Geometric filter algorithms for device-free localization using received-signal strength in wireless sensor networks

Talampas, Marc Caesar R.

2016

Talampas, M. C. R. (2016). Geometric filter algorithms for device-free localization using received-signal strength in wireless sensor networks. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/69011>

<https://doi.org/10.32657/10356/69011>

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

**GEOMETRIC FILTER ALGORITHMS FOR DEVICE-FREE
LOCALIZATION USING RECEIVED-SIGNAL STRENGTH IN
WIRELESS SENSOR NETWORKS**

**MARC CAESAR R. TALAMPAS
SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING
2016**

**Geometric Filter Algorithms for Device-free
Localization using Received-Signal Strength in
Wireless Sensor Networks**

Marc Caesar R. Talampas

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2016

Acknowledgments

I would not be at this stage of my academic life without the support of a number of people. I would like to express my gratitude to:

The DOST-ERDT program and the University of the Philippines, for the financial support throughout my graduate studies in Singapore.

The students and staff at the Satellite Research Centre, for warmly receiving and accommodating me during my stay. It was a pleasure working in the company of such motivated and talented people.

My supervisor, Dr. Kay-Soon Low, for giving me space to pursue my research interests, and for providing valuable feedback and firm guidance on my work. I am honored to have completed my Ph.D. under his supervision.

My family — parents Cesar and Noemi, and siblings Yey, Kel, Ninay, and Yayus. I am thankful for their quiet encouragement, and for taking full advantage of every opportunity for us to connect while I was away.

And lastly, my wife, Candice, for not letting distance get in the way of expressing her love and support, and for sharing in all the trials, tribulations, and triumphs that have culminated at this point.

Contents

1	Introduction	13
1.1	Received-signal Strength-based Localization	13
1.2	Motivation	15
1.3	Objectives	19
1.4	Main Contributions	19
1.5	Outline of the Thesis	20
2	Literature Review	22
2.1	Introduction	22
2.2	Link-based DFL Algorithms	24
2.2.1	RSS change detection	24
2.2.2	Geometric methods	27
2.2.3	Radio tomographic imaging	29
2.2.4	Statistical inversion methods	34
2.3	Proposed DFL Algorithm	40
2.4	Summary	41
3	Geometric Filter Algorithm for DFL	44
3.1	Problem statement	45
3.2	Proposed Geometric Filter Algorithm	47

3.2.1	Detecting the target-affected links	47
3.2.2	Defining the prior region	48
3.2.3	Link filter (LF)	49
3.2.4	Solving for the link intersections and obtaining RSS- based weights	51
3.2.5	Point filter (PF)	53
3.2.6	Distance-based weights	55
3.2.7	Generating a location estimate	55
3.3	Experimental Setup	56
3.4	Scene 1, Uncluttered outdoor environment	60
3.4.1	Results	61
3.4.2	Performance under different values of γ and r	66
3.5	Scene 2, Cluttered indoor environment	68
3.6	Average execution time	71
3.7	Summary	73
4	Multi-Channel Geometric Filter Algorithm	74
4.1	Motivation	75
4.2	Problem statement	76
4.3	Proposed Multi-Channel Geometric Filter algorithm	78
4.3.1	Calibration period	79
4.3.2	Detecting the target-affected links	81
4.3.3	Prior region	83
4.3.4	Outlier link filtering	83
4.3.5	Generating probable target locations and RSS-based weights	84
4.3.6	Generating fade level-based weights	84

4.3.7	Removing improbable target locations	85
4.3.8	Distance-based weights	86
4.3.9	Overall algorithm in generating a location estimate .	86
4.4	Experiment setup	87
4.4.1	Scene 1, Uncluttered outdoor experiment	90
4.4.2	Scene 2, Highly cluttered indoor experiment	90
4.5	Experimental results	90
4.5.1	Performance under different values of r	95
4.5.2	Performance under different values of C and P_n . . .	96
4.5.3	Effect of P_n and the link filter	100
4.5.4	Effect of fade level-based weights w_F^{i+}	101
4.5.5	Average execution time	101
4.6	Summary	103
5	Conclusion and Future Work	104
5.1	Summary	104
5.2	Future Work	106

Summary

Device-free localization (DFL) is a method of determining the location of a target without requiring the target to wear a device or tag. This capability to track a device-free target is useful in applications where the target may be uncooperative and unwilling to be located and monitored. In radio frequency-based DFL systems that use received-signal strength (RSS) measurements, the changes induced by the target's presence or motion on the RSS of the network's links are used to infer his location. A number of RSS-based DFL algorithms have been recently proposed that can locate and track a target accurately, albeit with high computational requirements. This thesis presents new DFL algorithms that have lower computational costs while able to track a single device-free target with high accuracy.

In this thesis, a new single target RSS-based DFL algorithm, referred to as the “Geometric Filter” (GF) algorithm is proposed. The GF algorithm uses simple geometric objects to represent radio links, probable target locations, and locational filters. The intersection points of line segments representing the target-affected links are used as probable locations of the device-free target. A locational filter is used to remove outlier links and points. Information about the target's prior location and induced RSS changes are used to further refine the target location estimates.

In order to perform accurate tracking in multipath-rich environments, the GF algorithm was extended further to utilize channel diversity. The “Multi-Channel Geometric Filter” (MCGF) fuses measurements of the RSS changes of each link across different frequency channels, and uses link-specific thresholds to detect the target-affected links. The measurements are then processed by a modified GF algorithm that uses estimates of the overall fade levels of intersecting links as weights to generate the target location estimates.

The GF and MCGF algorithms have been evaluated using single-target tracking experiments in both indoor and outdoor environments. In these experiments, the new algorithms have been shown to outperform existing DFL algorithms in both tracking accuracy and execution time.

List of Acronyms

BGA Bayesian grid approach

CPU central processing unit

DFL Device-free localization

ECDF empirical cumulative distribution function

ER exponential-Rayleigh

GF Geometric Filter

ID identification

LF link filter

LOS line-of-sight

MCGF Multi-Channel Geometric Filter

NLOS non-line-of-sight

PF point filter

RAM random access memory

RF radio frequency

RMSE root-mean-square error

RSS received-signal strength

RTI radio tomographic imaging

SIR sequential importance resampling

WSN wireless sensor networks

List of Figures

1.1	(a) Location of wireless nodes and target. (b) Output of an RTI-based approach. (c) Probable target locations generated by a statistical inversion-based DFL approach which uses a particle filter framework. (d) Detected links and probable target locations generated by a geometric approach.	16
2.1	Illustration of a link's multipath components when (a) the target is not present, (b) the target is obstructing the LOS, and (c) the target is off-LOS. Due to the target's presence, some multipath components are affected (—) and new multipath components are created (—). Figure adapted from [14].	25
2.2	(a) Midpoint, (b) intersection, and (c) best-cover algorithms [26], [27].	28
2.3	Comparison of fits for the (a) exponential and (b) magnitude models [32].	38
3.1	A DFL system with 16 nodes at the perimeter. The presence of the target affects the RSS of some links in the network. .	46

3.2	An illustration of the proposed algorithm. (a) Target-affected links and outlier links. (b) The outlier links are removed. (c) The intersection points of the remaining target-affected links. (d) Probable target locations outside the prior region are removed, and (e) a location estimate is generated using the weighted mean of the remaining points.	58
3.3	Scene 1, uncluttered outdoor experiment setup.	60
3.4	Scene 1 deployment area.	61
3.5	The mean tracking results for Scene 1, overlaid on the true target track.	62
3.6	Mean tracking errors vs time, obtained using the GF algorithm for Scene 1.	62
3.7	ECDF of tracking errors for each algorithm.	63
3.8	ECDF of tracking errors for GF with the locational filters disabled.	63
3.9	ECDF of tracking errors for GF variants with different weight schemes.	65
3.10	Error bars of tracking errors as the detection threshold γ is varied. The central marks indicate the median of tracking errors. The lower and upper bars indicate the 25th and 75th percentiles, respectively.	67
3.11	Error bars of tracking errors as the radius of prior region r is varied. The central marks indicate the median of tracking errors. The lower and upper bars indicate the 25th and 75th percentiles, respectively.	67
3.12	Scene 2, cluttered indoor experiment setup.	68

3.13 (a) Overall view of the cluttered indoor environment for Scene 2.	
2. (b) The target walking along the pre-defined path in Scene 2.	69
3.14 Mean tracking results for Scene 2, overlaid on the true target track.	70
4.1 Visualized tracking results for (a) Scene 1 and (b) Scene 2 using the MCGF algorithm.	92
4.2 Scene 1, ECDF of tracking errors for the multi-channel and single-channel GF variants.	93
4.3 Scene 1, ECDF of tracking errors for the multi-channel GF and RTI variants.	93
4.4 Scene 2, ECDF of tracking errors obtained using the tested algorithms.	94
4.5 Tracking RMSE of MCGF as α is varied for (a) Scene 1 and (b) Scene 2.	97
4.6 Contour plots of the tracking RMSE of MCGF as P_n and C are varied for (a) Scene 1 and (b) Scene 2.	98
4.7 Plots of the tracking RMSE for MCGF as C is varied for (a) Scene 1 and (b) Scene 2. The dashed horizontal lines indicate the RMSE levels for fixed values of γ_i	99
4.8 $ \mathbf{L}'(t) / L_{\mathbf{P}(t)}$ as P_n is varied for both scenes.	100

List of Tables

2.1	Comparison of link-based DFL algorithms.	43
3.1	Scene 1, Tracking error statistics	61
3.2	Tracking error statistics using different weighting schemes for GF	64
3.3	Scene 2, Tracking error statistics	71
3.4	Average execution times	73
4.1	Tracking error statistics - Scene 1	91
4.2	Tracking error statistics - Scene 2	94
4.3	Effect of w_F^{i+} on RMSE of MCGF	101
4.4	Average execution times	102

Chapter 1

Introduction

1.1 Received-signal Strength-based Localization

The study and use of wireless sensor networks (WSN) in a variety of applications has steadily grown over the years. This can be attributed to developments in radio, battery, and sensor technology allowing for more capable sensor nodes at costs that continue to decrease. As WSN technology continues to improve, researchers have found new and exciting applications that are enriched by the use of WSNs. WSNs are essentially a collection of sensor devices that are typically battery-powered and can communicate with each other wirelessly. Due to their small size and low cost as compared to traditional data logging systems, wireless sensor nodes can be deployed at high densities, allowing a researcher to observe phenomena of interest at high spatial and temporal resolutions. WSNs have been used in applications such as gas leak detection [1], electro-static discharge event detection [2], item localization [3], radio propagation model calibration [4], and E-health

[5]. In practically all applications of WSNs, location information is a critical requirement. Thus, extensive research has been conducted in the field of localization in WSNs.

A popular approach to wireless localization is through the use of radio received-signal strength (RSS) measurements between nodes in the network. The popularity of RSS for localization is due mostly to its high availability and low cost, as most modern transceivers are already equipped with the circuitry to provide RSS measurements and automatically measure the RSS during normal communication. Typically, the target to be tracked is equipped with a radio device or tag, and RSS measurements between the target's device and a set of known-location nodes are used to determine the target's location. Such a system, wherein the target's device willingly exchanges information with the anchor nodes of the WSN has been called "active localization". Techniques such as lateration [6], particle swarm optimization [7], [8], weighted centroid localization [9], particle filtering [10], radio fingerprinting [11]–[13], etc., have been used for active localization. Active localization systems work well in applications wherein the target or object to be monitored is cooperative, i.e. it is possible to attach a radio device or tag to the object to be monitored, and the object is willing to be located.

However, in some applications where it is desired to track the position of an uncooperative target, equipping the target with a radio device is not practical nor possible. To address these scenarios, techniques for passive localization or "device-free localization (DFL)" as it has become popularly known, have been recently devised for tracking a target without the need to wear a radio device [14]. In these applications, DFL systems determine the

location of the human target inside a WSN's deployment area by measuring the changes induced by the target on the RSS of the WSN's links. Due to the dense concentration of links that cross the WSN's deployment area in typical DFL systems, the target is likely to affect the RSS of multiple links, allowing its location to be accurately inferred.

RSS-based DFL is envisioned to enrich a large number of applications, especially those that require localization indoors. A potential application of DFL is in E-health systems, wherein a wearable wireless sensor is usually used for tracking the activities and location of an elderly person. This is an application wherein the target may be uncooperative as not every elderly person likes to wear a wireless sensor. There may also be cases when the elderly person forgets to wear or turn on the wireless sensor. Using conventional active localization systems may fail in such cases, and DFL systems can be used instead. Another potential application of DFL is in security, wherein DFL systems can extend the coverage of camera and tripwire-based systems due to the ability of radio frequency waves to penetrate through smoke and nonmetal walls [14], [15], and to work in darkness. DFL is also envisioned to be used in applications such as home and building automation, emergency response, and rescue operations. The accuracy of existing systems that track device-equipped targets can also be improved by using DFL as a complementary technology [16].

1.2 Motivation

Due to the variety of promising applications that can be enriched by DFL, a number of RSS-based DFL algorithms have been proposed in recent years. These include methods such as radio tomographic imaging (RTI) [17]–[25],

Figure 1.1: (a) Location of wireless nodes and target. (b) Output of an RTI-based approach. (c) Probable target locations generated by a statistical inversion-based DFL approach which uses a particle filter framework. (d) Detected links and probable target locations generated by a geometric approach.

geometric methods [26]–[28], and statistical inversion methods [29]–[36]. These methods are able to achieve good localization performance in both indoor and outdoor environments.

Fig. 1.1a shows a typical setup of a WSN for the purpose of DFL. In this example, sixteen wireless nodes are evenly placed at the perimeter of an $8\text{m} \times 8\text{m}$ area. In a DFL network, it is assumed that all nodes are able to communicate with every other node on a single hop. While the area is

vacant, i.e. in the absence of the target, a baseline measurement of the RSS of all links is obtained. The change between the baseline measurement and the RSS of the links with the target present inside the deployment area is used to estimate the target's location. In Fig. 1.1a, the target is at location (3.75,6.25), indicated by the red circle. Figs. 1.1b, 1.1c, and 1.1d illustrate the RTI, statistical inversion, and geometric methods for DFL, respectively.

RTI-based methods estimate an image of the change in the RF propagation field due to the target's presence. It can be seen that RTI-based methods do not give a direct estimate of the target's coordinates, and an additional step is required to convert the generated image into a coordinate estimate. Typically, the grid location with the highest intensity is used as a target location estimate. The division of the deployment area into a grid results in an inherent quantization error in RTI-based methods. Thus, the computed location estimates are usually processed further using a Kalman filter to smooth the estimated target trajectory.

For the statistical inversion method, it uses a particle filter framework with measurement models that relate the target's closeness to the link with its effect on the link RSS. Probable target locations are represented by particles and their corresponding weights, which are measures of how closely the positions represented by the particles agree with measurements of the RSS change at the current time instant. A coordinate estimate of the target's location is obtained via the weighted sum of the particles. Particle filters overcome the restrictive assumptions made by the Kalman filter on the state space dynamics. However, this flexibility comes at the cost of increased computational requirements. The observational models used in particle filter-based DFL methods are typically complex to evaluate, and a

large number of particles is usually required for good tracking performance.

In a class of geometric methods for DFL, the probable target locations are intersections of the links sufficiently affected by the target's presence. A coordinate estimate is generated in a similar manner as the statistical inversion-based methods, i.e. through a weighted sum of the intersection points. Geometric methods are simple to implement and have lower computational requirements compared to RTI-based and statistical inversion-based methods. It is well known that the geometric methods are prone to noise, resulting in links that intersect at points far from the target's true location as can be observed in Fig. 1.1d. These outlier points reduce the tracking accuracy of geometric methods.

