tilde{z}_p - \tilde{z}_q - 1}^{\tilde{z}_p - \tilde{z}_q + 1} f_{Z_{p-q}|C_i^1}(z_{p-q}) g(z_{p-q}) dz_{p-q} \\
&= 2\ g(\xi) \int_{\tilde{z}_p - \tilde{z}_q - 1}^{\tilde{z}_p - \tilde{z}_q + 1} f_{Z_{p-q}|C_i^1}(z_{p-q}) dz_{p-q} \\
&= 2\ g(\xi)\ P(Z_{p-q} \in \mathcal{I}_{p-q} | C_i^1) \\
&= \frac{1}{\eta_{pq}} P(Z_{p-q} \in \mathcal{I}_{p-q} | C_i^1),
\end{aligned} \tag{6.28}$$

where $\xi \in \mathcal{I}_{p-q}$ and $\eta_{pq} = \frac{1}{2\ g(\xi)}$ is the normalizing factor. □

H. Proof of Lemma 4

Proof. Assume robot q, $q \in M_0$, has an ideal receiver which does not have the sensing range limit. Denote $Z_q = \psi_q + \omega_q$ as the RSS readings of the ideal receiver. The ideal receiver would allow us to use Z_{p-q}^{11} as the RSS ratio instead of Z_{p-q}^{10} from a regular receiver. Also, $\psi_q \leq \zeta$. According to (6.4), we know

$$Z_{p-q}^{10} = Z_{p-q}^{11} + Z_q - (\zeta + \omega_q). \tag{6.29}$$

The PDF of $(Z_{p-q}^{10} | C_i^1)$ is rewritten by

$$\begin{aligned}
&f_{Z_{p-q}^{10}|C_i^1}(z_{p-q}^{10}) \\
&= f_{Z_{p-q}^{10}|C_i^1, Z_p > Z_q}(z_{p-q}^{10}) P(Z_p > Z_q) \\
&\quad + f_{Z_{p-q}^{10}|C_i^1, Z_p \leq Z_q}(z_{p-q}^{10}) P(Z_p \leq Z_q) \\
&= f_{Z_{p-q}^{10}|C_i^1, Z_p > Z_q}(z_{p-q}^{10})
\end{aligned} \tag{6.30}$$

where $P(Z_p > Z_q) = 1$ and $P(Z_p \leq Z_q) = 0$.

Conditioning on Z_q and using the first mean value theorem for integration, the PDF of $(Z^{10}_{p-q}|C^1_i, Z_p > Z_q)$ becomes,

$$\begin{aligned}
&f_{Z^{10}_{p-q}|C^1_i, Z_p>Z_q}(z^{10}_{p-q}) \\
&= \int_{-\infty}^{+\infty} f_{Z^{10}_{p-q}, Z_q|C^1_i, Z_p>Z_q}(z^{10}_{p-q}, z_q)dz_q \\
&= \int_{-\infty}^{\zeta+\omega_q} \frac{f_{Z^{10}_{p-q}, Z_q|C^1_i, Z_p>Z_q}(z^{10}_{p-q}, z_q)}{f_{Z_q|C^1_i}(z_q)} \cdot f_{Z_q|C^1_i}(z_q)dz_q \\
&= \int_{-\infty}^{\zeta+\omega_q} f_{Z^{10}_{p-q}|Z_q, C^1_i, Z_p>Z_q}(z^{10}_{p-q}|z_q) f_{Z_q|C^1_i}(z_q)dz_q \\
&= f_{Z_q|C^1_i}(\xi') \int_{-\infty}^{\zeta+\omega_q} f_{Z^{10}_{p-q}|Z_q, C^1_i, Z_p>Z_q}(z^{10}_{p-q}|z_q)dz_q \\
&= \frac{1}{\eta^{10}} \int_{-\infty}^{\zeta+\omega_q} f_{Z^{10}_{p-q}|Z_q, C^1_i, Z_p>Z_q}(z^{10}_{p-q}|z_q)dz_q,
\end{aligned} \tag{6.31}$$

where $-\infty \leq \xi' \leq \zeta + \omega_q$ and $\eta^{10} = \frac{1}{f_{Z_q|C^1_i}(\xi')}$ is the normalizing factor.

Plugging (6.29) in, we have,

$$\begin{aligned}
&f_{Z^{10}_{p-q}|C^1_i, Z_p>Z_q}(z^{10}_{p-q}) \\
&= \frac{1}{\eta^{10}} \int_{-\infty}^{\zeta+\omega_q} f_{Z^{10}_{p-q}|Z_q, C^1_i, Z_p>Z_q}(z^{10}_{p-q}|z_q)dz_q \\
&= \frac{1}{\eta^{10}} \int_{z^{10}_{p-q}}^{+\infty} f_{Z^{11}_{p-q}|C^1_i}(z^{11}_{p-q})dz^{11}_{p-q} \\
&= \frac{1}{\eta^{10}} \left(1 - \mathrm{F}_{Z^{11}_{p-q}|C^1_i}(z^{10}_{p-q})\right).
\end{aligned} \tag{6.32}$$

Thus, we have

$$
\begin{aligned}
&P(Z^{10}_{p-q} \in \mathcal{I}_{p-q} | C^1_i) \\
&= \int_{\tilde{z}_p - \tilde{z}_q - 1}^{\tilde{z}_p - \tilde{z}_q + 1} f_{Z^{10}_{p-q} | C^1_i}(z) dz \\
&= \int_{\tilde{z}_p - \tilde{z}_q}^{\tilde{z}_p - \tilde{z}_q + 1} f_{Z^{10}_{p-q} | C^1_i, Z_p > Z_q}(z) dz \\
&= \int_{\tilde{z}_p - \tilde{z}_q}^{\tilde{z}_p - \tilde{z}_q + 1} \frac{1}{\eta^{10}} \left(1 - \mathrm{F}_{Z^{11}_{p-q} | C^1_i}(z) \right) dz \\
&= \frac{1}{\eta^{10}} \left(1 - \int_{\tilde{z}_p - \tilde{z}_q}^{\tilde{z}_p - \tilde{z}_q + 1} \mathrm{F}_{Z^{11}_{p-q} | C^1_i}(z) dz \right). \qquad (6.33)
\end{aligned}
$$

□

CHAPTER VII

DECENTRALIZED MRMT LOCALIZATION

A. Introduction

The fast development of wireless sensor network (WSN) technology provides great tools to gather information from our surrounding environment. However, WSN can also be a significant threat to our security and privacy in hostile environments. For example, an enemy may deploy a sensor field to detect troop movements. The large number of miniature sensors in a large field makes it difficult to manually search and neutralize these sensor. We are interested in developing algorithms to enable a team of mobile robots to perform the task.

In this "robot network" vs. "sensor network" setup, each party has its own advantages and limitations. Robots have mobility while sensors do not. Robots know their own locations and received signal strength (RSS) readings. However, robots do not understand the radio transmission protocol of the sensor network and treat sensor nodes as plain radio sources. Furthermore, signal source anonymity, short transmission duration, dynamic transmission patterns, variable transmission power levels, and the unknown number of signal sources further challenge the robot team. In addition, there is usually much less number of robots than that of the radio sources. Also, the robot must consider they communication range constraints.

Building on our prior work, we propose a two step approach: first we decentralize belief functions that robots use to track source locations using checkpoint-based synchronization, and second we propose a motion planning strategy to coordinate robots to ensure the existence of checkpoints. We formally analyze memory usage, data amount in communication, and searching time for the proposed algorithm. We

show that our planning algorithm ensures the decentralized belief functions to be synchronized in a timely manner with explicit memory and communication requirements. Our searching time is insensitive to the number of radio sources. We have implemented the proposed algorithm and compare it with two heuristics in simulation based on real sensory data. Our algorithm successfully perform the searching task and has show a clear advantage in searching time without significant increase of memory usage.