In summary, RTI-based and statistical inversion-based methods offer better tracking accuracy at the cost of higher computational requirements. On the other hand, geometric methods have lower computational cost but are also less accurate. Thus, there is a need for a new DFL algorithm that has reduced computational requirements but is still able to achieve high tracking accuracy.

This thesis investigates whether the geometric method for DFL can be improved further, to the extent that it can outperform the more sophisticated RTI-based and statistical inversion-based methods in terms of tracking accuracy, yet maintain a low computational cost. Furthermore, this thesis also investigates the use of channel diversity to improve the tracking accuracy of the geometric method for DFL in multipath-rich propagation environments, the advantages of which were demonstrated by RTI-based DFL [22]–[25].

1.3 Objectives

The objectives of this thesis are as follows:

- Develop a new algorithm for single target device-free localization that uses geometric objects to represent radio links and probable target locations, and uses prior information on the target's estimated location to improve succeeding location estimates. The new algorithm should outperform existing DFL algorithms in terms of accuracy and execution time.
- Extend the proposed algorithm to work with RSS measurements on links operating on multiple frequency channels. The enhanced algorithm should outperform existing RTI-based DFL algorithms that also use channel diversity in terms of accuracy and execution time.
- Investigate the performance of the proposed algorithms under different parameter settings and in both indoor and outdoor environments.

1.4 Main Contributions

The main contributions of this thesis are as follows:

- A new algorithm for single target RSS-based device-free localization, referred to as the Geometric Filter (GF) algorithm, has been developed and evaluated. It uses geometric objects to represent the radio links and probable target locations, and locational filters to improve tracking accuracy. Experimental results have shown that the GF algorithm outperforms existing RTI-based and statistical inversion-based algorithms for DFL in terms of tracking accuracy and execution time.

- The Multi-Channel Geometric Filter (MCGF) algorithm, an extension of the GF algorithm that utilizes channel diversity, has been developed and evaluated. The MCGF algorithm fuses measurements of the RSS changes of each link across different frequency channels, and uses link-specific thresholds to detect the target-affected links. The measurements are then processed by a modified GF algorithm that uses estimates of the overall fade levels of intersecting links as weights to generate the target location estimates. Experimental results have demonstrated that the MCGF algorithm outperforms existing single-channel GF and multi-channel RTI-based methods in terms of tracking accuracy and execution time.

1.5 Outline of the Thesis

The rest of the thesis is structured as follows:

Chapter 2 reviews the related literature on RSS-based device-free localization. The details of the existing DFL algorithms are discussed, along with their advantages and disadvantages. This chapter also positions the proposed methods in relation to the existing DFL algorithms.

Chapter 3 discusses the proposed GF algorithm in detail. The GF algorithm's methods for target-affected link detection, locational filtering, and target location estimation are presented. This chapter also presents the experimental setup and results obtained with the GF algorithm in both indoor and outdoor environments. The tracking and execution time performance of the GF algorithm as its parameters are varied is also investigated in this chapter.

The MCGF algorithm which uses RSS measurements at different fre-

quency channels is introduced in Chapter 4, and the modifications made to the original GF algorithm are discussed in detail. Chapter 4 also presents the results of indoor and outdoor experiments to evaluate the MCGF algorithm and compare it with other DFL methods that also use channel diversity.

Finally, Chapter 5 summarizes this thesis and presents ideas for future research in RSS-based device-free localization.

Chapter 2

Literature Review

2.1 Introduction

Device-free localization (DFL) algorithms are designed to have the capability of tracking a person’s location without requiring the person to wear a radio device or tag. This makes them suitable for applications where the person is uncooperative and unwilling to be tracked and monitored — scenarios which active localization systems are ill-equipped to handle. The usefulness of DFL in such situations, coupled with the ubiquity of devices equipped with received-signal strength (RSS) measurement capabilities, has spurred the development of RSS-based DFL algorithms in recent years. These algorithms can be classified broadly into two types [37]:

- Location-based
- Link-based

Location-based algorithms, also known as “fingerprint-based methods” [37]–[41], construct a database of radio measurements during an extensive offline training phase, wherein a person stands at different locations while RSS

measurements are gathered. During the online phase, the current RSS measurements are then compared with the database entries, and the entry with the closest match to the current measurements is used as the target location estimate. Fingerprint-based methods for DFL take the view of absolute position dependence between the target's location and the changes in RSS. From this perspective, a simple model for the dependence of the RSS on the relative positions of the target and the link's nodes cannot be derived [14]. Fingerprint-based methods have the advantage of avoiding RSS modeling errors due to the explicit inclusion of the fading information in the RSS database entries. Furthermore, these methods require fewer deployed nodes as compared to other schemes. However, substantial effort must be exerted in manual calibration of the system during the offline phase — the dependency of the RSS measurements on the target location for each link across all target locations of interest must be determined. These methods are also sensitive to environmental changes, requiring frequent recalibration of the RSS measurement database [14].

On the other hand, link-based algorithms rely on the statistical relationship between a target's location and the RSS of links in the network. These algorithms take the perspective of relative position dependence [14], i.e. the change in RSS for a link is a function of the relative position of the link's nodes and the target.

Link-based DFL algorithms can be classified into the following types :

- Geometric methods
- Radio tomographic imaging
- Statistical inversion methods

In this thesis, we focus on link-based algorithms for DFL, and discuss these schemes in greater detail in the succeeding section.

2.2 Link-based DFL Algorithms

As previously mentioned, link-based DFL algorithms use the change in RSS to determine which links have been affected by the target, and consequently infer the target's location. A number of different ways to quantify the change in RSS have been proposed, and these methods are discussed briefly before delving into the details of some link-based DFL algorithms.

2.2.1 RSS change detection

A link's RSS is a measurement, in dB units, of the squared magnitude of the phasor sum of the multipath components that make up the radio signal between the link's nodes. Due to the target's presence, a subset of these multipath components will be absorbed, reflected, diffracted, or scattered [42], and additional paths between the link's nodes may also be created. These changes in the link's multipath components inevitably results in changes in the link's RSS. Link-based DFL algorithms detect and quantify this change in RSS and relate it to the target's location with respect to the affected link.

The most commonly used measurement of target-induced RSS change is that of shadowing or attenuation [18], [28], [29], [32], [34]. In order to detect shadowed links due to the target's presence, a baseline measurement of the RSS of all the network's links must be available. A typical approach to obtain the baseline RSS is to take measurements of the network's links during a calibration period, i.e. a time when the target is not present inside

Figure 2.1: Illustration of a link's multipath components when (a) the target is not present, (b) the target is obstructing the LOS, and (c) the target is off-LOS. Due to the target's presence, some multipath components are affected (—) and new multipath components are created (—). Figure adapted from [14].

the monitored region and the environment is static, as illustrated in Fig. 2.1a. The mean RSS of a link during the calibration period is treated as the baseline value with which all succeeding measurements of the RSS will be compared with. A target at or near the link line will obstruct the line-of-sight (LOS) component, as shown in Fig. 2.1b. If the link has a strong LOS component, this obstruction generally results in a decrease in the overall received power of the link. In practical applications of DFL, it may not be possible to have a time window in which the calibration of baseline RSS can be performed. To address this issue, some works have proposed using either the previous sampling period's RSS measurements [33], or a moving average of recent RSS measurements [23], [28], [43] as the baseline RSS.

The variance of a link’s RSS is another metric that is used to sense the target’s location relative to the link [19], [26], [27]. In [19], it was shown that the variance of a link’s RSS is related to the total power of the link’s multipath components that are affected by the target’s motion. Essentially, the link’s RSS variance increases as the target moves and obstructs its different multipath components, as illustrated in Fig. 2.1c. This phenomenon is particularly useful in environments with many non-line-of-sight (NLOS) links. Such environments cannot be handled adequately by attenuation-based DFL, since the mean RSS value of NLOS links are less predictable as compared to LOS links [19]. A drawback of variance-based DFL is that it can only detect targets that are in motion, and not targets that remain still for long periods of time.

Instead of detecting only the change in the mean or variance of the RSS, the work in [21] compares long-term and short-term histograms of the RSS and uses the “kernel distance” metric to quantify the change in the mean, variance, and other features of the RSS distribution due to the target’s presence. This allows [21] to work in both LOS and NLOS environments, for targets that are either stationary or in motion. Using the kernel distance also enables online calibration of the baseline RSS, removing the need for a calibration period when the monitored region is empty. However, it must be noted that [21] will fail to detect a target that has stayed motionless for a time period greater than the memory allotted for the long-term histogram [44].

After quantifying the amount of target-induced RSS change on the network’s links, the target’s location can now be inferred using different methods. These methods are discussed in the succeeding sections.

2.2.2 Geometric methods

Geometric methods for DFL use simple geometric objects such as line segments, rectangles, and points to represent links and probable target locations. The geometric methods for DFL proposed in [26] and extended further in [27] use a regular 2-D grid deployment of sensors mounted on the ceiling of an indoor environment. In these works, the effect of a target on a link's RSS is described by a "signal dynamic model". In this model, a link metric called "RSSI dynamic" is defined as

$$\delta R_i = \frac{\sum_{j=1}^m |b - a_j|}{m}, \quad (2.1)$$

where a_j are the m RSS measurements during the calibration period, and b is the RSS measurement for link i during the online period. The "RSSI dynamic" δR_i quantifies the variation of the RSS measurements, and is essentially the mean absolute deviation of the current RSS measurement and the historical RSS measurements for link i . In [26], it was observed that the value of δR_i is the highest when the target is located at the midpoint of the link line, and falls off as the target moves away from the midpoint. By computing for δR_i of all links and checking which ones exceed a pre-defined threshold, the links that are obstructed by the target can be detected. The work in [26] proposed three algorithms to estimate the target's location using the detected "influential links". In the midpoint algorithm, the weighted centroid of the influential links' midpoints is used as the target location estimate, using the measured δR_i as a basis for the weights. In the intersection algorithm, the weighted centroid of the influential links' intersection points are used instead. The weights are derived from the sum of δR_i of the inter-

Figure 2.2: (a) Midpoint, (b) intersection, and (c) best-cover algorithms [26], [27].

secting influential links. Finally, the best-cover algorithm creates a rectangle for each influential link, with the rectangle's length and width dependent on the link's δR_i . A “scanning square” is then used to determine which section of the monitored region has the most overlapping rectangles. The location of the scanning square at this region is then used as the target location estimate. A distributed version of the best-cover algorithm was presented in [27]. Illustrations of the midpoint, intersection, and best-cover algorithms are shown in Fig. 2.2.

In the works of [26], [27], target tracking is performed by taking snapshots of the target's location for every sampling instant, relying only on the current

measurements of δR_i and without using any prior information on the target's location. This makes the geometric methods of [26], [27] prone to noise, especially in cluttered indoor environments. In [28], prior information on the target's location was used with an intersection point algorithm via a Kalman filter. However, the work in [28] was only tested on data gathered from an uncluttered outdoor environment. While geometric methods that rely on intersection points of target-affected links such as those in [26]–[28] are simple and intuitive to implement, some attenuated links that are outside the target's vicinity may still be detected, especially in multipath rich environments. These outlier links can severely degrade the tracking accuracy of such algorithms, as they may intersect with other links at points far from the target's true location.

2.2.3 Radio tomographic imaging

Radio tomographic imaging (RTI) methods [17]–[25] treat DFL as an ill-posed image reconstruction problem which is solved using regularization. This method was first proposed in [17], which presented the idea of using the correlated shadowing of links in the network to infer the location of an attenuating object. The concept of RTI was further developed and refined in [18], demonstrating its effectiveness in imaging the attenuation caused by a single target on the links of a network deployed in an uncluttered outdoor environment.

Linear formulation

RTI-based methods assume that the change in a link's RSS is a spatial integral of the monitored region's radio propagation field [18]. Using a

discretized model of the propagation field, the change in RSS is formulated as a linear combination of the effect of each volumetric pixel (or voxel) on the link, i.e.

$$y_i = \sum_{j=1}^{N_{vox}} w_{ij} x_j + n_i \quad (2.2)$$

where y_i is the measurement of RSS change for link i , x_j is the change in RSS caused by voxel j , w_{ij} is the weight of voxel j for link i , n_i is the noise on link i , and N_{vox} is the number of voxels in the image. For a network of K wireless sensors, the total number of bidirectional links is $M = \frac{K(K-1)}{2}$. Thus, when all M links are considered, the change in RSS for all links is

$$\mathbf{y} = \mathbf{W}\mathbf{x} + \mathbf{n} \quad (2.3)$$

where $\mathbf{y} \in \mathbb{R}^M$ is the vector of RSS-based measurements, $\mathbf{x} \in \mathbb{R}^{N_{vox}}$ is the image to be estimated, $\mathbf{W} \in \mathbb{R}^{M \times N_{vox}}$ is a weight matrix, and $\mathbf{n}$ is the noise vector for all links. The objective of RTI is to estimate the image $\mathbf{x}$ from the measurements vector $\mathbf{y}$, which can either be measurements of shadowing [17], [18], RSS variance [19], or kernel distance [21].

Weight model

The weight matrix $\mathbf{W}$ used in RTI describes how each of the N_{vox} voxels of the image affects each of the M links. The commonly used model for the weight matrix is the elliptical model [17]–[19], [21]. In this model, the nodes that make up a link i are the foci of an ellipse, and voxels that are inside the ellipse have their weights set to a value inversely proportional to the square root of the link length. Pixels that fall outside the ellipse have their weights

set to zero. Formally, the elements of the weight matrix $\mathbf{W}$ are defined as

$$w_{ij} = \begin{cases} \frac{1}{\sqrt{d_i}} & \text{if } d_{ij}^t + d_{ij}^r - d_i < \lambda \\ 0 & \text{otherwise} \end{cases} \quad (2.4)$$

where w_{ij} is the weight corresponding to link i and voxel j , d_i is the length of link i , d_{ij}^t and d_{ij}^r are the distances from the center of voxel j to the transmitter and receiver of link i , respectively, and λ is the excess path length of the ellipse. The value of λ controls the ellipse width and is usually set to a very low value, such that the ellipse nearly approximates the link line [14]. Since the value of w_{ij} is inversely proportional to the link length d_i , this weighting model assigns higher weights to measurements on shorter links.

Image formation

Given the linear model of (2.3), estimating the image vector $\mathbf{x}$ is an ill-posed inverse problem due to the number of voxels N_{vox} being considerably greater than the number of links M . To solve this problem, a regularization technique is required. In [18], [19], Tikhonov regularization is performed to reduce the noise and smooth the image. In [21], a regularized least-squares approach is used instead, as it was found to outperform Tikhonov regularization. In the regularized least-squares formulation, an estimate $\hat{\mathbf{x}}$ of the image vector is obtained by

$$\hat{\mathbf{x}} = \mathbf{\Pi} \mathbf{y} \quad (2.5)$$

where

$$\mathbf{\Pi} = (\mathbf{W}^T \mathbf{W} + \sigma_n^2 \mathbf{C}_{\mathbf{x}}^{-1})^{-1} \mathbf{W}^T. \quad (2.6)$$

In (2.6), $(\cdot)^T$ is the transpose operation, σ_n^2 is the variance of the link measurement noise, and $\mathbf{C}_{\mathbf{x}}$ is the covariance matrix of the image vector $\mathbf{x}$. The *a priori* covariance matrix $\mathbf{C}_{\mathbf{x}}$ is calculated as

$$[\mathbf{C}_{\mathbf{x}}]_{jk} = \sigma_{\mathbf{x}}^2 \exp\left(\frac{-d_{jk}}{\delta_c}\right) \quad (2.7)$$

where $\sigma_{\mathbf{x}}^2$ is the variance of voxel attenuation, d_{jk} is the distance between the centers of voxels j and k , and δ_c is the voxel's correlation distance. Given that the positions of the K wireless nodes are known and fixed, the linear transformation matrix $\mathbf{\Pi}$ in (2.6) needs to be computed only once. This allows RTI-based methods to estimate the image vector $\hat{\mathbf{x}}$ in real-time. Due to the typically large number of voxels required for accurate localization in RTI, the total number of operations to transform the measurements vector $\mathbf{y}$ into the image $\hat{\mathbf{x}}$ can become very large.