B. Problem Definition

Our searching problem builds on the following setup and assumptions:

1. Both robots and radio sources reside in an open 2D Euclidean space.
2. Each robot has a limited communication range and a limited sensing range.
3. Each robot knows its position using the Global Positioning System (GPS). GPS clocks also provide accurate time for the synchronization purpose.
4. The unknown network traffic is light and each target radio transmission is short, which are the typical characteristics of a low power sensor network.
5. Transmission powers of radio sources are unknown to robots and may change from time to time. However, locations of radio sources do not change.

For the new decentralized approach, we will follow the same problem definition in the corresponding centralized versions Chapter V, where the searching problem is partitioned into two sub problems:

Definition 5 (Sensing Problem)**.** *Given the RSS readings and corresponding locations from robots, update robot belief functions for radio source locations.*

Definition 6 (Planning Problem)**.** *Given the belief functions, plan robot trajectories to increase searching efficiency.*

We will concretely define the belief functions in detail later in the paper. As we can see, this is a Monte Carlo type algorithmic approach with the following stopping time for radio source detection,

Definition 7 (Searching Condition)**.** *A radio source is considered as found if the belief function is bigger than a preset threshold p_t.*

Now let us begin with the sensing problem.

C. Decentralized Belief Functions

Belief functions track the radio source distribution based on RSS readings and robot locations. They are usually built on a Bayesian framework and antenna models to allow incremental update. In our previous Chapter V on the centralized localization of transient and unknown radio sources, we propose a Spatial Temporal Occupancy Grid (SPOG) as the robots' common belief functions. Let us review it first and then we will decentralize SPOG.

1. A Brief Review of SPOG

SPOG partitions the searching region into small and equal-sized grid cells. Define $i \in \mathcal{N}$ as the cell index variable where $\mathcal{N} := \{1, ..., n\}$ is the grid cell index set and n is the total number of cells. SPOG tracks two types of probabilistic events: C_i represents the event that cell i contains a radio source and C_i^1 represents the event that cell i is the active source when a transmission is detected. C_i^1 actually reflects the relative transmission rates among multiple sources, which is a temporal dimension

signature. Define $P(C)$ as the probability for event C. $P(C_i)$ and $P(C_i^1)$ characterize spatiotemporal behaviors of transient radio sources.

Let $l \in \mathcal{M} := \{1, ..., m\}$ be the robot index variable where m is the total number of robots and $\mathcal{M}$ is the robot index set. Note that m is always an even number since we will pair robots up later. Discrete time k or the corresponding continuous time t^k refers to each moment when a transmission is detected by robots. Let $\mathbf{x}_l^k = [x_l^k, y_l^k]^T$ be the location of robot l at time k and $\mathbf{X}^k = [\mathbf{x}_1^k, ..., \mathbf{x}_m^k]^T$ be a set of all robot locations at time k. Let the discrete random variable $\tilde{Z}_l^k$ be the RSS reading of the l-th robot at time k. Define $\tilde{\mathbf{Z}}^k = [\tilde{Z}_1^k, ..., \tilde{Z}_m^k]^T$ as a discrete random vector of all the RSS readings at time k and let $\tilde{\mathbf{z}}^k := [\tilde{z}_1^k, ..., \tilde{z}_m^k]^T$ be corresponding values. As a convention, we use lower cases of random variables or vectors to denote their values. Define $\mathcal{Z}^{1:k} := \{\tilde{\mathbf{z}}^1, ..., \tilde{\mathbf{z}}^k\}$ as the set of all RSSs sensed from the beginning of the searching to t^k. Define $P(C_i|\mathcal{Z}^{1:k})$ as the conditional probability that cell i contains at least one radio source given $\mathcal{Z}^{1:k}$. Similarly, we define $P(C_i|\mathcal{Z}^{1:k-1})$, $P(C_i^1|\mathcal{Z}^{1:k})$, and $P(C_i^1|\mathcal{Z}^{1:k-1})$.

At time k, event $\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k$ is perceived by robots. The posterior probability $P(C_i|\mathcal{Z}^{1:k})$ over the grid needs to be updated. According to Chapter V, this is actually a nested multivariate Bayesian process,

$$P(C_i|\mathcal{Z}^{1:k}) = \frac{\begin{array}{c} P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)P(C_i^1|\mathcal{Z}^{1:k-1})+ \\ P(C_i|\mathcal{Z}^{1:k-1}) \left[\sum_{s\neq i, s\in I} P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_s^1)P(C_s^1|\mathcal{Z}^{1:k-1})\right] \end{array}}{\sum_{i\in I} P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)P(C_i^1|\mathcal{Z}^{1:k-1})}, \quad (7.1)$$

$$P(C_i^1|\mathcal{Z}^{1:k}) = \frac{P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)P(C_i^1|\mathcal{Z}^{1:k-1})}{\sum_{s\in I} P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_s^1)P(C_s^1|\mathcal{Z}^{1:k-1})}, \quad (7.2)$$

where $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)$ refers to the conditional probability for event $(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k)$ when cell i is transmitting. This is the sensing model. In Chapter V, we have shown how to obtain $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k|C_i^1)$ from antenna radiation pattern and signal attenuation model.

It is worth noting that $P(\tilde{\mathbf{Z}}^k = \tilde{\mathbf{z}}^k | C_i^1)$ has to be established on signal ratios from robot pairs when the source transmission power is unknown. Since this problem is detailed in the centralized version in Chapter VI, we omit it here. It is worth noting that SPOG allows different antenna models and is flexible in different antennas or robot configurations. Eqs. (7.1) and (7.2) are in an incremental conditional format for recursive update. As more RSS readings enter the system over time, $P(C_i|\mathcal{Z}^{1:k})$ converges until $P(C_i|\mathcal{Z}^{1:k}) > p_t$ which means that searching condition in Def. 7 is satisfied.

2. Decentralized SPOG (D-SPOG)

In the decentralized system, each robot has to maintain its own local SPOG by accumulating RSS readings internally and exchanging information with other robots whenever other robots move into its communication range. However, the centralized SPOG described in (7.1) and (7.2) depends on the strict order of complete observation set $\mathcal{Z}^{1:k}$. Robots cannot arbitrarily use their partial receptions to generate a local SPOG. Furthermore, robots cannot keep their readings forever for future information exchange due to limited onboard memory space.

Before we address this problem, let us take a close look at the decentralized system. There are three types of discrete events in the decentralized system: *detection events* referring to moments when a transmission is detected by robots, *rendezvous events* describing moments when a robot moves into another robot's communication range, and *planning events* describing moments when a robot starts a new path planning. Recall that k is the time index variable for the detection event. Denote j and κ as the rendezvous event and the planning event, respectively. Define $t_\kappa^{j,k}$ to describe the three events in the continuous time domain as a convention in the paper. To reduce cluttering, we may also use a reduced version such as t^k and t^j for the

corresponding event time. t_κ indicates the beginning of the κ-th planning period.

An effective coordination plan should allow robots to exchange information among each other so that all robots have the same set of observations $\mathcal{Z}^{1:k}$ at time j, $t^j \geq t^k$. This is the time that all robots can update their SPOG up to time k. In such a way, the centralized SPOG can be decentralized and synchronized among all robots. The "delayed synchronization" concept is proposed as a checkpoint by Leung et al. [59]. Let us denote $Y(t^k, t^j)$ as the checkpoint. Note that each checkpoint for a robot always has two time variables: it begins with an early detection event time and ends with a future rendezvous event time because information is always generated by detection events and synchronized by rendezvous events.

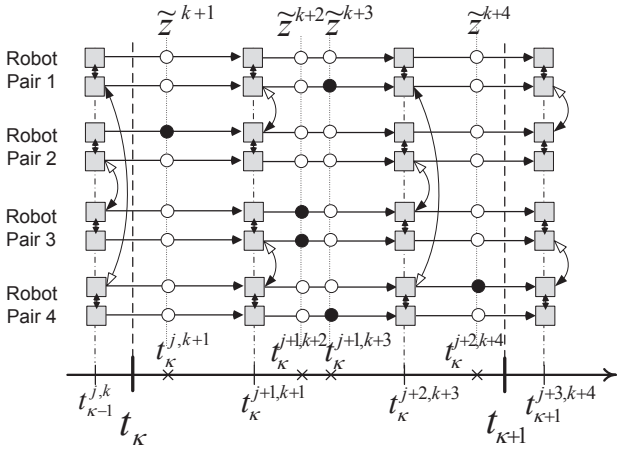


Fig. 31.: A sample information flow graph for four robot pairs. Gray rectangles represent robot rendezvous events. Arc arrows in vertical direction indicate information exchange between robots in communication range. Black and white circles represent events for robots with and without detection of active radio transmissions, respectively.

Fig. 31 shows an information flow graph to illustrate the checkpoint concept and how information is passed around the distributed robot pairs in D-SPOG. Note that

robots have been paired up in this graph because it takes two robots to obtain a signal ratio for radio sources with unknown and variable transmission powers. We ignore the intra-pair communication because a pair can always talk to each other according to planning. Following the arc arrows in vertical directions, we can see that both $Y(t_\kappa^{j,k+1}, t_\kappa^{j+2,k+3})$ and $Y(t_\kappa^{j+1,k+2}, t_\kappa^{j+3,k+4})$ are checkpoints.

To build a D-SPOG, the remaining question is how each robot stores and exchanges information. Say that $Y_l(t^{k-1}, t^{j-1})$ is the last checkpoint for robot l. After the update at t^{j-1}, D-SPOG for the robot l is synchronized up to t^{k-1} with the fictitious centralized SPOG according to the checkpoint property. Robot l only needs to store its own locations and RSS readings after t^{k-1}, which results in significant saving in memory. Due to the fact that robots without detection may not know the time of the radio transmission, each robot has to keep track of its trajectory in addition to RSS readings. Let $\mathcal{W}_l^{k-1,t}$ be the measurement set internally generated by robot l between t^{k-1} and current time $t, t > t^{k-1}$:

$$\mathcal{W}_l^{k-1,t} = \left\{ \mathbf{x}_l((t^{k-1}, t]), \mathbf{z}_l((t^{k-1}, t]) \right\}, \tag{7.3}$$

where $\mathbf{x}_l(\cdot)$ is the robot trajectory and $\mathbf{z}_l(\cdot)$ is the RSS reading set for the duration. Similarly, we define $\mathcal{W}_l^{k-1,j}$ and $\mathcal{W}_l^{k-1,k}$ by replacing t with t^j and t^k, respectively.

Let us define the measurement set of robot l at rendezvous time t^j as $\Gamma_l^{k-1,j}$ which contains information from both its on-board sensors and other robots. To describe the moment right before the robot l encounters another robot, we introduce a $(\cdot)^-$ notation. It is clear that $\mathcal{W}_l^{k-1,j} \subseteq (\Gamma_l^{k-1,j})^-$. At t^j, robot l meets robot p, which has measurement set $(\Gamma_p^{k'-1,j})^-$ prior to the information exchange where $t^{k'-1}$ is the detection event time of the last checkpoint that robot p has. Note that $t^{k'-1}$ and t^{k-1} are not necessarily the same. The two robots first compare the two times because a newer time means a more recent D-SPOG. The other robot should synchronize its

SPOG to the recent one. After synchronizing their SPOGs, they need to synchronize the measurement set. Note that we have $(\Gamma_l^{k-1,j})^-$ for robot l and $(\Gamma_p^{k'-1,j})^-$ for robot p before the synchronization. Without loss of generality, we assume $t^{k-1} \geq t^{k'-1}$, the synchronization process is,

$$\Gamma_l^{k-1,j} = \Gamma_p^{k-1,j} = (\Gamma_l^{k-1,j})^- \cup (\Gamma_p^{k-1,j})^-, \tag{7.4}$$

where $(\Gamma_p^{k-1,j})^- = (\Gamma_p^{k'-1,j})^- \setminus \Gamma_p^{k'-1,k-1}$ is obtained by discarding the measurement between $t^{k'-1}$ and t^{k-1}, a reduction in memory usage.

After the rendezvous event, each robot needs to search if a more recent checkpoint can be established. For robot l, it checks $\Gamma_l^{k-1,j}$ to see if the measurement set contains information from all other robots for detection events happened after the $k-1$-th detection event by searching for the maximum δ,

$$\delta = \arg \max_{\delta \in \mathbb{Z} \cap [-1,\infty) \text{ and } t^{k+\delta} \leq t} \delta \left[\Pi_{p=1}^{m} (\mathcal{W}_l^{k-1,k+\delta} \subset \Gamma_l^{k-1,j}) \right], \tag{7.5}$$

where $(\mathcal{W}_l^{k-1,k+\delta} \subset \Gamma_l^{k-1,j})$ is a logic operation which returns 0 if the relationship is not satisfied and 1 otherwise. Only the existence of nonnegative solution indicates a new checkpoint $Y_l(t^{k+\delta}, t^j)$ can be established and hence the D-SPOG can be updated. After the update, it is clear that D-SPOG is equivalent to the centralized SPOG update with a delay of $t - t^{k+\delta}$. We have the following lemma,

Lemma 6. *To ensure proper update of D-SPOG at checkpoints, both the amount of information that every robot stores onboard and the amount of information exchange during the rendezvous event between two robots are $O(n + m(t - t^{k-1}))$, where t is current time and t^{k-1} is the detection event time of the latest checkpoint.*

Proof. Each robot has to store a D-SPOG which takes $O(n)$ memory space. At the worst case scenario, $\Gamma_p^{k-1,t}$ may contain $m-1$ robots' trajectories and RSS reading

sets from t^{k-1} to t. Since the trajectory storage using a fixed period and the mean number of transmissions is linear to the time duration, hence the overall amount of information stored on each robot is $O(n + m(t - t^{k-1}))$. Since both D-SPOG and their measurement sets need to be synchronized during the rendezvous of robots, the lemma holds. □ □

However, if just one robot is geographically isolated with others which results in no communication to others, no checkpoint can be established. $t - t^{k-1}$ becomes unbounded and the robot may quickly run out of memory which leads to failure. To address this problem, we propose a decentralized planning that guarantees periodic checkpoint existence.

D. Decentralized Planning

The decentralized planning strategy needs to take checkpoint existence, communication range limit, synchronization, and searching time into consideration. We build the new planning strategy on our existing ridge walking algorithm (RWA), which was designed for the centralized version of the problem in Chapter V. RWA and its variants have proved to be effective in accelerating the convergence of $P(C_i|\mathcal{Z}^{1:k})$ to reduce searching time and have excellent scalability.

1. A Brief Review of RWA and Pairwise RWA (PRWA)

In SPOG or D-SPOG, $P(C_i|\mathcal{Z}^{1:k})$ is the conditional probability that cell i contains a radio source. RWA plans a path for a single robot by building on this spatial distribution of radio sources. We generate a level set $L(p)$, $p \in (0, 1]$ by using a plane parallel to the ground plane to intersect the mountain-like distribution $P(C_i|\mathcal{Z}^{1:k})$ at height p. The intersection generates $L(p)$ which contains all cells with $P(C_i|\mathcal{Z}^{1:k})$

above the plane. $L(p)$ usually consists of several disconnected components. The irregular contours in Fig. 32a is an example of $L(0.1)$. For each component, we define its ridge as the longest line segment along its dominating direction in Chapter V. We know each ridge has very high probability of being close to a potential signal source. We generate a Traveling Salesperson Problem (TSP) tour which contains all ridges. For off-ridge segments, the robot moves at its fastest speed. The solid red and dashed blue lines in Fig. 32a represent on-ridge and off-ridge movements, respectively. For on-ridge segments, the robot spends the time proportional to the summation of posterior conditional probability $P(C_i|\mathcal{Z}^{1:k})$ over the corresponding isolated level set on each ridge. This allows the robot to spend most of its time on ridges, then the intuition yields the ridge walking algorithm (RWA) as detailed in Chapter V. RWA has shown superior convergence performance and scalability in searching for multiple signal sources.