Recently, the Bayesian grid approach (BGA) was proposed in [45] to solve the DFL problem using only lightweight operations on “shadowing effect maps” and employing prior and constraint information to realize a location estimate. Similar to RTI, BGA also uses an elliptical model to represent the links of the network. However, instead of using all M links to build a weight matrix $\mathbf{W} \in \mathbb{R}^{M \times N_{vox}}$ as in RTI, BGA selects only the links that are shadowed by the target. The shadowing state of a link is determined by the magnitude of the difference in its RSS between the baseline state obtained from the calibration period and the current time instant. Essentially, the shadowing effect maps used in BGA are the rows in the weight matrix $\mathbf{W}$ of

RTI which correspond to shadowed links. To generate a location estimate, the shadowed link maps are merged via weighted sum to obtain the feasible region where the target is most likely located. BGA also uses a subset of the unshadowed link maps to build an infeasible region, i.e. voxels where the target is highly unlikely to be located. The feasible and infeasible regions are merged to form the current observation image in BGA. To improve the location estimate, BGA also applies a circular spatial filter which zeros out the voxels in the current observation image that are far from the target's previous estimated location. After application of the spatial filter, the final image is obtained and the target location is estimated as the weighted centroid of the voxel centers with the largest shadowing values.

Due to the fewer operations employed by BGA, it can execute much faster as compared to RTI. However, it still needs to maintain an $M \times N_{vox}$ matrix of shadowing effect maps, which can be considerably large. Furthermore, the $N_{vox} \times 1$ vector representing the circular spatial filter needs to be constructed at every iteration, even for locations where the target is unlikely to be located. Thus, while BGA is faster compared to RTI, its computational requirements are still substantial.

A distinct feature of imaging-based methods such as RTI and BGA is that visualization of the target's location is an inherent part of the algorithm. A disadvantage of this approach is that these methods do not provide direct coordinate estimates of the target's true location, and an intermediate stage has to be performed where data from the image is processed to obtain a coordinate estimate. This is typically done by finding the voxel coordinates where the largest attenuation is observed, and using those coordinates for the target's location estimate. Other works use the average of the voxel lo-

cations with the highest observed attenuation values [45]. These coordinate estimates are then input into a tracking algorithm such as the Kalman filter, for smoothing of the target’s estimated trajectory.

2.2.4 Statistical inversion methods

Statistical inversion methods in [29]–[36] use a target-induced fading model that is dependent on the target’s closeness to the link. These models are derived experimentally and are used with particle filters for tracking the location of the target. The use of statistical models offer some advantages over imaging-based methods such as RTI. As it is no longer required to discretize the monitored region into voxels, the inherent quantization error is avoided. DFL approaches that use statistical inversion methods also do not require an intermediate imaging step before determining the coordinates of the target’s estimated location.

In the succeeding paragraphs, we discuss the details of the models in [31]–[34].

Skew-Laplace model

The skew-Laplace model of target-induced fading was proposed in [31], where the concept of fade level was introduced to quantify the steady-state narrowband fading experienced by a link. With the person absent from the monitored region, links that are experiencing destructive multipath interference are classified as *deep fade* links. Deep fade links tend to display high variance and an increase in the RSS when the target crosses the link’s line-of-sight (LOS). Links that experience constructive multipath interference are classified as *antifade*. Different from deep fade links, antifade links tend to

display lower variance and a decrease in the RSS when the target crosses its LOS. A distance threshold is used to determine if the target is within the link’s LOS. The work in [31] quantifies the fade level of a static link as the difference between its mean RSS during a calibration period where the target is not present, and with the RSS predicted by a path loss model. Typically, the log-distance path loss model [42] is used to compute the predicted RSS.

It was observed in [31] that the attenuation of a link’s RSS depends on the link’s static fade level and the target’s location with respect to the link, i.e. whether the target is on or off the link’s LOS. This relationship was modeled using a skew-Laplace distribution. The use of a particle filter with the skew-Laplace model was demonstrated to effectively track the location of stationary and moving people, even through walls. The work in [31] also suggests that the parameters of the skew-Laplace model are portable, i.e. the decay and mode parameters obtained in one environment can be applied to a different environment.

Exponential model

The parameters of the skew-Laplace model in [31] depend on whether the target is on a link’s LOS or not — a binary classification dependent on a pre-defined distance threshold. Instead of using a binary classification, the work in [29] quantifies the distance of a target to a link and developed a model that relates the change in RSS attenuation to this distance. In [29], the attenuation of a link’s RSS is modeled as an exponentially decaying function of the target’s “closeness” to the link. Using experimental data gathered from an uncluttered outdoor environment, the mean RSS attenuation for a

link i is modeled as

$$g_i(\mathbf{x}) = \phi \exp \left(-\frac{\lambda_i(\mathbf{x})}{\sigma_\lambda} \right) , \quad (2.8)$$

where $\mathbf{x}$ is the target's location, ϕ is the attenuation when the target is in the direct LOS of the link, $\lambda_i(\mathbf{x})$ is a measure of the target's "closeness" to the link i , and σ_λ is a decay rate parameter. The value of $\lambda_i(\mathbf{x})$ is essentially the excess path length of the ellipse with foci at the nodes that make up link i , and is computed as

$$\lambda_i(\mathbf{x}) = d_i^t(\mathbf{x}) + d_i^r(\mathbf{x}) - d_i \quad (2.9)$$

where $d_i^t(\mathbf{x})$ and $d_i^r(\mathbf{x})$ are the distances between the target at location $\mathbf{x}$ and the transmitter and receiver of link i , respectively, and d_i is the length of link i . Thus, as the target moves farther away from the link, $\lambda_i(\mathbf{x})$ becomes larger and the predicted attenuation using Eq. (2.8) becomes smaller. This approach avoids the use of a distance threshold to classify the target's location with respect to a link as LOS or not, as is done in [31].

To complete the statistical model, [29] models the observed noisy attenuation values as

$$z_i = g_i(\mathbf{x}) + w , \quad (2.10)$$

where w is zero-mean Gaussian noise with variance σ_w^2 . Through simulations, the exponential model coupled with a particle filter was evaluated and compared with an RTI-based method that uses a Kalman filter for tracking. The simulation results demonstrated the increased tracking accuracy of the proposed method in [29] as compared to RTI and Kalman filter-based approaches.

Magnitude model

The work in [32] further improved the exponential model of [29] by noting that it does not capture the amplification of RSS which may occur due to multipath effects in an indoor environment. Since the model of [29] was developed using data from an uncluttered outdoor environment, the links of the network predominantly experience RSS attenuation when a target is near the link's LOS. In order to capture both the RSS amplification and attenuation that may be experienced in cluttered indoor environments, the magnitude of the observed attenuation values, $|z_i|$, is used to derive a new model in [32]. The form of the magnitude model of [32] is the same as that of the exponential model shown in (2.8), except the values for the parameters ϕ and σ_λ are significantly different. The fit of the magnitude and exponential models to indoor RSS measurements were compared in [32], and it was observed that the magnitude model can more easily determine the target's closeness to the link. At small values of the excess path length $\lambda_i(\mathbf{x})$, the magnitude model has a higher slope as compared to the exponential model. This can be seen in the inset figures in Fig. 2.3, where the exponential and magnitude models are overlaid on the box-and-whisker plots of the attenuation and its magnitude, respectively, as the excess path length $\lambda_i(\mathbf{x})$ is increased [32].

Experiments in [32] demonstrate the higher tracking accuracy obtained by the magnitude model as compared to the skew-Laplace model in [31]. The work in [32] also reported that the skew-Laplace model of [31] resulted in multiple lost tracks, and that, contrary to the claim of [31], parameters obtained in one environment cannot be effectively applied in another.

(a)

(b)

Figure 2.3: Comparison of fits for the (a) exponential and (b) magnitude models [32].

Exponential-Rayleigh

The exponential and magnitude models previously discussed essentially describe the attenuation experienced by a link as an exponentially decaying function dependent on the target's relative position to the link and corrupted by Gaussian noise. In both models, the effect of the target on the measured attenuation is observed to be significant for the region where $\lambda_i(\mathbf{x})$ is very small, i.e. the target is in the link's LOS. The work in [34] extends the exponential and magnitude models and proposed an exponential-Rayleigh (ER)

model that aims to capture the small scale fading effects of a target that is not in the link’s LOS. The ER model uses a Rayleigh function to characterize the small-scale increase in the RSS attributed to the waves reflected from the target when the target is within a certain distance from the link, which [34] calls the “enhancement range”.

The attenuation of a link’s RSS as described by the ER model of [34] is

$$g_i(\mathbf{x}) = \phi \exp\left(-\frac{\lambda_i(\mathbf{x})}{\sigma_\lambda}\right) - \beta_b \lambda_i(\mathbf{x}) \exp\left(-\frac{\lambda^2(\mathbf{x})}{\sigma_b}\right) \quad (2.11)$$

where β_b and σ_b are the parameters for the Rayleigh component of the model, and ϕ and σ_λ are the parameters of the exponential model shown in (2.8). The predicted attenuation values of the exponential, magnitude, and ER models were compared with the actual measurements obtained by [34] in their experiments, and the ER model was reported to have the lowest modeling error compared to the others, with its model error approximating a zero-mean Gaussian distribution. Tracking experiments performed by [34] in multiple environments also demonstrate the higher tracking accuracy obtained with the ER model as opposed to the exponential and magnitude models.

Other models such as those in [35] and [36] have also been recently proposed. The work in [35] treats the target as a cylinder instead of a point mass, and uses diffraction theory to formulate the measurement model. In [36], an elaborate saddle surface model is used to describe the RSS variation caused by the target within the link’s spatial impact area.

In summary, the use of statistical models to describe the target-induced changes in a link’s RSS allow the direct estimation of a target’s location coordinates without going through an intermediate imaging step as in RTI

and BGA. Due to the nonlinear nature of the statistical models in [29]–[36], a Kalman filter or its extended and unscented variants will perform suboptimally. Instead, [29]–[36] use a particle filter for estimating the posterior distribution of the target’s location. The use of a particle filter framework also allows nonlinear dynamical models of the target motion to be utilized. However, particle filters are known to have high computational costs, typically requiring a large number of particles to achieve accurate results. The models proposed in [29]–[36] are highly nonlinear, and have to be evaluated for each particle. Thus, while statistical inversion methods for DFL are typically more accurate than RTI-based methods, they are also more computationally expensive.

2.3 Proposed DFL Algorithm

In this thesis, a new algorithm that uses geometric methods to solve the problem of single-target device-free localization and tracking is proposed. Similar to the intersection algorithm of [26], [27], the proposed method also uses the intersection points of target-affected links as probable target locations. The proposed method is different from the geometric methods in [26], [27] in the following ways:

- The proposed algorithm uses information on prior target location estimates to remove outlier points and links, making it more applicable for noisy, multipath-rich environments.
- Weights that depend on the link attenuation and distance of intersection points to the prior location estimate are used to improve the current location estimate.

- The proposed algorithm does not rely on a regular 2-D grid deployment of nodes, making it suitable for applications where such a deployment scheme may be impossible.

The proposed algorithm is named as the “Geometric Filter” (GF) algorithm. The details of the GF algorithm are presented in the next chapter.

2.4 Summary

This chapter presented a review of the literature on RSS-based DFL algorithms. These algorithms can be classified into location-based and link-based types. The location-based algorithms, also known as fingerprint-based algorithms, have the advantage of needing fewer nodes to solve the DFL problem. However, these algorithms require substantial effort in calibration of the reference RSS database. Link-based DFL algorithms avoid this tedious calibration step by using the relationship between a link’s RSS and the relative position of the target with respect to the link. Link-based algorithms can further be classified by the method with which location estimates are generated. Geometric methods, RTI-based methods, and statistical inversion methods fall under the class of link-based DFL algorithms. Each method presents its own advantages and disadvantages, and a tradeoff between computational cost and tracking accuracy is observed. A summary of the main features of the link-based DFL algorithms discussed in this section is shown in Table 2.1. Geometric methods have the lowest computational overhead, but are prone to noise and suffer from poor accuracy, especially in multipath-rich environments. On the other hand, statistical inversion methods that use a particle filter framework are highly accurate, but are also the

most computationally expensive due to the complex observation models and large number of particles required.

In the next chapter, a new algorithm based on the geometric method for DFL is introduced. The algorithm uses locational filtering to remove outlier points and links, and applies RSS-based and distance-based weights to refine the location estimate. The new algorithm aims to outperform the existing RTI-based and statistical inversion methods in terms of tracking accuracy and execution time.

Algorithm	Measurement Type	Tracking Filter	Advantages	Disadvantages
RTI	attenuation [18], [22]–[25], variance [19], [20], kernel distance [21]	Kalman filter [19], [21]	Inherent visualization of target location, Linear formulation	High memory requirements, quantization error due to discretization, requires additional step for coordinate estimate
Geometric	attenuation [28], mean absolute deviation [26], [27]	Kalman filter [28]	Simple and intuitive implementation, low computational requirements	Prone to errors due to multipath effects
Statistical inversion	attenuation [29]–[31], [34]–[36], magnitude of attenuation [32]	Particle filter	Can use nonlinear models of target motion, high accuracy	High computational requirements

Table 2.1: Comparison of link-based DFL algorithms.

Chapter 3

Geometric Filter Algorithm for DFL

In the previous chapter, we have presented a review of DFL algorithms. Geometric, RTI, and statistical inversion methods are link-based DFL algorithms that quantify the relationship between the change in RSS and the relative position of the target to the link's nodes to determine the target's location. Among the algorithms, the geometric methods offer the lowest computational cost but are prone to noise and poor tracking accuracy, while statistical inversion methods achieve high accuracy at the expense of complex measurement models and high computational requirements.

In this chapter, a new geometric filter (GF) algorithm for DFL is presented. The GF algorithm uses geometric objects such as line segments, points, and a circle to represent target-affected links, probable target locations, and a locational filter, respectively. This reduces the algorithm's storage and computational requirements, resulting in fast execution times. Different from other geometric methods, the GF algorithm uses the prior

location estimate to build a locational filter which is used to (a) remove outlier links, (b) remove improbable target locations, and (c) assign distance-dependent weights to probable target locations, ensuring robust tracking performance.

In the latter part of the chapter, we test the performance of the GF algorithm in uncluttered outdoor and cluttered indoor environments. The GF algorithm is compared with RTI, BGA, and particle filter-based methods for DFL in terms of tracking accuracy and execution time.

3.1 Problem statement

Consider a wireless network consisting of N nodes with known locations and distributed at the perimeter of the region to be monitored, such as that shown in Fig. 3.1. For the rest of this thesis, it is assumed that all wireless nodes are within radio range of each other, and are coplanar with the single human target to be tracked, i.e. only single target 2-D tracking is performed.

A link $\mathbf{S}_i$ is defined by the line segment connecting its two end nodes located at $\mathbf{s}_i^a$ and $\mathbf{s}_i^b$. The RSS measurement $R(i, t)$ of a link $\mathbf{S}_i$ is defined as the mean of the RSS measurements from $\mathbf{s}_i^a$ to $\mathbf{s}_i^b$ and vice versa. Given N wireless nodes, there are $M = \frac{N(N-1)}{2}$ bidirectional links in the network that go through the monitored region. A target inside the monitored region will affect a subset of these M links, causing changes in their RSS measurements. Let $\mathbf{R}(t) = [R(1, t) \ R(2, t) \ \dots \ R(M, t)]^T$ be the vector of RSS measurements for all links at time t , and $\bar{\mathbf{R}}_0 = [\bar{R}(1) \ \bar{R}(2) \ \dots \ \bar{R}(M)]^T$ be the vector of baseline RSS measurements for all links, where T is the transpose operation.