PRWA extends RWA to plan trajectories for a team of robots to handle unknown and changing transmission power in Chapter VI. First, PRWA expands SPOG by developing a pairwise sensing model based on RSS ratios from robot pairs instead of assuming known absolute source transmission power. Second, PRWA coordinates robots in pairs by minimizing information entropy so that a robot pair can be viewed as a super-robot in planning. We will inherit the pairwise sensing model and coordinate robots in pairs in this decentralized version.

2. Decentralized Pairwise Ridge Walking Algorithm (DPRWA)

Similar to RWA, DPRWA coordinate robot pairs to patrol on TSP tours that link all ridges. We generate a TSP tour in each planning period. In fact, each planning period is divided into two parts: short inter-ring movements for transition between TSP tours in adjacent planning periods followed by long intra-ring movements for

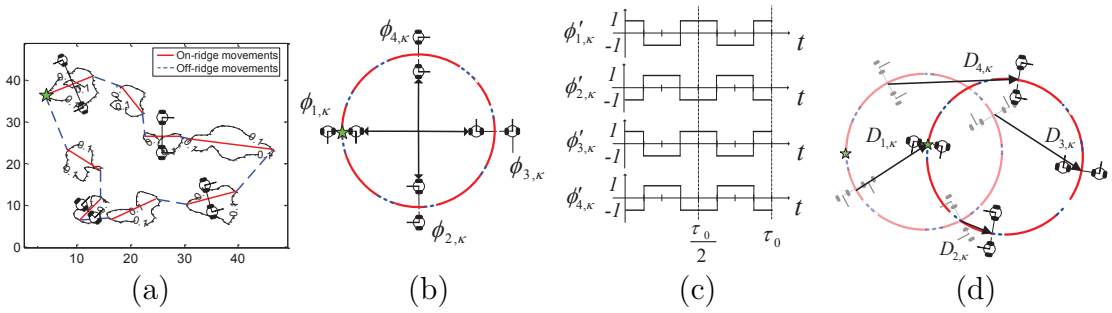


Fig. 32.: Sample results for the decentralized planning using 4 robot pairs. (a) Sample robot trajectories of DPRWA. The solid red and dashed blue lines represent the on-ridge and off-ridge movements, respectively. (b) An example of the intra-ring movements on the time ring. (c) Changes of robot pair directions corresponding to the intra-ring movements. (d) An example of the inter-ring movements using time ring space instead of Euclidean space.

time allocated for robots to patrol the TSP tour.

a. Inter-Ring and Intra-Ring Movements

Let us begin with inter-ring movements. Before the planning period starts, each robot pair computes the TSP tour. All robot pairs actually share the same TSP tour from the synchronized D-SPOG. Inter-ring movements allow robots to move from current TSP tour to the next. Each robot has a pre-allocated beginning position on the TSP tour (detailed later as initial positions for intra-ring movements). Therefore, the amount of travel time for inter-ring movement can be predicted as soon as the TSP tour for the planning period is established. Define $D_{u,\kappa}$ as the inter-ring travel time of the u-th robot pair. To synchronize the starting time of intra-ring movements of all robot pairs, every robot pair waits until all other robot pairs reach at their initial positions. Define D_{κ}^{max} as the maximum travel time: $D_{\kappa}^{max} = \arg\max_u D_{u,\kappa}$. For

those robots that arrive early due to short traveling distance, they need to wait $\omega_u^\kappa = D_\kappa^{max} - D_{u,\kappa}$ before the synchronized intra-ring movements start. Synchronization will be detailed later in Section c. To save time for patrolling TSP tours, robots moves at their fastest speed to shorten inter-ring movement time.

Now let us introduce intra-ring movements. Since the TSP tour is a continuous loop, it can be mapped to a circular ring in time with its circumference being the time for a single pair of robots to traverse the entire TSP tour, which is defined as τ_0. The mapping is one-to-one if we fix a point correspondence in the mapping. For example, the leftmost point (the smallest in lexicographic order) on the TSP tour corresponds to the 9 clock position on the time ring as the green stars shown in Figs. 32a and 32b. All robots share this mapping rule to synchronize their positions on the time ring. The introduction of time ring can facilitate our planning. Under the time ring, the inter-ring movements can also be simplified as shown in Fig. 32d.

As illustrated in Fig. 32b, each pair of robots are evenly distributed on the time ring. Define $\phi_{u,\kappa}$ and $\phi'_{u,\kappa}$ as the position and speed of the u-th robot pair on the time ring, respectively. Robots' speeds on the time ring are unitary based on the definition of the time ring. Odd and even pairs are initially assigned to move on the time ring counterclockwise and clockwise, which are represented as 1 and -1, respectively. Recall there are m robots and hence $m/2$ pairs. We have

$$\phi_{u,\kappa} = \frac{2\tau_0(u-1)}{m}, \quad \text{and} \quad \phi'_{u,\kappa} = \begin{cases} 1 & \text{if } u \text{ is odd,} \\ -1 & \text{otherwise,} \end{cases} \tag{7.6}$$

as the initial positions and speeds for robot pairs. Fig. 32b illustrates initial positions and directions (represented by the heading direction of each robot) of four robot pairs. When two robot pairs rendezvous on the time ring, they exchange information and then reverse their moving directions. Therefore, each robot pair oscillates on the time

ring centered at its initial position as shown in Fig. 32c.

Define T as the time of the intra-ring movements. The robot pairs have to execute the intra-ring movements long enough to ensure the existence of the checkpoint.

Lemma 7. *Each robot pair has at least one checkpoint if the intra-ring movement time T is*

$$T = \begin{cases} \frac{\tau_0}{2} & \text{if } \frac{m}{2} \text{ is even,} \\ \frac{\tau_0}{2} + \frac{\tau_0}{m} & \text{otherwise.} \end{cases} \tag{7.7}$$

Proof. Starting from its initial position, robot pair u meets its two neighbors $u-1$ and $u+1$ and returns to the initial position with a period of $\frac{2\tau_0}{m}$. The furthest point is half circle away, $\tau_0/2$. The two rendezvous bring information from further both downstream (from $u+1$ at lower half circle) and upstream (from $u-1$ at upper half circle). Imagine the information is sent out from the furthest robot pair from both upstream and downstream directions. When the information reaches the robot pair u, it contains information from all robot pairs. There are two cases: even and odd numbers of robot pairs. For the even case, due to the unitary speed, $\tau_0/2$ is the exact time robot pair u gathers the information. Eq. (7.5) has a nonnegative solution and a new checkpoint is established. For the odd case, the proof is similar except that there needs to be an additional half period for meeting the additional pair. □ □

Fig. 31 illustrates the information flow and checkpoint existent for the four robot pair case in Fig. 32b and Fig. 32c under the oscillating intra-ring movements.

b. Memory Usage and Expected Searching Time

DPRWA ensures periodical checkpoint existence which leads to guaranteed performance. To measure the algorithm performance, we employ two metrics: memory usage for each robot and the expected searching time for each radio source.