The RSS vector $\bar{\mathbf{R}}_0$ typically consists of the mean RSS measurement for each link taken while the monitored region is vacant and static, but

Figure 3.1: A DFL system with 16 nodes at the perimeter. The presence of the target affects the RSS of some links in the network.

may also be obtained using RSS measurements from the previous sampling period [33], a moving average of recent RSS measurements [23], [25], and other background subtraction techniques [46]. In this thesis, the vacant scene RSS measurements are used for $\bar{\mathbf{R}}_0$.

With knowledge of $\mathbf{R}(t)$, $\bar{\mathbf{R}}_0$, and the locations of the N wireless nodes, the goal is to estimate the target's true location $\mathbf{x}(t)$ at time t .

The RSS $R(i, t)$ measured at time t for link i can be modeled as :

$$R(i, t) = P(i) - L(i) - S(i, t) + F(i, t) - \nu(i, t) \quad (3.1)$$

where $P(i)$ is the transmitted power in dB, $L(i)$ is the large scale path loss, $S(i, t)$ is the shadowing loss due to attenuating objects, $F(i, t)$ is the multi-

path fading gain or fade level, and $\nu(i, t) \sim \mathcal{N}(0, \sigma^2(i))$ is the measurement noise [18]. DFL systems that use the target's shadowing effect typically assume that the quantities $P(i)$ and $L(i)$ are time-invariant, and treat the multipath fading loss $F(i, t)$ as noise. Hence, the change in RSS between the current time instant and the baseline RSS is assumed to be dominated by the target's shadowing effect on the link, i.e.

$$\Delta R(i, t) = R(i, t) - \bar{R}(i) \approx \Delta S(i, t) . \quad (3.2)$$

where $\Delta R(i, t)$ is the change in RSS and $\Delta S(i, t)$ is the target's shadowing effect on the link i . If a link is shadowed, then the target is likely to be located along the link line [18]. With multiple shadowed links, the intersection points of the link lines are likely locations of the target.

3.2 Proposed Geometric Filter Algorithm

In this section, the methodology of the proposed GF algorithm is presented. The detection of target-affected links, definition of the prior region, the method for solving the link intersections, computation of RSS-based and distance-based weights, and generation of target location estimate are discussed in the succeeding sections.

3.2.1 Detecting the target-affected links

The presence of the target inside the monitored region causes some of the M links to be obstructed, as illustrated in Fig. 3.1. To determine which links are affected by the target, the baseline RSS vector $\bar{\mathbf{R}}_0$ is subtracted

from the current RSS measurement vector $\mathbf{R}(t)$ to obtain $\Delta\mathbf{R}(t)$,

$$\Delta\mathbf{R}(t) = |\mathbf{R}(t) - \bar{\mathbf{R}}_0| , \quad (3.3)$$

where $\Delta\mathbf{R}(t) = [\Delta R(1, t) \ \Delta R(2, t) \ \dots \ \Delta R(M, t)]^T$ is the vector of RSS change measurements.

In rich multipath environments, a link may experience either attenuation or amplification due to the presence of the target in its vicinity [32]. Thus, a link i is considered to be affected by the target if its corresponding change in RSS measurement $\Delta R(i, t)$ is above the detection threshold γ . The set $\mathbf{L}_s(t)$ of target-affected links at time t is then defined as the set containing all segments $\mathbf{S}_i$ such that $\Delta R(i, t) > \gamma$, i.e.

$$\mathbf{L}_s(t) = \{\mathbf{S}_i \mid \Delta R(i, t) > \gamma\} , \quad (3.4)$$

where γ is a user-defined parameter indicating the detection threshold.

3.2.2 Defining the prior region

Since the target to be tracked is uncooperative, its true speed and direction is unknown. In general, the only information about the target that is available to a DFL system is its previous location estimate and the assumption that the target's maximum speed is below a threshold $v_{max} \neq 0$. Given these information, the prior region $\mathbf{P}(t)$ is defined as a circle centered on the previous target location estimate,

$$\mathbf{P}(t) = \{\mathbf{x} \mid \|\mathbf{x} - \hat{\mathbf{x}}(t-1)\| < r\}, \quad (3.5)$$

where $\hat{\mathbf{x}}(t-1)$ is the previous location estimate, $\mathbf{x}$ is the coordinate of a point inside the monitored region, $\|\cdot\|$ is the Euclidean norm, and r is the radius of the circular prior region. The radius r of the prior region is dependent on the maximum distance that the target can travel in the time between sampling instants Δt , i.e. $r > v_{max} \times \Delta t$. The prior region essentially defines the section of the monitored region where the target is most likely to be located next, given information about the previous location estimate. It must be noted that for $t = 1$, since $\hat{\mathbf{x}}_0$ is unknown, the radius r is set to a large enough value such that no points are filtered, and all intersection points obtained from the target-affected links are used in location estimation.

In the proposed algorithm, the prior region $\mathbf{P}(t)$ is used in two ways: (1) to remove target-affected links and intersection points that are far from the target, and (2) to assign distance-based weights to intersection points before a location estimate is made. The details of these procedures are presented in the succeeding sections.

3.2.3 Link filter (LF)

Due to noise and multipath effects, not all of the detected target-affected links go through the vicinity of the target. Including these outlier links in generating a location estimate may result in large tracking errors, as they may intersect with other target-affected links at points far away from the target. To eliminate the effect of these outlier links on the target location estimate, only the links that are in the vicinity of the prior target location estimate are considered. Target-affected links that do not intersect the prior region are removed. Let the target-affected link i be represented by the line segment $\mathbf{S}_i$ from the set $\mathbf{L}_s(t)$, with endpoints at location $\mathbf{s}_i^a$ and $\mathbf{s}_i^b$. To

determine if $\mathbf{S}_i$ intersects the prior region $\mathbf{P}(t)$, the algorithm first checks if the distance between $\hat{\mathbf{x}}(t-1)$ and any of the link's endpoints $\mathbf{s}_i^a$ and $\mathbf{s}_i^b$ is less than the radius r . If this is the case, then the link intersects $\mathbf{P}(t)$ and no further computations are necessary. If not, it then proceeds to check if the shortest distance between a point on the link and the center of the prior region $\hat{\mathbf{x}}(t-1)$ is not greater than the radius r . Let the vectors $\mathbf{v}_\mathbf{A} = \mathbf{s}_i^b - \mathbf{s}_i^a$ and $\mathbf{v}_\mathbf{B} = \hat{\mathbf{x}}(t-1) - \mathbf{s}_i^a$ be the vectors for the link $\mathbf{S}_i$ and for the segment connecting one endpoint of $\mathbf{S}_i$ to the center of $\mathbf{P}(t)$, respectively. To determine the point $\mathbf{c}$ on $\mathbf{v}_\mathbf{A}$ that is closest to $\hat{\mathbf{x}}(t-1)$, $\mathbf{v}_\mathbf{B}$ is projected onto $\mathbf{v}_\mathbf{A}$ to obtain l ,

$$l = \mathbf{v}_\mathbf{B} \bullet \frac{\mathbf{v}_\mathbf{A}}{\|\mathbf{v}_\mathbf{A}\|} \quad (3.6)$$

where $(\bullet)$ is the dot product. The point $\mathbf{c}$ is then obtained by

$$\mathbf{c} = \mathbf{s}_i^a + l \cdot \frac{\mathbf{v}_\mathbf{A}}{\|\mathbf{v}_\mathbf{A}\|}. \quad (3.7)$$

The distance from $\mathbf{c}$ to $\hat{\mathbf{x}}(t-1)$ determines whether the link $\mathbf{S}_i$ intersects the prior region $\mathbf{P}(t)$. If $\|\mathbf{c} - \hat{\mathbf{x}}(t-1)\| \leq r$, the link $\mathbf{S}_i$ is added to a new set $\mathbf{L}(t)$ of links that intersect the prior region $\mathbf{P}(t)$, and is considered for generating the target location estimate. The procedure for the link filter (LF) is summarized in Proc. 3.1.

Essentially, applying the detection threshold γ and the link filter ensures that the set $\mathbf{L}(t)$ of links to be used for localization have RSS values that have been sufficiently changed by the presence of the target, and intersect a section of the monitored region where the target is most likely to be located next.

Procedure 3.1 Link filter

Input: The previous location estimate $\hat{\mathbf{x}}(t-1)$, the radius r of the circular prior region, and the set of target-affected links $\mathbf{L}_s(t)$.

Output: The set $\mathbf{L}(t)$ of target-affected links that intersect with the prior region.

```
1: Initialize  $\mathbf{L}(t)$  to the empty set.
2: for all  $\mathbf{S}_i = [\mathbf{s}_i^a \ \mathbf{s}_i^b]^T \in \mathbf{L}_s(t)$  do
3:   if  $\|\mathbf{s}_i^a - \hat{\mathbf{x}}(t-1)\| \leq r$  or  $\|\mathbf{s}_i^b - \hat{\mathbf{x}}(t-1)\| \leq r$  then
4:      $\mathbf{L}(t) \cup \{\mathbf{S}_i\}$   $\triangleright$  add  $\mathbf{S}_i$  to the set  $\mathbf{L}(t)$ 
5:   else
6:      $\mathbf{v}_A = \mathbf{s}_i^b - \mathbf{s}_i^a$ 
7:      $\mathbf{v}_B = \hat{\mathbf{x}}(t-1) - \mathbf{s}_i^a$ 
8:     Solve for  $l$  using (3.6).
9:     Solve for  $c$  using (3.7).
10:    if  $\|\mathbf{c} - \hat{\mathbf{x}}(t-1)\| \leq r$  and  $\|\mathbf{c}\| \leq \|\mathbf{v}_A\|$  then
11:       $\mathbf{L}(t) \cup \{\mathbf{S}_i\}$   $\triangleright$  add  $\mathbf{S}_i$  to the set  $\mathbf{L}(t)$ 
12:    end if
13:  end if
14: end for
```

3.2.4 Solving for the link intersections and obtaining RSS-based weights

Given the set $\mathbf{L}(t)$ of target-affected links obtained in Proc. 3.1, the intersection points among all members of $\mathbf{L}(t)$ is computed by testing each pairwise combination of segments. Let segments $\mathbf{S}_j$ and $\mathbf{S}_k$, from node locations $\mathbf{s}_j^a$ to $\mathbf{s}_j^b$, and $\mathbf{s}_k^a$ to $\mathbf{s}_k^b$, respectively, be two segments from $\mathbf{L}(t)$ which are to be tested for intersections. The parametric form for these segments is

$$\mathbf{S}_j = \begin{bmatrix} x_j^a \\ y_j^a \end{bmatrix} + u \begin{bmatrix} x_j^b - x_j^a \\ y_j^b - y_j^a \end{bmatrix}, u \in [0, 1] \quad (3.8)$$

$$\mathbf{S}_k = \begin{bmatrix} x_k^a \\ y_k^a \end{bmatrix} + v \begin{bmatrix} x_k^b - x_k^a \\ y_k^b - y_k^a \end{bmatrix}, v \in [0, 1] \quad (3.9)$$

where (x_j^a, y_j^a) , (x_j^b, y_j^b) , (x_k^a, y_k^a) , and (x_k^b, y_k^b) are the coordinates of points $\mathbf{s}_j^a$, $\mathbf{s}_j^b$, $\mathbf{s}_k^a$, and $\mathbf{s}_k^b$, respectively. The intersection point, if it exists, must lie on both segments $\mathbf{S}_j$ and $\mathbf{S}_k$. Thus, equations (3.8) and (3.9) must be equal at the intersection point. Equating (3.8) and (3.9) yields the linear equations

$$\begin{bmatrix} x_k^b - x_k^a & -(x_j^b - x_j^a) \\ y_k^b - y_k^a & -(y_j^b - y_j^a) \end{bmatrix} \begin{bmatrix} v \\ u \end{bmatrix} = \begin{bmatrix} x_j^a - x_k^a \\ y_j^a - y_k^a \end{bmatrix}. \quad (3.10)$$

The parameters u and v are solved using Cramer's rule. Let the determinant

$$D = (x_k^b - x_k^a)(y_j^a - y_j^b) - (x_j^b - x_j^a)(y_k^a - y_k^b). \quad (3.11)$$

If $D = 0$, the two segments are parallel and the procedure moves on to the next pair combination of segments from $\mathbf{L}_s(t)$. Otherwise, the parameters u and v are computed. Applying Cramer's rule results in

$$u = \frac{(x_k^b - x_k^a)(y_j^a - y_k^a) - (x_j^a - x_k^a)(y_k^b - y_k^a)}{D} \quad (3.12)$$

$$v = \frac{(x_j^a - x_k^a)(y_j^a - y_j^b) - (x_j^b - x_j^a)(y_k^a - y_j^a)}{D}. \quad (3.13)$$

If both conditions $0 < u < 1$ and $0 < v < 1$ are satisfied, the intersection point $\mathbf{q}$ exists and is given by

$$\mathbf{q} = \begin{bmatrix} x_k^a \\ y_k^a \end{bmatrix} + v \begin{bmatrix} x_k^b - x_k^a \\ y_k^b - y_k^a \end{bmatrix}. \quad (3.14)$$

In addition to the intersection point $\mathbf{q}$, the sum of the magnitude of RSS change experienced by intersecting links $\mathbf{S}_j$ and $\mathbf{S}_k$ is stored for later use as a weight when a location estimate is generated,

$$w_{\Delta R} = |\Delta R(j, t)| + |\Delta R(k, t)| \quad (3.15)$$

where $w_{\Delta R}$ is the non-normalized weight for intersection point $\mathbf{q}$, and $\Delta R(j, t)$, $\Delta R(k, t)$ are the RSS change measurements for links $\mathbf{S}_j$ and $\mathbf{S}_k$, respectively. Essentially, higher weights are assigned to intersection points of links that are more heavily affected by the target, as the target is more likely to be in the path of these links. This procedure is then repeated for the next pair combination of segments from $\mathbf{L}(t)$. All intersection points found are stored in the set $\mathbf{Q}_s(t)$. The RSS-based non-normalized weights corresponding to each intersection point are stored in the set $\mathbf{W}_{\Delta R}^s$. The elements of $\mathbf{Q}_s(t)$ and $\mathbf{W}_{\Delta R}^s$ are indexed with $i = 1, 2, \dots, |\mathbf{Q}_s(t)|$, such that the weight $w_{\Delta R}^i$ corresponds to the intersection point $\mathbf{q}_i(t)$ and $|\mathbf{Q}_s(t)|$ is the total number of intersection points found. A summary of the procedure is shown in Proc. 3.2.

3.2.5 Point filter (PF)

Given the set of intersection points $\mathbf{Q}_s(t) = \{\mathbf{q}_1(t), \mathbf{q}_2(t), \dots, \mathbf{q}_{|\mathbf{Q}_s(t)|}(t)\}$ obtained in the previous step, not all members of $\mathbf{Q}_s(t)$ may lie inside the circular prior region. Since each intersection point $\mathbf{q}_i(t)$ represents a possible location of the target at time t , points that lie outside the circular prior region may introduce large errors into the location estimate. These outlier points are filtered out by determining their distance to the previous location

Procedure 3.2 Solve for segment intersections and non-normalized RSS-based weights

Input: The set of target-affected links $\mathbf{L}(t)$ from Proc. 3.1.

Output: The set of intersection points $\mathbf{Q}_s(t)$ of all segments in $\mathbf{L}(t)$, and the set of RSS-based non-normalized weights $\mathbf{W}_{\Delta R}^s$ corresponding to each member of $\mathbf{Q}_s(t)$.

```

1: Initialize  $\mathbf{Q}_s(t)$  and  $\mathbf{W}_{\Delta R}^s$  to the empty set.
2: for all segments  $\mathbf{S}_i$  in  $\mathbf{L}(t)$  do
3:   for all segments  $\mathbf{S}_j$  in  $\mathbf{L}(t)$ ,  $i \neq j$  do
4:     Compute the determinant  $D$  using (3.11).
5:     if  $D \neq 0$  then
6:       Solve for parameters  $u$  and  $v$  using
         (3.12) and (3.13).
7:       if  $0 < u < 1$  and  $0 < v < 1$  then
8:         Solve for point  $\mathbf{q}$  using (3.14).
9:         Solve for the weight  $w_{\Delta R}$  using (3.15).
10:         $\mathbf{Q}_s(t) \cup \{\mathbf{q}\}$ 
11:         $\mathbf{W}_{\Delta R}^s \cup \{w_{\Delta R}\}$ 
12:      end if
13:    end if
14:  end for
15: end for

```

estimate $\hat{\mathbf{x}}(t - 1)$ and only considering the points that have distances less than or equal to the radius r . Thus, the set $\mathbf{Q}(t)$ is defined as

$$\mathbf{Q}(t) = \{\mathbf{q}_i(t) \mid \|\mathbf{q}_i(t) - \hat{\mathbf{x}}(t - 1)\| \leq r, \mathbf{q}_i(t) \in \mathbf{Q}_s(t)\}. \quad (3.16)$$

Due to the removal of the outlier points, the set $\mathbf{W}_{\Delta R}^s$ of RSS-based weights must also be updated accordingly. All weights that correspond to intersection points that have been removed by the point filter (PF) must also be

removed. The updated set of RSS-based weights are defined as

$$\mathbf{W}_{\Delta R} = \{w_{\Delta R}^i \mid \mathbf{q}_i(t) \in \mathbf{Q}(t)\}. \quad (3.17)$$

3.2.6 Distance-based weights

In addition to the previously obtained RSS-based weights, weights that are dependent on the intersection point's distance to the previous location estimate are also used. The distance-based weight w_d^i for the point $\mathbf{q}_i(t) \in \mathbf{Q}(t)$ is computed as

$$w_d^i = \left(1 - \frac{\|\mathbf{q}_i(t) - \hat{\mathbf{x}}(t-1)\|}{r}\right) \quad (3.18)$$

In essence, points that are closer to the previous location estimate are assigned higher weights as compared to farther points. The set of distance-based weights $\mathbf{W}_d$ for all intersection points $\mathbf{q}_i(t) \in \mathbf{Q}(t)$ is defined as

$$\mathbf{W}_d = \{w_d^i \mid \mathbf{q}_i(t) \in \mathbf{Q}(t)\}. \quad (3.19)$$

It must be noted that for $t = 1$, since $\hat{\mathbf{x}}(0)$ is unknown, all values of w_d^i are set to 1. Thus, the first location estimate of the GF algorithm is generated using only the RSS-based weights $\mathbf{W}_{\Delta R}$ and without any prior information on the target's location.

3.2.7 Generating a location estimate

Using the set of intersection points $\mathbf{Q}(t)$ obtained after the point filtering procedure, the set of RSS-based weights $\mathbf{W}_{\Delta R}$, and the set of distance-based weights $\mathbf{W}_d$, the location estimate $\hat{\mathbf{x}}(t)$ can now be generated. Essentially, the location estimate $\hat{\mathbf{x}}(t)$ is the weighted mean of all intersection points in

$\mathbf{Q}(t)$,

$$\hat{\mathbf{x}}(t) = \sum_{i=1}^{|\mathbf{Q}(t)|} w^i \cdot \mathbf{q}_i(t) \quad (3.20)$$

where

$$w^i = \frac{w_{\Delta R}^i \cdot w_d^i}{\sum (w_{\Delta R}^i \cdot w_d^i)} \quad (3.21)$$

are the normalized weights, and $|\mathbf{Q}(t)|$ denotes the cardinality of the set $\mathbf{Q}(t)$. The GF algorithm is summarized in Proc. 3.3.

An illustration of the GF algorithm is shown in Fig. 3.2, using data obtained from an outdoor experiment. Fig. 3.2a shows the links affected by the target's presence. Links that do not intersect with the prior region are treated as outlier links and removed, as shown in Fig. 3.2b. Fig. 3.2c shows the intersection points of the remaining links which are used as probable target locations. Points that lie outside the prior region are removed and the weighted mean of the remaining points is computed for the target location estimate, as shown in Fig. 3.2d and Fig. 3.2e.

3.3 Experimental Setup

Experiments were conducted in an uncluttered outdoor environment and a cluttered indoor environment to evaluate the performance of the proposed GF algorithm and demonstrate its applicability in different scenarios. In both scenes, the wireless nodes were placed at the perimeter of the square monitored region. Placing the nodes as evenly as possible at the perimeter ensures that the radio frequency (RF) links cover more sections of the monitored region. The wireless nodes are 2.4 GHz IEEE 802.15.4 transceivers. Each node is programmed with a unique identification (ID) number and maintains a RSS measurement array that contains the latest RSS measure-

Procedure 3.3 Geometric filter algorithm for DFL

- 1: Obtain the baseline RSS vector $\mathbf{R}_0$.
 - 2: **for** $t \geq 1$ **do**
 - 3: Solve for $\Delta\mathbf{R}(t)$ using (3.3).
 - 4: Determine the set of target-affected links $\mathbf{L}_s(t)$ using (3.4).
 - 5: Build the prior region $\mathbf{P}(t)$ using (3.5).
 - 6: Apply the link filter using Proc. 3.1 to obtain $\mathbf{L}(t)$.
 - 7: Solve for the intersections of all segments in $\mathbf{L}(t)$ and their corresponding RSS-based weights using Proc. 3.2.
 - 8: Apply the point filter using (3.16).
 - 9: Update the set of RSS-based weights using (3.17).
 - 10: Generate the set of distance-based weights using (3.18).
 - 11: Generate the location estimate $\hat{\mathbf{x}}(t)$ using (3.20) and (3.21).
 - 12: **end for**
-

Figure 3.2: An illustration of the proposed algorithm. (a) Target-affected links and outlier links. (b) The outlier links are removed. (c) The intersection points of the remaining target-affected links. (d) Probable target locations outside the prior region are removed, and (e) a location estimate is generated using the weighted mean of the remaining points.

ment between itself and the other nodes in the network. The nodes operate sequentially by order of their node IDs to prevent transmission collisions. The RSS scanning operation is initiated by one of the nodes by broadcasting its node ID and its RSS measurement array. Upon reception of the broadcast packet, the other nodes update their RSS measurement arrays with the RSS between the transmitter and itself, and check whether it is their turn to broadcast. A central node listens to all broadcasts from the perimeter nodes and logs the RSS information to a mobile computer.

For both experiments, an empty scene calibration period of T_{cal} s is performed to obtain the baseline RSS of each link in the network. In both scenes, a single human target walks on a pre-defined track inside the monitored region at a speed of $v \approx 0.5\text{m/s}$. Markers were placed on the true locations along the track, and a metronome is used to assist the target in maintaining constant speed. The target's location is estimated once per second, and each experiment is repeated for N_{trials} trials. The tracking error is defined as the distance between the estimated target location and the known true target location, i.e. $e = ||\mathbf{x} - \hat{\mathbf{x}}||$, where $\mathbf{x}$ is the known target location and $\hat{\mathbf{x}}$ is the target location estimate obtained using the GF algorithm.

The performance of the GF algorithm is compared to BGA [45], RTI [18], and a statistical inversion-based method that uses a sequential importance resampling (SIR) particle filter with the observation model of [32]. The algorithms are compared in terms of the tracking root-mean-square error (RMSE) and execution time.

Figure 3.3: Scene 1, uncluttered outdoor experiment setup.

3.4 Scene 1, Uncluttered outdoor environment

For the uncluttered outdoor environment, 16 nodes are deployed at the perimeter of an $8\text{ m} \times 8\text{ m}$ area, spaced 2 m apart. A calibration period of $T_{cal} = 90\text{ s}$ was used to obtain the baseline RSS. The default parameters of the proposed algorithm are detection threshold $\gamma = 3\text{ dB}$, and radius of prior region $r = 2\text{ m}$. The number of voxels for the RTI and BGA methods is $N_{vox} = 1600$, while $N_p = 50$ particles are used for the SIR. The number of trials for the outdoor experiment is $N_{trials} = 90$. Fig. 3.3 and Fig. 3.4 show the experiment setup and photo of the deployment area for Scene 1, respectively.

Figure 3.4: Scene 1 deployment area.

Table 3.1: Scene 1, Tracking error statistics

Algorithm	Median [m]	Mean [m]	RMSE [m]
GF	0.494	0.540	0.638
BGA	0.517	0.714	1.13
RTI	0.573	1.06	1.82
SIR	0.51	0.633	0.85

3.4.1 Results

The tracking results for Scene 1 using the proposed GF algorithm are shown in Fig. 3.5, and are obtained by taking the mean location estimate across all experiment trials. Fig. 3.6 shows the mean tracking error at each time instant. From Figs. 3.5 and 3.6, it is seen that the proposed approach can track the target's motion around the square path very well.

The empirical cumulative distribution function (ECDF) and statistics of all tracking errors obtained by each algorithm are shown in Fig. 3.7 and

Figure 3.5: The mean tracking results for Scene 1, overlaid on the true target track.

Figure 3.6: Mean tracking errors vs time, obtained using the GF algorithm for Scene 1.

Figure 3.7: ECDF of tracking errors for each algorithm.

Figure 3.8: ECDF of tracking errors for GF with the locational filters disabled.

Table 3.2: Tracking error statistics using different weighting schemes for GF

Scheme	Median [m]	Mean [m]	RMSE [m]
GF-RW	0.596	0.659	0.773
GF-DW	0.511	0.559	0.657
GF-EW	0.622	0.699	0.832

Table 3.1, respectively. While the median errors for the algorithms are all less than 0.6 m, it is seen from Fig. 3.7 that the GF is more robust to large errors as compared to BGA, RTI, and SIR. For GF, 95% of the tracking errors are below 1.1 m, while BGA has its 95th percentile at 1.9 m, RTI is at 5 m, and SIR is at 1.7 m. From Table 3.1, the tracking RMS error obtained using GF is 43% lower compared to BGA, 65% lower compared to RTI, and 25% lower compared to SIR. These results show that the proposed GF algorithm achieves significantly greater accuracy compared to the BGA, RTI, and SIR algorithms.

The contribution of the locational filters to the performance of the GF algorithm is also evaluated. The “GF-NoFilter” scheme does not implement the point and link filters, i.e. all target-affected links and their intersections are used in generating a location estimate. The ECDF of the results obtained with this scheme is shown in Fig. 3.8. It is observed that the use of the PF and LF scheme has a great effect on the accuracy of the proposed GF algorithm. The RMSE of the GF-NoFilter scheme is 1.45 m, which is 127% greater compared to that of the GF algorithm. Since some intersection points are located far from the target’s true location, filtering these points before generating a location estimate results in a significant increase in accuracy.

To evaluate the contribution of the RSS-based weights $w_{\Delta R}$ and distance-

Figure 3.9: ECDF of tracking errors for GF variants with different weight schemes.

based weights w_d to the performance of the proposed algorithm, its performance using different weighting schemes is evaluated. The GF-EW scheme uses an unweighted mean, the GF-RW scheme uses only the RSS-based weights computed using (3.15) and (3.17), and the GF-DW scheme uses only the distance-based weights computed using (3.18). Since the weights $w_{\Delta R}$ and w_d are non-normalized, the GF-RW and GF-DW schemes normalize them first before generating a location estimate. A comparison of the ECDF of tracking errors obtained using these schemes is shown in Fig. 3.9. From Fig. 3.9, the GF-DW scheme performs nearly as well as the GF algorithm, while the GF-RW scheme is only slightly better than the GF-EW scheme which uses unweighted means. Table 3.2 shows the tracking error statistics for the different weighting schemes. Comparing the RMSE of the different schemes with the RMSE of the GF algorithm, the RMSEs of the GF-RW,

GF-DW, and GF-EW schemes are greater by 21%, 2.9%, and 30.3%, respectively. These results further show the importance of using prior information in generating a location estimate. By weighting possible target locations inversely proportional to their distance to the previous estimated target location, higher weights are assigned to points where the target is more likely located. This also confirms that combining the RSS-based weights $w_{\Delta R}$ and distance-based weights w_d yields the best performance for the GF algorithm.

3.4.2 Performance under different values of γ and r

The GF algorithm requires only two parameters to tune: the detection threshold γ and the prior region radius r . The performance of the proposed algorithm is evaluated under different values of these parameters, with the results shown in Fig. 3.10 and Fig. 3.11. It is seen from Fig. 3.10 that selecting a very high value for γ results in large tracking errors. This is because a large γ will yield fewer detected target-affected links, resulting in fewer intersection points for estimating the target's true location. On the other hand, a low γ makes the GF algorithm more susceptible to noise, especially in cluttered environments. From Fig. 3.10, it is seen that a value of γ between 1 and 3 dB yields good performance.

The radius r of the prior region has an effect on the link and point filters. Setting the value of r too high or too low results in an increase in the tracking error. Large r will result in the point filter's failure to remove outlier points. On the other hand, a small r will cause the point filter to remove points that are likely locations of the target but are outside the constrictive prior region. From Fig. 3.11, it is seen that the GF scheme is robust to using larger values of r . Thus, it is better to overestimate the target's maximum

speed v_{max} for determining r in practice.

Figure 3.10: Error bars of tracking errors as the detection threshold γ is varied. The central marks indicate the median of tracking errors. The lower and upper bars indicate the 25th and 75th percentiles, respectively.

Figure 3.11: Error bars of tracking errors as the radius of prior region r is varied. The central marks indicate the median of tracking errors. The lower and upper bars indicate the 25th and 75th percentiles, respectively.

Figure 3.12: Scene 2, cluttered indoor experiment setup.

3.5 Scene 2, Cluttered indoor environment

To demonstrate the applicability of the GF algorithm in rich multipath scenarios, an experiment was also conducted in a highly cluttered indoor environment. For the cluttered indoor environment, 25 nodes are placed at the perimeter of an $8.5 \text{ m} \times 8.5 \text{ m}$ area. In this scene, the calibration period is $T_{cal} = 90 \text{ s}$ and the number of trials is $N_{trials} = 50$. The parameters of the proposed algorithm are detection threshold $\gamma = 3 \text{ dB}$, and radius of prior region $r = 3 \text{ m}$. The number of voxels for the RTI and BGA methods is $N_{vox} = 4624$. For the SIR, the number of particles is kept at $N_p = 50$.

(a)

(b)

Figure 3.13: (a) Overall view of the cluttered indoor environment for Scene 2. (b) The target walking along the pre-defined path in Scene 2.

The number of trials for Scene 2 is $N_{trials} = 50$. Fig. 3.12 and Fig. 3.13 show the experiment setup and photos of the deployment area for Scene 2, respectively.

Links with RSS variances greater than 1 during the vacant scene are excluded since these adversely affect the tracking results of all the algorithms[32]. All other settings are similar to those in the outdoor experiment.

Fig. 3.14 shows the mean tracking results of 50 trials for Scene 2. From

Figure 3.14: Mean tracking results for Scene 2, overlaid on the true target track.