For the memory usage, following Lemma 7, we have the following theorem,

Theorem 3. *DPRWA guarantees D-SPOG has a time delay less than* $D_\kappa^{max} + T$ *if comparing the D-SPOG to the centralized SPOG. To achieve that, each robot requires* $O(n + m(D_\kappa^{max} + T))$ *memory space.*

The expected searching time for a radio source has to depend on the source transmission rate. Assume a radio source i transmits signals according to a Poisson process with a rate of λ_i. In Chapter III, we have introduced the expected searching time (EST) for a single-robot-single-target case. Let us extend this analysis to DPRWA. Denote T_s as the searching time. Similar to the EST analysis of RWA in Chapter V, we tighten the convergence condition from the probability threshold p_t to the condition of signal saturation. Radio source i is considered to be found if the robot pairs hear the transmission within the distance of d_a of the radio source. d_a is set to be small such that if the transmission is heard, the probability threshold p_t must be reached. This defines *a sensing circle* with its center at the radio source i and a radius of d_a. Define τ_{IN} and τ_{OUT} as portions of the time when traveling within and outside distance of d_a of radio source i, respectively. Hence

$$D_\kappa^{max} + T = \tau_{\text{IN}} + \tau_{\text{OUT}}. \tag{7.8}$$

We have the following theorem,

Theorem 4. *The expected searching time* $E(T_s)$ *of radio source* i *has the following upper bound,*

$$E(T_s) \leq D_\kappa^{max} + \frac{\tau_0}{m} + \frac{1}{\lambda_i} + (D_\kappa^{max} + T)E\big(\frac{e^{-\lambda_i \tau_{IN}}}{1 - e^{-\lambda_i \tau_{IN}}}\big). \tag{7.9}$$

Proof. From Theorem 1 in Chapter III, the expected searching time $E(T_s)$ of transient

radio source i is,

$$E(T_s) = E(D) + \frac{1}{\lambda_i} + E\big(\tau_{\text{OUT}} \frac{e^{-\lambda_i \tau_{\text{IN}}}}{1 - e^{-\lambda_i \tau_{\text{IN}}}}\big), \tag{7.10}$$

where D is the amount of time from the beginning of the search to the moment that the robot is within distance d_a of the radio source i for the first time. The theorem is built on the general case that the searching process can be modeled as a delayed alternative renewal reward process [66]. In our case, the renewal period starts at the first entry to the sensing circle by the robot team at each planning period. It is not exactly the same as the planning period but will share the same expected period length with the planning period. Since the search begins with the inter-ring movements in the first planning period, let us define probability event B as the event if the robot team meets the radio source during inter-ring movements. Therefore,

$$E(D) = E(D|B)P(B) + E(D|\overline{B})(1 - P(B)). \tag{7.11}$$

Event B is a small probability event given the large searching field size. Hence $P(B) \ll 1 - P(B) \approx 1$. Since the point of the first entry to the sensing circle could be anywhere on the time ring, it is uniformly distributed on the time ring. Also, we have $m/2$ pairs of robots evenly distributed on the time ring,

$$E(D|B) = D_\kappa^{max} + \frac{1}{2}\frac{\tau_0}{m/2} = D_\kappa^{max} + \frac{\tau_0}{m} \geq E(D). \tag{7.12}$$

Since the search region is usually much larger than the sensing region $\tau_{\text{IN}} \ll D_\kappa^{max} + T$, we have

$$E(\tau_{\text{OUT}}) \approx D_\kappa^{max} + T \tag{7.13}$$

from (7.8). Since each renewal period is independent and identically distributed (*i.i.d.*), it allows us to apply Theorem 1 in Chapter III. Since $D_\kappa^{max} + T$ is independent

of τ_{IN}, plugging (7.12 and 7.13) into (7.10), we have the result in (7.9). □ □

Remark 7. *An important result given by Theorem 4 is the fact that entries in (7.9) are not sensitive to the number of radio sources. This means that our DPRWA algorithm has excellent scalability when number of radio sources increases.*

c. Algorithm

We summarize our DPRWA as follows. Note that this algorithm runs on each robot pair, which skips the details of the internal coordination within each pair.

Synchronization: The algorithm runs at t_κ, the beginning of planning period κ. The algorithm relies on the D-SPOG at $t_{\kappa-1}$, which is the synchronized belief function across all robots. Therefore, all robots will have the same TSP tour, which ensures their motions are synchronized given the same plan, accurate clocks from GPS, and the same mapping rule between the time ring and the Euclidean space.

Virtual ridges: One point that we have yet to explain is the virtual ridge mentioned in line 3 of Alg. 5. Define $s_{\max}$ as the maximum number of ridges. If there are not enough ridges generated from the D-SPOG, we employ virtual ridges to ensure that there are $s_{\max}$ ridges. The main reason is that we do not want the robot to only explore regions with high probabilities. Virtual ridges are generated uniformly random in the searching region and also refreshed at every planning period. The virtual ridge sets are synchronized in the same way that D-SPOG does. The introduction of virtual ridge can be simply viewed as a sampling approach to cover regions with low probabilities.

Sparse sensor fields: The DPWRA in Alg. 5 forces all robots to share a single TSP tour. This is efficient when radio sources are relatively dense (i.e. the distances between disconnected components in the level set are less than the communication

Algorithm 5: Decentralized Pairwise Ridge Walking Algorithm

input : D-SPOG at $t_{\kappa-1}$
Apply the level set $O(n)$
Compute ridges and merge them with pre-generated virtual ridges $O(s_{\max})$
Compute the TSP tour from the merged ridge set $O((s_{\max}-1)!)$
`// Inter Ring Movements`
Compute $D_{u,\kappa}$ and D_{κ}^{max} $O(1)$
Move to initial positions $O(1)$
Wait ω_u^{κ} $O(D_{\kappa}^{max})$
`// Intra Ring Movements`
while $t \leq t_{\kappa} + D_{\kappa}^{max} + T$ **do** $O(T)$
  Patrolling along the TSP tour $O(1)$
  if $\phi_{u,\kappa}^{j,k} = \phi_{u-1,\kappa}^{j,k}$ *or* $\phi_{u,\kappa}^{j,k} = \phi_{u+1,\kappa}^{j,k}$ **then**
    $\phi_{u,\kappa}^{'j,k} = -\phi_{u,\kappa}^{'j,k}$ $O(1)$
    Update $\Gamma_{u,\kappa}^{j,k}$ $O(1)$
  end
end

range). However, sharing a TSP tour might not be efficient when the radio source are sparsely distributed in the searching region because robots have to waste a lot of time on off-ridge movements running between radio sources. Since the communication range is much larger than the sensing range given that robots have more power and better antenna than radio sources, the sensing processes in D-SPOG are independent across distant groups, which allows us to partition D-SPOG spatially into disjoint distant groups. Each group is treated as a separated problem with no requirement to merge D-SPOG during the partition. We can regroup periodically should inter-group distances change. At the moment of re-grouping, we can merge D-SPOG across groups. Robot pairs will be proportionally dispatched to different groups according to the total $P(C_i|\mathcal{Z}^{1:k})$ of each group. We may have to merge some close groups when there is insufficient number of robot pairs. For each group, we apply Alg. 5.

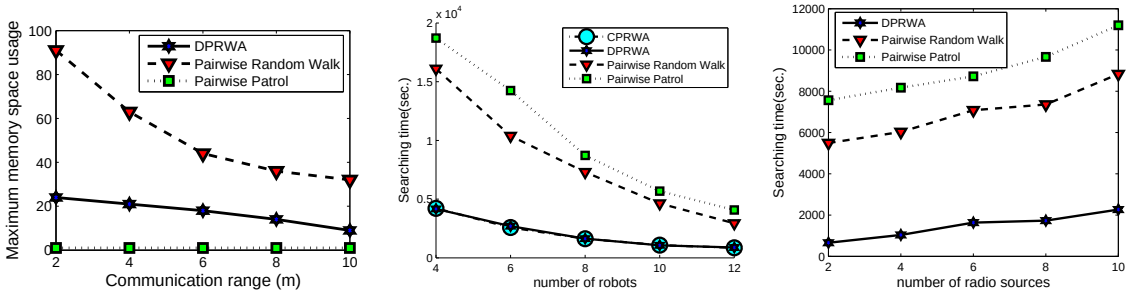


Fig. 33.: Experiment results when comparing the DPRWA, the pairwise random walk, and the pairwise fixed-route patrol (a) The maximum memory space usage using number of detection events stored on-board. (b) and (c) Searching time comparison while changing number of robots (b) or number of radio sources (c).

E. Experiments

To validate the algorithm, we have implemented the algorithm and a simulation platform. The radio sources are XBee Pro with ZigBeeT/802.15.4 OEM radio frequency modules produced by Digi International Inc. We use the RSS readings from XBee Pro to drive the simulation experiments. We cannot use real robots because we do not have robots that can patrol the searching field of about a football field size for more than 20 hours, which is a basic requirement for a realistic setup. We simulate iRobot Create in the process, which has a maximum speed of 40 cm/s. The grid is a square with 50×50 cells. Each grid cell has a size of 50.0×50.0 cm^2. Each radio source generates radio transmission signals according to an *i.i.d.* Poisson process with a rate of $\lambda = 0.05$ packets per second. The radio sources also dynamically vary their transmission power using one of 5 power settings in XBee Pro, which results in a varying sensing range from 1.67 to 3.45 meters. We set $\tau_0 = 500$ seconds in the simulation. We choose the probability convergence threshold as $p_t = 0.9$. During

each trial, we randomly generate radio source locations in the grid.

We compare the DPRWA algorithm to two heuristics including a pairwise random walk and a pairwise patrol. In both heuristics, robots are paired just as DPRWA does. In the former, each pair is treated as a super robot to perform a random walk together. In the later, robot pairs follow a linear formation with an equal inter-pair distance to be the maximum communication distance. Since global connectivity is maintained for the pairwise patrol, it degenerates to the centralized planning.

Fig. 33 illustrates the simulation results by using memory usage and searching time as metrics while changing communication range, number of robots, and number of radio sources. Each data point is an average of 20 independent trials. There are six radio sources to be searched in Figs. 33a and 33b. In Figs. 33a and 33c, eight robots are employed. The communication range is set to be six meters in Figs. 33b and 33c. Since the pairwise patrol maintains global connectivity, it requires the least amount memory for synchronization purpose. The pairwise random walk is the opposite because the time between checkpoints for robots can be very long. Our DPWRA requires more memory than that of the patrol but still much less than that of the random walk (see Fig. 33a). When it comes to the searching time, DPRWA is significantly faster than its counterparts (see Figs. 33b and 33c). The advantage is even more when number of robots are limited, which happens when the search is constrained by resources. Fig. 33b also compares DPRWA with the centralized PRWA (CPRWA) in Chapter VI. It is surprising that DPRWA EST is about the same as CPRWA despite the advantage that CPRWA has in coordination and synchronization. Fig. 33c further confirms Theorem 4 that the EST of DPRWA is insensitive to number of radio sources. In conclusion, DPRWA successfully trades a modest amount of memory for the fastest searching time among the three methods along with excellent scalability due to its invariance to number of radio sources.

F. Conclusion

We developed a decentralized algorithm to coordinate a group of mobile robots to search for unknown and transient radio sources in an open field under mobility, communication range, and sensing constraints. Building on our prior work, we proposed a two-step approach: first we decentralized belief functions that robots use to track source locations using checkpoint-based synchronization, and second we proposed a decentralized planning strategy to coordinate robots to ensure the existence of checkpoints and coordinated searching. We formally analyzed memory usage, data amount in communication, and searching time for the proposed algorithm. We implemented the proposed algorithm and compared it with two heuristics in simulation based on real sensory data. Our algorithm successfully performed the searching task and showed a clear advantage in searching time without significant increase of memory usage.

CHAPTER VIII

CONCLUSION AND FUTURE WORK

A. Contributions

1. Searching for a Single Target

One contribution is to bring analytical results for the searching time to interpret well-known searching strategies. Our EST metric can reveal our common believes about existing searching methods and predict how the effectiveness of those methods changes as trajectory selection, sensing range, searching space size, and robot distribution change. Built on the latest development in random walk in constrained space from stochastic modeling community, our analysis, for the first time, reveals that the traditional slap method ($\Theta(a^2)$) actually is asymptotically faster than the random walk ($O(a^2 \log a)$) for a squared searching space of the size length a, which is never known before. In the second case, we compare a team of n homogenous low-cost robots with a super robot that has the sensing coverage equal to the summation of the n low-cost robots. The EST analysis shows that the low-cost robot team outperforms the super robot because its EST is $\Theta(1/n)$ while the EST for the super robot is $\Theta(1/\sqrt{n})$ as $n \to \infty$. Again, this new analytical result has not been seen before and is important for developing new searching strategies.

2. SRMT Localization

We reported the CSMA SRMT localization scheme to use a single mobile robot equipped with a directional antenna to localize unknown networked radio targets. We proposed a particle filter-based localization approach that combines a CSMA model and a directional antenna model. We also proposed a motion planning algo-

rithm based on the particle distribution. The sensing algorithm runs in $O(n)$ time for n particles and the motion planning algorithm runs in $O(nl)$ time for l radio sources and n particles. We have implemented the algorithm and tested it using a real data-driven simulation platform. The results show that the algorithm is capable of localizing unknown networked radio targets. The experiment results have shown that the proposed localization method is faster, more accurate, and more robust than the two other heuristic methods.

Moreover, we reported our system and algorithm developments of the protocol independent localization method that enable a mobile robot equipped with a directional antenna to localize unknown transient radio targets. We modeled the radio transmission activities using an SPOG and proposed an SPOG update algorithm and an RWA algorithm for robot motion planning. We tested the algorithm using simulation with the data from the real hardware. In the experiment, we compared our algorithms with a random walk and a fixed-route patrol heuristics. Our algorithms showed a consistently superior performance over the two heuristics.

3. MRMT Localization

The MRMT Localization is divided by the centralized and decentralized MRMT localizations. The contributions in the centralized MRMT localization are twofold. First, we formally prove that the sensed conditional joint posterior probability of source locations for the $m-$robot team can be obtained by combining that of pairwise joint posterior probabilities, which are based on RSS ratios and also consider reception range limit. The new sensing model can be combined with the SPOG to address signal correspondence issue. Second, we propose the PRWA to coordinate robot pairs based on the clustering of high probability regions and the minimization of local Shannon entropy. We have implemented the algorithm and tested it under a hardware-driven

simulation and physical experiments. Results show that the PRWA-based localization consistently outperforms the other four heuristics in all settings.

For the decentralized MRMT localization, we proposed decentralized belief functions that robots use to track source locations using checkpoint-based synchronization, and the motion planning strategy to coordinate robots to ensure the existence of checkpoints and coordinated searching.

Through the exploration of the transient target localization, we conclude that proposed localization schemes are able to localize transient targets under different system setting and assumptions. We demonstrate the state-of-art in sensor networks and robotics through the transient target localization. The transient target localization has profound impact in search and rescue, security and military applications. The transient target localization is a promising direction for future sensor network and robotics advances.

B. Future Work

The research on these transient target localizations is still in its infancy. These localization will lead to a rich set of exciting future work. As an extension of localization, we will address different sensor models which have different geometric coverage and range constraints. Also, we also need to consider cases where obstacles clutter the searching space. We will also develop the EST metrics and the SPOG framework for moving targets. As an extension of motion planning, we will address an energy efficient issue.

1. Different Sensor Models

Different sensors have different geometric coverage and range constraints. For example, a regular camera can be described by a pin hole perspective projection model. Its coverage requires direct line of sight. A radio antenna has a unique radiation pattern and signal dispersion function. Most sensor properties can be described by a geometric coverage model associated with a signal dispersion model. We will generalize this property in our new searching time analysis and the SPOG framework.

2. Obstacles

Obstacles exist in many searching processes. We plan to modify the SPOG framework by first marking obstacle-occupied cells as non-traversable. For motion planning, we can still use the idea of RWA but connecting the ridges at the presence of obstacle is different because simple line connections may not work. This can be addressed by using the established path planning techniques. Depending upon the methods used, the path length, which is the most important factor that determines the convergence speed, may vary. Furthermore, the final searching time analysis will reflect the complexity of obstacles. We will also develop estimation methods to predict EST based on the planing results.