Fig. 3.14, it is observed that with a sufficient number of nodes, the GF algorithm can still track the target with sub-meter accuracy, even in the presence of multiple obstructions in the deployment area. Table 3.3 summarizes the tracking error statistics for all algorithms in Scene 2. The GF algorithm has the lowest tracking RMSE among all the algorithms evaluated in Scene 2.

In the indoor environment, multipath fading effects contribute to higher variance in RSS measurements. This may result in more target-affected links being detected, some of which will intersect at points very far from the target's true location. In addition, the higher RSS variance also affects

Table 3.3: Scene 2, Tracking error statistics

Algorithm	Median [m]	Mean [m]	RMSE [m]
GF	0.782	0.826	0.949
BGA	0.747	0.96	1.24
RTI	2.11	2.03	2.18
SIR	0.567	0.727	0.964

the RSS-based weights $w_{\Delta R}$. Using only the RSS-based weights in the GF-RW scheme as described in Section 3.4.1, the indoor tracking RMS error is 1.18 m. However, the GF algorithm’s use of the point filter, and the combination of the weights $w_{\Delta R}^i$ and w_d^i in Eq. (3.21) mitigate the effect of higher RSS variance, reducing the tracking RMS error to 0.9487 m.

3.6 Average execution time

In the previous sections, it is shown that the GF algorithm can outperform other link-based DFL methods in terms of tracking accuracy. In this section, we compare the average execution times of the previously tested algorithms and relate it qualitatively to each algorithm’s computational requirements. All algorithms were coded in C++, compiled without optimizations, and run on a 3.40 GHz desktop computer.

For the SIR, a sufficient number of particles (typically $N_p > 50$) must be used to achieve acceptable performance. At each iteration of the SIR, N_p operations are used for the prediction stage and $M \times N_p$ operations are needed to update the particle weights. The weight update operation is computationally expensive as it involves the evaluation of a complex observation

model. For the BGA and RTI schemes, an $M \times N_{vox}$ matrix is computed offline before any location estimates are made. RTI requires the multiplication of a $1 \times M$ link RSS vector with an $M \times N_{vox}$ weighting matrix to generate an image of the RF propagation field in the deployment area. BGA generates a location estimate through addition and multiplication operations on a series of $N_{vox} \times 1$ vectors. However, the $N_{vox} \times 1$ prior region in BGA needs to be constructed at each iteration, even for grids where the target is highly improbable to locate.

In the GF algorithm, the use of geometric objects (i.e. points, segments, circle) removes the need for storing and processing large amounts of data, resulting in fewer computations. For example, in BGA, a target-affected link is represented by an $N_{vox} \times 1$ shadowing effect map and its corresponding ΔR measurement. In contrast, the GF algorithm represents a target-affected link by a 5-member vector containing its endpoints' x - y coordinates and its corresponding ΔR measurement. The reduced number of operations and storage required by the GF algorithm results in significantly faster execution times as compared to the RTI, BGA, and statistical inversion methods. In addition, the use of the link filter ensures that outlier links are not included in the computation for intersection points, resulting in reduced algorithm execution time in rich multipath scenarios.

The average execution times of all algorithms for both indoor and outdoor environments are shown in Table 3.4. It can be seen that in both scenes, the GF algorithm executes much faster compared to the other DFL algorithms. These results demonstrate that the GF algorithm is more suitable for storage and computational resource constrained applications.

Table 3.4: Average execution times

Scenes	GF	BGA	RTI	SIR
Scene 1, $M = 120$	$2\ \mu s$	$22\ \mu s$	$46\ \mu s$	1.1 ms
Scene 2, $M = 300$	$8\ \mu s$	$180\ \mu s$	$600\ \mu s$	2.52 ms

3.7 Summary

In this chapter, we have presented the methodology of the proposed GF algorithm. The GF algorithm’s methods for inferring the target’s location using information from the target-induced changes in RSS were discussed in detail. We have evaluated the performance of the proposed GF algorithm in experiments conducted in outdoor and indoor environments. The experimental results in both environments show significant improvements in tracking accuracy and average execution time over RTI-based and particle filter-based methods for DFL.

In the next chapter, the GF algorithm is extended to work with RSS measurements taken across different frequency channels. The use of channel diversity enables the GF algorithm to achieve more accurate tracking in multipath-rich environments.

Chapter 4

Multi-Channel Geometric Filter Algorithm

In the previous chapter, the methodology and experimental results of the GF algorithm for single target device-free localization and tracking have been presented. It was shown that the GF algorithm is able to accurately track the target and can outperform existing DFL algorithms in terms of accuracy and average execution time. In this chapter, an enhanced version of the GF algorithm is presented. The algorithm is modified to work with multi-channel RSS measurements, and to use link-specific thresholds in detecting target-affected links. The MCGF algorithm's tracking accuracy and execution time is evaluated and benchmarked with multi-channel RTI-based methods and single-channel GF. The performance of MCGF as its parameters are varied is also investigated.

4.1 Motivation

In uncluttered outdoor environments, the target's effect on a link's RSS is generally that of attenuation. The use of DFL indoors is more challenging due to the presence of obstructions and objects, resulting in a rich multipath environment. Due to multipath fading, the target's presence can cause either RSS attenuation or amplification. To quantify the steady-state narrowband fading experienced by a link, the concept of fade level was introduced in [31], and it was shown that a person's effect on the link's RSS depends on the link's fade level. Links experiencing destructive multipath interference are classified as links in *deep fade*, while links experiencing constructive multipath interference are classified as links in *anti-fade*. Links in anti-fade are more informative for DFL systems, since the target's effect on such links are more predictable and are constrained to a smaller section of the monitored region [24].

A link's fade level depends on the frequency channel it uses. Thus, it may be in deep fade on one channel but in anti-fade on another. To increase the likelihood that at least one of a link's channels is an anti-fade channel, the work in [22] used multi-channel communications among the nodes in a DFL system. By selecting only the top channels in anti-fade, the localization accuracy of an RTI-based DFL system was improved. The work in [22] was extended in [25], wherein a link's RSS measurements across different channels are combined using a fade level-weighted average. A fade level-based spatial model was also developed in [24] that considers the spatial impact area of each link as a function of its fade level. Recently, [47] used multiple power levels and channels for each link to increase the number of available RSS measurements. A recursive compressed sensing algorithm was

developed to estimate the target’s location from the multidimensional link information.

Inspired by [22], [24], [25], this work uses channel diversity to improve the GF algorithm’s tracking accuracy. In this chapter, the GF algorithm is modified to work with multi-channel RSS measurements to mitigate the multipath fading effects in cluttered environments. The GF algorithm’s target-affected link detection method is also improved by using link-specific thresholds based on the link’s multi-channel fade levels and RSS variance. This is different from the detection methods in [33], [45], in which a single fixed threshold value is used to detect the target-affected links, disregarding the differences in noise experienced by each link due to multipath interference. Finally, the point-weighting scheme of the GF algorithm’s location estimation method is further improved by the inclusion of weights based on the overall fade levels of the intersecting links. We refer to the improved version of the GF algorithm as “Multi-Channel Geometric Filter” (MCGF) algorithm.

4.2 Problem statement

Similar to the problem defined in Chapter 3.1, our objective is to determine the location of a single device-free target using only information about the target-induced changes in the RSS and the location of the network’s nodes.

Consider a network of N wireless nodes communicating on a set of channels $\mathbf{J}$, with known locations and distributed around the region to be monitored. Assuming that all nodes can communicate with any other node in a single hop, there are $M = \frac{N \times (N-1)}{2}$ bidirectional links in the network. The target’s presence inside the monitored region causes changes in the RSS

of a subset of these M links due to scattering, reflection, and absorption. These target-affected links tend to intersect with each other at points near the target's true location. Thus, an estimate of the target's location can be obtained by fusing information about the intersection points, the change in RSS of the target-affected links, the links' fade levels, and the previous location estimate.

The model for the RSS of a link presented in (3.1) is extended to include a dependency on the frequency channel on which the link operates. The RSS $R(i, j, t)$ measured at time t for link i on channel $j \in \mathbf{J}$ can be modeled as :

$$R(i, j, t) = P_t(j) - L(i) - S(i, j, t) + F(i, j, t) - \nu(i, j, t) \quad (4.1)$$

where $P_t(j)$ is the transmit power on channel j , $L(i)$ is the large scale path loss, $S(i, j, t)$ is the shadowing loss due to attenuating objects, $F(i, j, t)$ is the multipath fading gain or fade level, $\nu(i, j, t) \sim \mathcal{N}(0, \sigma^2(i, j))$ is the measurement noise, and $\mathbf{J}$ is the set of radio communication channels [22].

Similar to the GF algorithm, the average RSS of links during a calibration period when the target is not present inside the monitored area is used for the baseline RSS measurements. Different from the previous model in (3.1), the fade level is no longer discounted as noise. Let

$$\bar{R}(i, j) = P_t(j) - L(i) - \bar{S}(i, j) + \bar{F}(i, j) \quad (4.2)$$

be the mean RSS for link i on channel j during the calibration period, and $\bar{S}(i, j)$ and $\bar{F}(i, j)$ be the mean shadowing loss and fading gain, respectively. The change between the baseline RSS and the RSS at time t when the target

is known to be inside the monitored area is

$$\begin{aligned}\Delta R(i, j, t) &= R(i, j, t) - \bar{R}(i, j) \\ &= \Delta S(i, j, t) + \Delta F(i, j, t) - \Delta \nu(i, j, t)\end{aligned}\quad (4.3)$$

where $\Delta S(i, j, t)$ is the change in the shadowing loss and $\Delta F(i, j, t)$ is the change in fade level, both due to the target's presence inside the monitored area at time t . By measuring the change in RSS $\Delta R(i, j, t)$, a DFL system can infer whether or not the target is located along the link's path.

Let $\mathbf{R}_0 \in \mathbb{R}^{M \times |\mathbf{J}|}$ denote the baseline RSS measurement matrix, where $|\mathbf{J}|$ is the number of radio channels used for communication. $\mathbf{R}_0$ is obtained by taking the mean RSS measurements of each link on each frequency channel for a time duration when the monitored area is vacant. Let $\mathbf{R}(t) \in \mathbb{R}^{M \times |\mathbf{J}|}$ be the RSS measurements taken when the target is known to be inside the monitored region. Our objective is to estimate the true target location $\mathbf{x}(t)$ at time t , with knowledge of $\mathbf{R}_0$, $\mathbf{R}(t)$, and the true locations of the N wireless nodes.

4.3 Proposed Multi-Channel Geometric Filter algorithm

This section introduces the proposed multi-channel geometric filter (MCGF) algorithm for DFL in detail. The MCGF algorithm shares some features with the GF algorithm discussed in Chapter 3, specifically the methods for solving the link intersections, link and point filtering, and computation of the RSS-based and distance-based weights. For clarity and the reader's convenience,

we briefly review these methods in this section as well.

4.3.1 Calibration period

During the calibration period, the RSS of each link i on each frequency channel j are collected to determine the baseline RSS measurement matrix, static fade levels, and detection thresholds.

The static fade level $\bar{F}(i, j)$ for a link i on channel j is defined by [24], [31] as the difference between the RSS predicted by a radio propagation model and the mean calibration RSS $\bar{R}(i, j)$ shown in (4.2). Since $\bar{R}(i, j)$ is obtained when the target is not present in the monitored region, it is assumed that $\bar{S}(i, j) \approx 0$. Thus, (4.2) can be written as

$$\bar{F}(i, j) = \bar{R}(i, j) - \hat{R}(i, j), \quad (4.4)$$

where $\hat{R}(i, j) = P_i(j) - L(i)$ is the predicted RSS using the radio propagation model. For the predicted RSS, we use the log-distance path loss model [42],

$$\hat{R}(i, j) = P_0(j) - 10\eta \log_{10} \frac{d_i}{d_0} \quad (4.5)$$

where $P_0(j)$ is the loss on channel j at a reference distance of d_0 , η is the path loss exponent, and d_i is the distance between the nodes that make up the link i . The reference loss $P_0(j)$ is obtained by setting d_0 to the length of the shortest links in the network and taking the mean RSS of those links on channel j as $P_0(j)$. With knowledge of the link lengths d_i and $\bar{R}(i, j)$ for all links i on channels j , the path loss exponent η is estimated using a

linear least-squares fit. The fade levels $\bar{F}(i, j)$ are then obtained using (4.4). Recall that channels with large positive values of $\bar{F}(i, j)$, i.e. anti-fade links, are preferred for DFL since links on these channels typically experience a decrease in RSS when the target is obstructing the link line.

To determine the link-specific detection thresholds, the standard deviation values of the RSS measurements for link i on channel j are calculated offline during the calibration period. The standard deviations $\sigma(i, j)$ are estimated using the square root of the sample variance, i.e.

$$\sigma(i, j) = \sqrt{\frac{1}{N_{cal} - 1} \sum_{n=1}^{N_{cal}} (R(i, j, n) - \bar{R}(i, j))^2}, \quad (4.6)$$

where N_{cal} is the number of RSS measurements for link i on channel j during the calibration period. The values of $\sigma(i, j)$ are then combined using weights derived from $\bar{F}(i, j)$ to generate the detection threshold γ_i . Since the fade levels $\bar{F}(i, j)$ are signed values, the scheme from [25] is adopted wherein the minimum fade level for each link i is subtracted to obtain $\hat{F}(i, j)$,

$$\hat{F}(i, j) = \bar{F}(i, j) - \min_j \bar{F}(i, j). \quad (4.7)$$

This scheme assigns a weight of zero to the channel in deepest fade. The combined standard deviations for link i across all channels $j \in \mathbf{J}$ can then be computed as

$$\bar{\sigma}_i = \frac{\sum_{j \in \mathbf{J}} \hat{F}(i, j) \cdot \sigma(i, j)}{\sum_{j \in \mathbf{J}} \hat{F}(i, j)}. \quad (4.8)$$

By weighting the values of $\sigma(i, j)$ with $\hat{F}(i, j)$, deep fade channels are given lower weights and thus have a reduced influence on the detection threshold

for link i . The detection threshold γ_i for link i is then computed by

$$\gamma_i = C \cdot \bar{\sigma}_i , \quad (4.9)$$

where C is a user-defined threshold parameter.

4.3.2 Detecting the target-affected links

To detect the target-affected links, the absolute difference between the baseline RSS measurements for link i and its current RSS measurements is computed,

$$\Delta R(i, j, t) = |\bar{R}(i, j) - R(i, j, t)| , \quad (4.10)$$

where $\bar{R}(i, j)$ and $R(i, j, t)$ correspond to the i - j -th element of $\mathbf{R}_0 \in \mathbb{R}^{M \times |\mathbf{J}|}$ and $\mathbf{R}(t) \in \mathbb{R}^{M \times |\mathbf{J}|}$, respectively. The mean absolute difference of RSS for link i across all channels $j \in \mathbf{J}$ is then computed[25],

$$\Delta \bar{R}(i) = \frac{\sum_{j \in \mathbf{J}} \hat{F}(i, j) \cdot \Delta R(i, j, t)}{\sum_{j \in \mathbf{J}} \hat{F}(i, j)} . \quad (4.11)$$

By taking the weighted sum of absolute differences in the RSS using weights derived from the link fade levels, the measured differences on the more informative anti-fade channels have a greater contribution to the overall sum $\Delta \bar{R}(i)$. The RSS of some links in the network are attenuated or amplified due to the target's presence, depending on whether a link is in deep fade or anti-fade. A link i is classified as target-affected if its mean absolute difference $\Delta \bar{R}(i)$ exceeds the detection threshold, i.e. $\Delta \bar{R}(i) > \gamma_i$. Thus, the set

Procedure 4.1 Generate the elements of the set $\mathbf{L}(t)$

Input: The set of target-affected links $\mathbf{L}_s(t)$ using (4.12).

Output: The set of target-affected links for localization, $\mathbf{L}_\phi(t)$.

- 1: Initialize $\mathbf{L}_\phi(t)$ to the empty set.
 - 2: Sort the set $\mathbf{L}_s(t)$ in descending order of $\Delta\bar{R}(i)$.
 - 3: Use (4.13) to solve for ϕ .
 - 4: $\mathbf{L}_\phi(t)$ gets the first ϕ elements of $\mathbf{L}_s(t)$.
-

$\mathbf{L}_s(t)$ of target-affected links at time t is defined as

$$\mathbf{L}_s(t) = \{i \mid \Delta\bar{R}(i) > \gamma_i\}. \quad (4.12)$$

In multipath-rich environments, it is observed that using all elements of the set $\mathbf{L}_s(t)$ for localization can result in poor accuracy. To mitigate this, the MCGF algorithm selects only the top ϕ links of the set $\mathbf{L}_s(t)$ that have the largest values of $\Delta\bar{R}(i)$. The value of ϕ is controlled by the user-defined parameter P_n , where $0 < P_n \leq 1$. The value of ϕ is computed by

$$\phi = \max(\text{floor}(|\mathbf{L}_s(t)| \times P_n), 2) \quad (4.13)$$

where $|\mathbf{L}_s(t)|$ is the number of elements in $\mathbf{L}_s(t)$. Since the MCGF algorithm uses the intersection points of target-affected links as probable target locations, ϕ is configured to have a minimum value of 2, which is the minimum number of links required to generate an intersection point. The set $\mathbf{L}_\phi(t)$ of target-affected links to be used for localization is populated using Proc. 4.1.

4.3.3 Prior region

Due to multipath interference, some links far from the target can experience RSS changes. These outlier links reduce the localization accuracy as they may intersect other target-affected links at points far from the target. Similar to the GF algorithm, the MCGF algorithm also uses a circular prior region $\mathbf{P}(t)$ to (a) remove outlier points and links, and (b) assign weights to the probable target locations when generating a location estimate. Let $\hat{\mathbf{x}}(t-1)$ be the target location estimate generated by the MCGF algorithm for the previous time instant $t-1$. $\mathbf{P}(t)$ is defined as

$$\mathbf{P}(t) = \{\mathbf{x} \mid \|\mathbf{x} - \hat{\mathbf{x}}(t-1)\| < r\} \quad (4.14)$$

where $\mathbf{x}$ is a point inside the monitored area, r is the radius of $\mathbf{P}(t)$, and $\|\cdot\|$ is the Euclidean norm. For $t=1$, r is set to a large value and all intersection points of the target-affected links are used to generate a location estimate.

4.3.4 Outlier link filtering

Let the line segment $\overline{\mathbf{U}_i\mathbf{V}_i}$, with endpoints at node locations $\mathbf{U}_i$ and $\mathbf{V}_i$ represent the target-affected link $i \in \mathbf{L}_\phi(t)$. Let $\boldsymbol{\rho}_i$ be the point on the line segment $\overline{\mathbf{U}_i\mathbf{V}_i}$ closest to $\hat{\mathbf{x}}(t-1)$, the center of $\mathbf{P}(t)$. A link i is considered as an outlier and is removed from the set $\mathbf{L}_\phi(t)$ if all of the points $\mathbf{U}_i$, $\mathbf{V}_i$, and $\boldsymbol{\rho}_i$ on the line segment $\overline{\mathbf{U}_i\mathbf{V}_i}$ are outside the prior region $\mathbf{P}(t)$. For clarity, let $\mathbf{L}(t)$ be the set of valid target-affected links,

$$\mathbf{L}(t) = \{ i \mid (d_{\mathbf{U}_i} \leq r) \vee (d_{\mathbf{V}_i} \leq r) \vee (d_{\boldsymbol{\rho}_i} \leq r), i \in \mathbf{L}_\phi(t) \} , \quad (4.15)$$

where $d_{\mathbf{U}_i} = \|\mathbf{U}_i - \hat{\mathbf{x}}(t-1)\|$, $d_{\mathbf{V}_i} = \|\mathbf{V}_i - \hat{\mathbf{x}}(t-1)\|$, $d_{\boldsymbol{\rho}_i} = \|\boldsymbol{\rho}_i - \hat{\mathbf{x}}(t-1)\|$, and $\vee$ is the logical OR operator.

4.3.5 Generating probable target locations and RSS-based weights

Since the target-affected links tend to intersect near the target's true location, the intersection points of all the links in $\mathbf{L}(t)$ are considered as probable target locations. In this step, each pairwise combination of line segments $\overline{\mathbf{U}_m \mathbf{V}_m}$ and $\overline{\mathbf{U}_n \mathbf{V}_n}$ corresponding to links m and n in $\mathbf{L}(t)$ are tested for intersection. If $\overline{\mathbf{U}_m \mathbf{V}_m}$ and $\overline{\mathbf{U}_n \mathbf{V}_n}$ intersect, their intersection point $\mathbf{q}_i(t)$ is added to the set of all intersection points $\mathbf{Q}_s(t)$. The sum of the mean absolute RSS differences of the intersecting links m and n is also recorded for later use as a weight when a location estimate is generated,

$$w_{\Delta R}^i = \Delta \bar{R}(m) + \Delta \bar{R}(n) , \quad (4.16)$$

where $w_{\Delta R}^i$ is the non-normalized RSS-based weight for intersection point $\mathbf{q}_i(t)$. In this way, higher weights are assigned to intersection points of links that experience greater RSS change due to the target's presence. All weights corresponding to each intersection point in $\mathbf{Q}_s(t)$ are stored in the set $\mathbf{W}_{\Delta R}^s$.

4.3.6 Generating fade level-based weights

Given the fade level information for each link i across all channels $j \in \mathbf{J}$, the MCGF algorithm also generates a set of fade level-based weights which can further improve the tracking accuracy. Since target-affected links with positive fade levels (anti-fade links) are more likely to have the target

crossing its link line, the intersection point of two anti-fade links is given a higher weight. Since a link may be in deep fade (i.e. with negative fade level) on one channel and in anti-fade on another, the sum of all fade levels for the link is used as a measure of its overall fade level. Links which are in anti-fade in all channels $j \in \mathbf{J}$ will tend to have higher overall fade levels as compared to links which are in deep fade in some or all of the channels. Thus, for the intersection point $\mathbf{q}_i(t) \in \mathbf{Q}(t)$ of links m and n , its corresponding fade level-based weight w_F^i is

$$w_F^i = \sum_{j \in |\mathbf{J}|} \bar{F}(m, j) + \sum_{j \in |\mathbf{J}|} \bar{F}(n, j) . \quad (4.17)$$

The fade level-based weights corresponding to each intersection point $\mathbf{q}_i(t) \in \mathbf{Q}(t)$ are stored in the set $\mathbf{W}_F^s$. Depending on the overall fade levels of the intersecting links, the weights w_F^i may take on negative values. These are handled in the final stage of the algorithm.

4.3.7 Removing improbable target locations

To improve the target location estimate, points that are outside the prior region $\mathbf{P}(t)$ are removed from the set $\mathbf{Q}_s(t)$, and only the points inside $\mathbf{P}(t)$ are retained. Let $\mathbf{Q}(t)$ be the set of intersection points that lie inside $\mathbf{P}(t)$,

$$\mathbf{Q}(t) = \{\mathbf{q}_i(t) \mid \|\mathbf{q}_i(t) - \hat{\mathbf{x}}(t-1)\| \leq r, \mathbf{q}_i(t) \in \mathbf{Q}_s(t)\}. \quad (4.18)$$

The set of weights $\mathbf{W}_{\Delta R}^s$ and $\mathbf{W}_F^s$ must also be updated, removing the weights corresponding to the points outside $\mathbf{P}(t)$. Let the set of updated

RSS-based weights $\mathbf{W}_{\Delta R}$ be defined as

$$\mathbf{W}_{\Delta R} = \{w_{\Delta R}^i \mid \mathbf{q}_i(t) \in \mathbf{Q}(t)\} , \quad (4.19)$$

and the set of updated fade level-based weights $\mathbf{W}_F$ be similarly defined as,

$$\mathbf{W}_F = \{w_F^i \mid \mathbf{q}_i(t) \in \mathbf{Q}(t)\} . \quad (4.20)$$

4.3.8 Distance-based weights

A distance-based weight w_d^i is generated for each intersection point in $\mathbf{Q}(t)$. The weight w_d^i is dependent on the distance of the point $\mathbf{q}_i(t)$ to the previous location estimate $\hat{\mathbf{x}}(t-1)$, and is computed as

$$w_d^i = 1 - \frac{\|\mathbf{q}_i(t) - \hat{\mathbf{x}}(t-1)\|}{r} . \quad (4.21)$$

This weighting scheme assigns higher weights to points that are closer to $\hat{\mathbf{x}}(t-1)$. The set $\mathbf{W}_d$ of distance-based weights for the points in $\mathbf{Q}(t)$ is defined as

$$\mathbf{W}_d = \{w_{d_i} \mid \mathbf{q}_i(t) \in \mathbf{Q}(t)\} . \quad (4.22)$$

4.3.9 Overall algorithm in generating a location estimate

Before a location estimate can be generated, the fade level-based weights in $\mathbf{W}_F$ must first be modified such that none of its elements are negative. This is achieved by subtracting the lowest valued element of $\mathbf{W}_F$ from all

its elements. The updated set of fade level-based weights $\mathbf{W}_F^+$ is

$$\mathbf{W}_F^+ = \{w_F^i - \min_i(\mathbf{W}_F) \mid w_F^i \in \mathbf{W}_F\} . \quad (4.23)$$

With the set of points $\mathbf{Q}(t)$, and the weights $\mathbf{W}_{\Delta R}$, $\mathbf{W}_F^+$, and $\mathbf{W}_d$, a target location estimate $\hat{\mathbf{x}}(t)$ is generated,

$$\hat{\mathbf{x}}(t) = \sum_{i=1}^{|\mathbf{Q}(t)|} w_i \cdot \mathbf{q}_i(t) \quad (4.24)$$

where

$$w_i = \frac{w_{\Delta R}^i \cdot w_d^i \cdot w_F^{i+}}{\sum (w_{\Delta R}^i \cdot w_d^i \cdot w_F^{i+})} \quad (4.25)$$

are the normalized weights, and $|\mathbf{Q}(t)|$ denotes the number of members in the set $\mathbf{Q}(t)$. The MCGF algorithm is summarized in Proc. 4.2.

4.4 Experiment setup

Experiments have been conducted in both an uncluttered outdoor environment and a highly cluttered indoor environment to evaluate the performance of MCGF. In both experiments, the wireless nodes are placed at the monitored region's perimeter. The experimental setup is similar to that of Chapter 3, with some slight differences in the target's track, the size of the outdoor area, and the operation of the nodes.

The nodes are 2.4 GHz IEEE 802.15.4 transceivers placed on stands with a height of 1 m. The nodes communicate on channels 11, 18, and 26, and broadcast transmissions in sequence of their node IDs. Upon reset, all nodes wait for a transmission on the default channel. The data collection is initiated by a node broadcasting a packet containing its unique node ID

Procedure 4.2 Multi-channel geometric filter algorithm

- 1: Collect N_{cat} measurements of the RSS of each link i on each channel $j \in \mathbf{J}$.
 - 2: Solve for the fade levels $\bar{F}(i, j)$ using (4.4) and (4.5).
 - 3: Calculate the standard deviation of RSS $\sigma(i, j)$ using (4.6).
 - 4: Determine the detection threshold γ_i using (4.7), (4.8), and (4.9).
 - 5: **for** $t \geq 1$ **do**
 - 6: Solve for $\Delta\bar{R}(i)$ using (4.10) and (4.11) for all links i .
 - 7: Generate the initial set of target-affected links $\mathbf{L}_s(t)$ using (4.12).
 - 8: Determine the set $\mathbf{L}_\phi(t)$ using Proc. 4.1.
 - 9: Build the prior region $\mathbf{P}(t)$ using (4.14).
 - 10: Remove outlier links to obtain $\mathbf{L}(t)$ using (4.15).
 - 11: Solve for the intersections of all segments in $\mathbf{L}(t)$ and the weights $w_{\Delta R}^i$ and w_F^i as described in Section 4.3.5.
 - 12: Remove outlier points to obtain $\mathbf{Q}(t)$ using (4.18).
 - 13: Update the set of RSS-based weights and fade level-based weights using (4.19) and (4.20), respectively.
 - 14: Generate the set of distance-based weights using (4.21) and (4.22).
 - 15: Generate the location estimate $\hat{\mathbf{x}}_t$ using (4.23), (4.24), and (4.25).
 - 16: **end for**
-

and a copy of its RSS array, which contains the RSS between the node and all other nodes in the network. Upon reception of this packet, the other nodes measure the RSS between the transmitter and itself, and update their respective RSS arrays. In addition, each node also checks if it's their turn to broadcast and if it is time to switch to a different channel. For data recording, a basestation node connected to a mobile computer listens to all broadcasts and logs the RSS arrays of each node. The average time between transmissions is $T_{tx} = 10$ ms.

For both experiments, an empty scene calibration period of $T_{cat} = 60$ s

is performed to obtain the baseline RSS and estimate the fade levels and variances of each link in the network. In both scenes, a single human target walks on a pre-defined track inside the monitored region at a speed of $v \approx 0.5$ m/s. Markers were placed on the true locations along the track, and a metronome is used to assist the target in maintaining constant speed. The target's location is estimated once per second, and each experiment is repeated for 50 trials. The tracking error is defined as the distance between the estimated target location and the known true target location.

The tracking performance of MCGF is compared to the RTI-based methods of [22] and [25]. The RTI method in [25] uses fade-level weighted averaging to combine RSS measurements from different channels, while [22] uses an unweighted average. We refer to the method of [25] as RTI-FL, and [22] as RTI-A in the sequel. For tracking, a Kalman filter was applied to the location estimates generated by RTI-A and RTI-FL. The Kalman filter and image reconstruction parameters of the RTI-based methods were tuned to give the best performance for each scene. In addition to RTI-A and RTI-FL, the performance of MCGF was also compared with its MCGF-A and MCG-KF variants. The MCGF-A variant does not use fade level-weighted RSS. Instead, a link's RSS measurements across all channels is simply averaged, similar to [22]. The MCG-KF uses fade level-weighted RSS, but does not use the link and point filters, i.e. all target-affected links and intersection points are used to generate a location estimate. Furthermore, MCG-KF applies a Kalman filter to the location estimates for tracking. The GF-C11, GF-C18, and GF-C26 algorithms use the original GF algorithm [48], a fixed threshold of $\gamma_i = 3$ dB, and only on single-channel RSS measurements on Channels 11, 18, and 26, respectively.

4.4.1 Scene 1, Uncluttered outdoor experiment

For the uncluttered outdoor experiment, 16 nodes are deployed to cover a $6\text{ m} \times 6\text{ m}$ area, with the nodes spaced 1.5 m apart. For Scene 1, the parameters of MCGF are threshold parameter $C = 2.5$, radius of prior region $r = 2\text{ m}$, and $P_n = 0.5$.

4.4.2 Scene 2, Highly cluttered indoor experiment

The site of the highly cluttered indoor experiment is inside a laboratory with numerous obstructions such as computers, monitors, and other equipment. In this scene, 26 nodes are deployed to cover a $8.5\text{ m} \times 8.5\text{ m}$ area, with the nodes irregularly placed at the perimeter. For Scene 2, the parameters of MCGF are threshold parameter $C = 1$, radius of prior region $r = 3\text{ m}$, and $P_n = 0.1$.