3. Moving Targets

The transient targets may move around in the multi-target case, which creates a more difficult signal correspondence issue. Our existing Bayesian framework has been built on the assumption of stationary target and will need substantial new development to handle the moving targets. This will require some prior information about the target such as target trajectories, mass, speed, and dynamics. This is

not unusual because information, such as object mass, size and dynamics, is often available before the searching process starts as we know what we are searching for. We will establish necessary conditions on the feasibility of tracking multiple targets based on the number of moving targets, signal transmission rate, and target dynamics. It is also an interesting direction to explore necessary and sufficient conditions for searching such targets.

4. The Energy Efficient Issue

It takes energy for a robot to search for transient targets due to a limited on-board battery. Our existing approaches have not taken the energy efficiency into consideration. We will incorporate the energy cost model into the planning process. Since the decrease of the information entropy of a distribution means less uncertainty, we plan to develop an efficiency objective function based on the information entropy decrease for source locations per unit energy usage. Using the objective function, we can model the energy-efficient planning problem as an optimal control problem. We will develop algorithms to solve the energy-efficient planning problem.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, pp. 393–422, 2002.

[2] K. Römer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 54–61, December 2004.

[3] H. Hertz, *Electric Waves*. Ithica, New York: Cornell University Library, 1893.

[4] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *INFOCOM'00*, 2000, pp. 775–784.

[5] B. H. Wellenhoff, H. Lichtenegger, and J. Collins, *Global Positioning System: Theory and Practice*. Wien, Austria: Springer-Verlag, 1997.

[6] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," in *Proceedings of the 7th annual international conference on Mobile computing and networking*, ser. MobiCom '01. New York, NY, USA: ACM, 2001, pp. 166–179.

[7] Y. Han, E. Stuntebeck, J. Stasko, and G. Abowd, "A visual analytics system for radio frequency fingerprinting-based localization," in *Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on*, Oct. 2009, pp. 35 –42.

[8] E. Gelenbe, N. Schmajuk, J. Staddon, and J. Reif, "Autonomous search by robots and animals: A survey," *Robotics and Autonomous Systems*, vol. 22, no. 1, pp. 23–34, 1997.

[9] L. Stone, Ed., *Theory of Optimal Search, 2nd Edition.* New York, New York, USA: Military Applications Section, Operations Research Society of America, 1989.

[10] K. Trummel and J. Weisinger, "The complexity of the optimal searcher path problem," *Operations Research*, vol. 34, no. 2, pp. 324–327, Mar.-Apr. 1986.

[11] H. Choset, "Coverage for robotics a survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1-4, pp. 113–126, Oct. 2001.

[12] S. Martinez, F. Bullo, J. Cortes, and E. Frazzoli, "On synchronous robotic networks part i: Models, tasks, and complexity," *Automatic Control, IEEE Transactions on*, vol. 52, no. 12, pp. 2199 –2213, Dec. 2007.

[13] M. Schwager, D. Rus, and J. Slotine, "Decentralized, adaptive control for coverage with networked robots," *International Journal of Robotics Research*, vol. 28, no. 3, pp. 357–375, March 2009.

[14] M. Kao, J. Reif, and S. Tate, "Searching in an unknown environment: An optimal randomized algorithm for the cow-path problem," *Information and Computation*, vol. 131, no. 1, pp. 63–79, Nov. 1996.

[15] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry, Algorithms and Applications.* New York, New York, USA: Springer, 1991.

[16] S. Eidenbenz, C. Stamm, and P. Widmayer, "Inapproximability results for guarding polygons and terrains," *Algorithmica*, vol. 31, no. 1, pp. 79–113, 2001.

[17] F. Preparata and M. Shamos, *Computational Geometry: An Introduction.* Springer-Verlag, Berlin, 1985.

[18] J. Latombe, *Robot Motion Planning*. Kluwer Acadmic, Boston, MA, 1991.

[19] A. Beck and M. Beck, "Son of the linear search problem," *Israel Journal of Mathematics*, vol. 48, no. 2-3, pp. 109–122, 1984.

[20] A. López-Ortiz and S. Schuierer, "Online parallel heuristics and robot searching under the competitive framework," in *SWAT '02: Proceedings of the 8th Scandinavian Workshop on Algorithm Theory*, 2002, pp. 260–269.

[21] R. Baeza-Yates, J. Culberson, and J. Rawlins, "Searching in the plane," *Information and Computation*, vol. 106, no. 2, pp. 234–252, Oct. 1993.

[22] S. Benkoski, M. Monticino, and J. Weisinger, "A survey of the search theory literature," *Naval Research Logistics*, vol. 38, no. 4, pp. 469–494, August 1991.

[23] S. Thrun, "Robotic mapping: A survey," in *Exploring Artificial Intelligence in the New Millenium*, G. Lakemeyer and B. Nebel, Eds. Morgan Kaufmann, 2002.

[24] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.

[25] D. Hähnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose, "Mapping and localization with RFID technology," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, 2004.

[26] K. Murphy, "Bayesian map learning in dynamic environments," in *In Neural Info. Proc. Systems (NIPS)*. MIT Press, 1999, pp. 1015–1021.

[27] D. Fox, S. Thrun, F. Dellaert, and W. Burgard, "Particle filters for mobile robot localization," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. New York: Springer Verlag, 2000.

[28] C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," *The International Journal of Robotics Research*, vol. 26, no. 9, pp. 889–916, September 2007.

[29] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, pp. 46–57, June 1989.

[30] H. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Mag.*, vol. 9, pp. 61–74, July 1988.

[31] S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.

[32] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA, USA: MIT Press, 2005.

[33] K. Laventall and J. Cortes, "Coverage control by multi-robot networks with limited-range anisotropic sensory," *International Journal of Control*, vol. 82, no. 6, pp. 1113 – 1121, June 2009.

[34] P. Brass, "Bounds on coverage and target detection capabilities for models of networks of mobile sensors," *ACM Transactions on Sensor Networks (TOSN)*, vol. 3, no. 2, p. 9, June 2007.

[35] M. Batalin and G. Sukhatme, "The design and analysis of an efficient local algorithm for coverage and exploration based on sensor network deployment," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 661–675, August 2007.

[36] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart, "Distributed multirobot exploration and mapping," *Proceedings of The IEEE*, vol. 94, no. 7, pp. 1325–1339, July 2006.

[37] W. Burgard, M. Moors, C. Stachniss, and F. Schneider, "Collaborative multi-robot exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, June 2005.

[38] J. Letchner, D. Fox, and A. LaMarce, "Large-scale localization from wireless signal strength," in *Proc. of the National Conference on Artificial Intelligence (AAAI-05)*, 2005.

[39] N. Malhotra, M. Krasniewski, C. Yang, S. Bagchi, and W. Chappell, "Location estimation in ad hoc networks with directional antennas," in *ICDCS '05: Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 633–642.

[40] M. Youssef, A. Agrawala, and U. Shankar, "Wlan location determination via clustering and probability distributions," in *IEEE PerCom 2003*, 2003, p. 143.

[41] G. Mao, B. Fidan, and B. Anderson, "Wireless sensor network localization techniques," *Computer Networks*, vol. 51, no. 7, pp. 2529–2553, 2007.

[42] N. Bulusu, J. Heidemann, and D. Estrin, "Gps-less low cost outdoor localization for very small devices," *IEEE Personal Communications Magazine*, vol. 7, no. 5, pp. 28–34, October 2000.

[43] X. Ji and H. Zha, "Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling," in *INFOCOM 2004. Twenty-third AnnualJoint*

Conference of the IEEE Computer and Communications Societies, vol. 4, March 2004, pp. 2652 – 2661 vol.4.

[44] K. Lorincz and M. Welsh, "MoteTrack: A Robust, Decentralized Approach to RF-Based Location Tracking." in *LoCA*, ser. Lecture Notes in Computer Science, vol. 3479. Springer, 2005, pp. 63–82.

[45] M. Sichitiu and V. Ramadurai, "Localization of wireless sensor networks with a mobile beacon," in *first IEEE International conference on Mobile Ad hoc and Sensor Systems*, 2004, pp. 174 – 183.

[46] N. Bulusu, V. Bychkovskiy, D. Estrin, and J. Heidemann, "Scalable, ad hoc deployable rf-based localization," in *Grace Hopper Celebration of Women in Computing Conference 2002, Vancouver, British Columbia, Canada.* University of California at Los Angeles, October 2002. [Online]. Available: http://www.cs.ucla.edu/ bulusu/papers/Bulusu02a.html

[47] D. Koutsonikolas, S. Das, and Y. Hu, "Path planning of mobile landmarks for localization in wireless sensor networks," *Computer Comunications*, vol. 30, pp. 2577–2592, 2007.

[48] T. Sit, Z. Liu, M. A. Jr., and W. Seah, "Multi-robot mobility enhanced hop-count based localization in ad hoc networks," *Robotics and Autonomos Systems*, vol. 55, pp. 244–252, 2007.

[49] B.-C. Liu, K.-H. Lin, and J.-C. Wu, "Analysis of hyperbolic and circular positioning algorithms using stationary signal-strength-difference measurements in wireless communications," *Vehicular Technology, IEEE Transactions on*, vol. 55, no. 2, pp. 499 – 509, Mar. 2006.

[50] N. Patwari, I. Hero, A.O., M. Perkins, N. Correal, and R. O'Dea, "Relative location estimation in wireless sensor networks," *Signal Processing, IEEE Transactions on*, vol. 51, no. 8, pp. 2137 – 2148, Aug. 2003.

[51] Y. Sun, J. Xiao, and F. Cabrera-Mora, "Robot localization and energy-efficient wireless communications by multiple antennas," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, St Louis, MO, USA*, Oct. 2009.

[52] D. Li, K. Wong, Y. H. Hu, and A. Sayeed, "Detection, classification, and tracking of targets," *Signal Processing Magazine, IEEE*, vol. 19, no. 2, pp. 17 –29, Mar. 2002.

[53] M. Kim and N. Y. Chong, "Direction sensing rfid reader for mobile robot navigation," *IEEE Transactions on Automation Science and Engineering*, vol. 6, no. 1, pp. 44–54, January 2009.

[54] T. Berg and H. Durrant-Whyte, "Distributed and decentralized estimation," in *Intelligent Control and Instrumentation, 1992. SICICI '92. Proceedings., Singapore International Conference on*, vol. 2, Feb. 1992, pp. 1118 –1123.

[55] S. Grime and H. Durrant-Whyte, "Data fusion in decentralized sensor networks," *Control Engineering Practice*, vol. 2, no. 5, pp. 849 – 863, 1994.

[56] H. F. Durrant-Whyte, "Data fusion in sensor networks," in *AVSS*, 2006, p. 39.

[57] S. Roumeliotis and G. Bekey, "Distributed multirobot localization," *Robotics and Automation, IEEE Transactions on*, vol. 18, no. 5, pp. 781 – 795, Oct. 2002.

[58] E. Nerurkar, S. Roumeliotis, and A. Martinelli, "Distributed maximum a posteriori estimation for multi-robot cooperative localization," in *Robotics and Au-*

tomation, 2009. ICRA '09. IEEE International Conference on, May 2009, pp. 1402 –1409.

[59] K. Leung, T. Barfoot, and H. Liu, "Decentralized localization of sparsely-communicating robot networks: A centralized-equivalent approach," *Robotics, IEEE Transactions on*, vol. 26, no. 1, pp. 62 –77, Feb. 2010.

[60] ——, "Distributed and decentralized cooperative simultaneous localization and mapping for dynamic and sparse robot networks," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, May 2011, pp. 3841 –3847.

[61] E. Nerurkar and S. Roumeliotis, "Asynchronous multi-centralized cooperative localization," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, Oct. 2010, pp. 4352 –4359.

[62] J. Capitan, L. Merino, F. Caballero, and A. Ollero, "Delayed-state information filter for cooperative decentralized tracking," in *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, May 2009, pp. 3865 –3870.

[63] G. A. S. Pereira, G. A. S. Pereira, A. K. Das, V. Kumar, and M. F. M. Campos, "Decentralized motion planning for multiple robots subject to sensing and communication constraints," in *in Proceedings of the Second MultiRobot Systems Workshop*. Kluwer Academic Press, 2003, pp. 267–278.

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

[65] S. Martinez, F. Bullo, J. Cortes, and E. Frazzoli, "On synchronous robotic networks part ii: Time complexity of rendezvous and deployment algorithms," *Au-*

tomatic Control, IEEE Transactions on, vol. 52, no. 12, pp. 2214 –2226, Dec. 2007.

[66] S. Ross, *Introduction to Probability Models, Ninth Edition*. Burlington, MA, USA: Academic Press, 2007.

[67] F. Al-Athari, "Estimation of the mean of truncated exponential distribution," *Journal of Mathematics and Statistics*, vol. 4, no. 4, pp. 284–288, 2008.

[68] M. Brummelhuis and H. J. Hilhorst, "How a random walk covers a finite lattice," *Physica A: Statistical Mechanics and its Applications*, vol. 185, no. 1-4, pp. 35–44, June 1992.

[69] S. Condamin, O. Bénichou, and M. Moreau, "First-passage times for random walks in bounded domains," *Physical Review Letters*, vol. 95, no. 26, p. 260601, Dec. 2005.

[70] J. D. Noh and H. Rieger, "Random walks on complex networks," *Physical Review Letters*, vol. 92, no. 11, p. 118701, Mar. 2004.

[71] S. Condamin, O. Benichou, V. Tejedor, R. Voituriez, and J. Klafter, "First-passage times in complex scale-invarant media," *Nature*, vol. 450, pp. 77–80, Nov. 2007.

[72] L. Lovasz, "Random walks on graphs: A survey," in *Combinatorics: Paul Erdos is Eighty*, D. Miklos and V. T. Sos, Eds. Janos Bolyai Mathematical Society, 1993, vol. 2, pp. 1–46.

[73] N. Balakrishnan and A. Cohen, *Order Statistics & Inference: Estimation Methods*. Burlington, MA, USA: Academic Press, 1991.

[74] C. Rose and M. Smith, "Computational order statistics," *The Mathematica Journal*, vol. 9, no. 4, pp. 790–802, 2005.

[75] R. S. Elliott, *Antenna Theory and Design*. Hoboken, New Jersey, USA: The IEEE Press, 2003.

[76] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," Tech. Rep., 2008.

[77] W. L. Stutzman and G. A. Thiele, *Antenna Theory and Design*. New York, New York, USA: John Wiley & Sons, Inc., 2003.

[78] S. Lin, "Computer solutions of the traveling-salesman problem," *Bell System Technology Journal*, vol. 44, pp. 2245–2269, 1965.

[79] S. Arora, "Polynomial time approximation schemes for euclidean tsp and other geometric problems," *Journal of the ACM*, vol. 45, no. 5, pp. 753–782, 1998.

[80] Z. Zaidi and B. Mark, "Real-time mobility tracking algorithms for cellular networks based on kalman filtering," *Mobile Computing, IEEE Transactions on*, vol. 4, no. 2, pp. 195–208, March-April 2005.

[81] J. A. Hartigan and M. Wong, "A k-means clustering algorithm," *Applied Statistics*, no. 28, pp. 100–108, 1979.

[82] A. Leon-Garcia, *Probability and Random Processes for Electrical Engineering*, 2nd ed. Menlo Park, CA, USA: Addison-Wesley Pub Co, 1993.