4.5 Experimental results

The mean tracking results for Scenes 1 and 2 obtained using MCGF are shown in Figs. 4.1a and 4.1b, respectively. From Fig. 4.1a, it is observed that MCGF can track the target well. The Scene 1 tracking error statistics for MCGF and its variants are shown in Table 4.1, where it is observed that MCGF has the lowest tracking root-mean-square error (RMSE). It can also be seen that while the MCGF-A variant uses channel diversity to reduce the effects of noise, it is outperformed by the single-channel variant GF-C11. Since MCGF-A simply averages the RSS across all channels, its performance can be severely degraded if one of the channels is noisy, as in the case of C26 in the Scene 1 experiment. This can be mitigated by measuring the

Table 4.1: Tracking error statistics - Scene 1

Algorithm	Median [m]	Mean [m]	RMSE [m]
MCGF	0.354	0.411	0.496
MCGF-A	0.528	0.673	0.848
MCG-KF	0.406	0.530	0.691
GF-C11	0.478	0.609	0.768
GF-C18	0.509	0.661	0.848
GF-C26	0.623	0.768	0.944
RTI-A	0.605	0.778	0.996
RTI-FL	0.419	0.574	0.763

RSS across more channels, at the expense of increased RSS scanning times. By using the link fade levels as weights, MCGF and its MCG-KF variant are less prone to noisy channels. The lower RMSE of MCGF compared to MCG-KF can be attributed to its use of the link and point filters to remove outlying points and links. The empirical cumulative distribution function (ECDF) of the tracking errors for MCGF and its variants are shown in Fig. 4.2. The MCGF algorithm also outperforms RTI-FL and RTI-A, as shown in the ECDF of tracking errors in Fig. 4.3. It can also be observed from Fig. 4.3 and Table 4.1 that the fade-level weighted methods MCGF and RTI-FL outperform their respective unweighted average counterparts MCGF-A and RTI-A.

(a)

(b)

Figure 4.1: Visualized tracking results for (a) Scene 1 and (b) Scene 2 using the MCGF algorithm.

Figure 4.2: Scene 1, ECDF of tracking errors for the multi-channel and single-channel GF variants.

Figure 4.3: Scene 1, ECDF of tracking errors for the multi-channel GF and RTI variants.

Figure 4.4: Scene 2, ECDF of tracking errors obtained using the tested algorithms.

Table 4.2: Tracking error statistics - Scene 2

Algorithm	Median [m]	Mean [m]	RMSE [m]
MCGF	0.592	0.658	0.754
MCGF-A	0.723	0.756	0.843
MCG-KF	0.827	0.920	1.06
GF-C11	0.981	1.09	1.26
GF-C18	1.01	1.10	1.23
GF-C26	0.977	1.17	1.47
RTI-A	1.35	1.53	1.79
RTI-FL	0.708	1.01	1.32

For Scene 2, the tracking error is larger for all algorithms due to the more challenging indoor environment. Nevertheless, the MCGF algorithm can track the target with sub-meter accuracy. Fig. 4.1b shows the mean tracking results obtained using MCGF for Scene 2. The tracking error statistics for Scene 2 are shown in Table 4.2. Among the algorithms compared in Scene 2, the MCGF algorithm performs best, with an improvement of 16.3% in tracking RMSE over the next-best algorithm. Of the tested algorithms, only MCGF and MCGF-A achieved sub-meter accuracy in Scene 2. MCGF and MCGF-A's use of the link and point filters in the cluttered environment of Scene 2 allow them both to outperform the MCG-KF variant. The ECDF of the tracking errors of all methods in Scene 2 are shown in Fig. 4.4.

4.5.1 Performance under different values of r

In this section, the MCGF algorithm's tracking performance is investigated as the radius of $\mathbf{P}(t)$ is increased. As the radius r is increased, the area of the circular prior region $\mathbf{P}(t)$ increases as well. Let us define $\alpha = \frac{A_{\mathbf{P}(t)}}{A_D}$ as the ratio of the average area of the prior region $\mathbf{P}(t)$ and the monitored region D . For Scenes 1 and 2, $A_D = 36 \text{ m}^2$ and $A_D = 72.25 \text{ m}^2$, respectively. $A_{\mathbf{P}(t)}$ is computed as the area of a circle with radius r , and concentric with the square monitored region D . To illustrate the effect of the radius r on the tracking performance of MCGF for both scenes, the tracking RMSE is shown in Fig. 4.5 as r and consequently α is increased. For both scenes, the tracking performance of MCGF is good when the ratio α is between 0.35 and 0.7. This corresponds to $2 \text{ m} \leq r \leq 2.85 \text{ m}$ for Scene 1, and $2.84 \text{ m} \leq r \leq 4 \text{ m}$ for Scene 2. It is observed from Fig. 4.5 that MCGF outperforms MCGF-A in nearly all tested values of α for both scenes. At high values of α ,

the prior region $\mathbf{P}(t)$ is less informative and more links are considered for localization, even if a majority do not pass the vicinity of the target. In this situation, MCGF gives better results as links in deep fade channels are given less weight as compared to the more informative links in anti-fade channels.

4.5.2 Performance under different values of C and P_n

The values of the parameters C and P_n greatly affect the tracking performance of the MCGF algorithm. The parameter C determines the threshold for detecting the target-affected links, while the parameter P_n determines the percentage of the target-affected links that are used for localization. Fig. 4.6 shows contour plots for Scene 1 and 2 of the tracking RMSE as the parameters P_n and C are varied. It is seen from Fig. 4.6 that certain combinations of parameter values for P_n and C yield high tracking accuracy. For both scenes, using a low value of P_n with a high value of C , and vice versa, results in large tracking errors. In the first case, few links are used for localization since the high value of C results in failed detection of target-affected links due to the high detection threshold γ_i . The number of target-affected links are further reduced by the low value of P_n . In the second case where a low value of C is used along with a high value of P_n , the algorithm is more prone to RSS measurement noise. Links with minimal change in RSS are treated as target-affected links due to the low value of C , and the high value of P_n results in a majority of the falsely-detected links being used for localization. For Scene 1, a good value for P_n is between 0.5 and 0.9, with the value of $C = 2.5(P_n) + 1.3$. This equation was derived using a linear fit to the contour at the 0.5 m RMSE level. For Scene 2, sub-meter accuracy is achieved when P_n is between 0.1 and 0.85, with $C = 2(P_n) + 1.2$. Fig. 4.7

(a)

(b)

Figure 4.5: Tracking RMSE of MCGF as α is varied for (a) Scene 1 and (b) Scene 2.

(a)

(b)

Figure 4.6: Contour plots of the tracking RMSE of MCGF as P_n and C are varied for (a) Scene 1 and (b) Scene 2.

(a)

(b)

Figure 4.7: Plots of the tracking RMSE for MCGF as C is varied for (a) Scene 1 and (b) Scene 2. The dashed horizontal lines indicate the RMSE levels for fixed values of γ_i .

shows the tracking RMSE of MCGF as C is varied from 0.01 to 5 for Scene 1 and 2, with the default values of P_n for each scene. In both scenes, using the link-specific thresholds resulted in lower tracking RMSE as compared to using a fixed threshold value.

4.5.3 Effect of P_n and the link filter

Figure 4.8: $|L'(t)| / L_{P(t)}$ as P_n is varied for both scenes.

In this section, the effect of the parameter P_n and the link filter of Eq. (4.15) on the number of links used for localization is evaluated. The average link density ρ_L is defined as the total number of links in the network divided by the area of the monitored region D . For Scene 1, $N = 16$ nodes, $A_D = 36 \text{ m}^2$, and $\rho_L = 3.33/\text{m}^2$. For Scene 2, $N = 26$ nodes, $A_D = 72.25 \text{ m}^2$, and $\rho_L = 4.5/\text{m}^2$. With knowledge of the radius of $\mathbf{P}(t)$ in both scenes, the average number of links crossing $\mathbf{P}(t)$, $L_{P(t)} = \pi r^2 \rho_L$, can be estimated. For Scene 1, $L_{P(t)} = 41.89$, and $L_{P(t)} = 127.19$ for Scene 2. In practice,

Table 4.3: Effect of w_F^{i+} on RMSE of MCGF

Weights used	Scene 1	Scene 2
$w_F^{i+}, w_d^i, w_{\Delta R}^i$	0.496 m	0.754 m
$w_d^i, w_{\Delta R}^i$	0.526 m	0.810 m

not all links that cross $\mathbf{P}(t)$ are used for localization. The average number of links used for localization $|\mathbf{L}'(t)|$ is obtained by counting the elements of $\mathbf{L}(t)$ across all sampling instants and taking its average. In Fig. 4.8, the ratio of $|\mathbf{L}'(t)|$ and $L_{\mathbf{P}(t)}$ is shown as P_n is varied, for both scenes. Fig 4.8 shows that even with $L_{\mathbf{P}(t)}$ of Scene 2 more than three times greater than that of Scene 1, only up to 50% of these links are used for localization. The rest of the links are not used despite being within the target's vicinity, as some may be unaffected by the target, having low RSS magnitude change due to being in deep fade.

4.5.4 Effect of fade level-based weights w_F^{i+}

The fade level-based weights w_F^{i+} give larger weights to intersection points of links with higher overall fade levels. Table 4.3 shows the tracking RMSE for MCGF when w_F^{i+} is used and when only the RSS-based weights $w_{\Delta R}^i$ and distance-based weights w_d^i are used. From Table 4.3, it is noted that the use of the overall fade level information of the intersecting links as weights on the probable target locations results in increased tracking accuracy.

4.5.5 Average execution time

All the algorithms that were evaluated in the preceding sections were implemented in C++, compiled without optimizations, and ran on a desktop

Table 4.4: Average execution times

Scenes	RTI -FL	MCGF
Scene 1, $M = 120$	$59.3 \mu s$	$6.40 \mu s$
Scene 2, $M = 325$	$93.5 \mu s$	$9.13 \mu s$

computer with a 3.40 GHz central processing unit (CPU) and 8 GB random access memory (RAM). The GF-based algorithms' execution time scale with the number of links in the network, while the RTI-based algorithms' execution time scale with the number of links and with the resolution of the desired image. For the experiments in Scene 1 and 2, the number of pixels used for the RTI-based images are 1600 and 1156, respectively. Due to the grid-based nature of the RTI-based methods, they have higher computational overhead as compared to the GF-based algorithms, which use simple geometric objects to represent links, probable target locations, and prior information on the target's location. It must be noted that there is little difference in execution times between MCGF and MCGF-A, and between RTI-FL and RTI-A, as they differ only in how the differential RSS measurements from multiple channels are combined. The average execution time to generate one location estimate of the MCGF and RTI-FL algorithms for both scenes are shown in Table 4.4. It is observed from Table 4.4 that the GF-based algorithms execute much faster when compared to the RTI-based algorithms, and the difference in execution times is more pronounced when there is a higher number of network links.

4.6 Summary

The use of RSS-based DFL for tracking a target indoors is made difficult by the existence of multiple propagation paths between nodes in the wireless network. Channel diversity has been proposed in the literature to mitigate the multipath fading effects and increase the accuracy of RSS-based DFL systems. In this chapter, we have presented an enhanced geometric filter algorithm, referred to as the “Multi-channel geometric filter” (MCGF) algorithm. The MCGF algorithm is a modified version of the GF algorithm discussed in Chapter 3, with the following improvements:

- The MCGF algorithm uses channel diversity to diminish the effects of multipath fading on the RSS of the networks’ links.
- The MCGF algorithm uses link-specific detection thresholds which are derived from each link’s fade level and RSS variance across multiple radio channels. This approach takes advantage of the unique characteristic of each link in the network and aims to improve the detection of target-affected links. In addition, the algorithm also ranks the target-affected links in terms of their change in RSS and selects only a fraction of these links for localization.
- The MCGF algorithm uses a weighted mean with weights derived from the static fade levels of the target-affected links to generate a location estimate from their intersection points, giving larger weights to intersection points of predominantly anti-fade links.

Through experiments in outdoor and cluttered indoor environments, we have shown that the MCGF algorithm outperforms existing multi-channel RTI-based methods in terms of tracking accuracy and average execution time.

Chapter 5

Conclusion and Future Work

5.1 Summary

In this thesis, the geometric method for device-free localization (DFL) has been investigated and shown to outperform other more sophisticated DFL algorithms in terms of tracking accuracy while maintaining its low computational overhead. In particular, we have proposed and developed the geometric filter (GF) algorithm, which uses prior information to improve location estimates obtained using geometric methods for DFL. To further improve the accuracy of GF in indoor environments, the multi-channel geometric filter (MCGF) algorithm has been proposed and developed. It is a variation of GF with extended capability to work with multi-channel RSS measurements, as well as improved target-affected link detection, link selection, and location estimation methods.

Geometric methods for DFL are known to be prone to noise, since some outlier links may be detected at sections of the monitored region that are far from the target's true location. Previous works that use geometric methods for DFL relied on the multiplicity of target-affected links that intersect near

the target’s true location to mitigate the effect of outlier links, essentially averaging out the noise. In this thesis, filters have been proposed and used to remove outlier links before location estimates are made. This leads to significant improvements in tracking accuracy. The proposed GF algorithm applies locational filters based on the previous target location estimate to remove outlier points and links. In addition, the GF algorithm uses the prior location estimate to assign distance-dependent weights to probable target locations, which was shown to further improve the succeeding location estimate.

Indoor environments are challenging for DFL algorithms due to the presence of obstructions and objects, resulting in a rich multipath environment. In this thesis, the MCGF algorithm — a variant of the GF algorithm that uses channel diversity has been presented to mitigate the effects of multipath fading. It combines measurements of the absolute change in a link’s RSS across multiple channels, uses a link-specific detection threshold to determine the target-affected links, and refines target location estimates further with fade level-based weights.

The tracking accuracy and execution time of the proposed GF and MCGF algorithms have been evaluated through experiments conducted in uncluttered outdoor and cluttered indoor environments. Within each environment, a wireless network was deployed at the perimeter of a square deployment region. The wireless nodes were configured to communicate using a turn-based protocol to reduce the probability of collisions. Each node measures the RSS of each link it is involved in and broadcasts its measurement array during its turn to transmit. While the nodes communicate, the target walks along a pre-defined track at constant speed. A basestation computer was configured

to log all RSS measurements and transmissions from the network.

The experimental results demonstrate that the proposed GF and MCGF algorithms outperform existing DFL methods in terms of tracking accuracy and average execution time. The GF algorithm has a tracking RMSE that is up to 65% lower as compared to existing radio tomographic imaging (RTI) and statistical inversion methods for DFL. The MCGF algorithm has a tracking RMSE that is up to 50% lower as compared to single-channel GF-based methods and RTI-based methods that also use channel diversity. The GF and MCGF algorithm's use of simple geometric objects to represent links, probable target locations, and prior target information allows them to generate location estimates with appreciably lower execution times as compared to existing RTI and statistical inversion methods. From the experimental results, the GF and MCGF algorithms were shown to have average execution times that are up to $10\times$ faster compared to RTI and up to $550\times$ faster as compared to the statistical inversion methods. Finally, the performance of the proposed algorithms under different parameter settings have also been investigated experimentally.

5.2 Future Work

An automatic method for selecting the parameter values for the GF and MCGF algorithms is important for applying these methods in various operating scenarios. For example, there may be cases when there is no prior information about the target's maximum speed. In such a scenario, the radius of the prior region will have to be determined automatically. Methods for automatic parameter value selection for the prior region's radius, as well as the threshold and link selection parameters of the GF and MCGF

algorithms is an area for future work.

Practical applications of DFL are not limited to tracking the movements of a single device-free target. Thus, a logical further development of the GF and MCGF algorithms is to extend their capabilities for multiple target tracking. In a scenario where multiple targets are to be localized, clustering algorithms such as k -means or mean-shift could be used to detect the number of previously un-localized targets and the algorithm can then assign separate filters for each cluster. A multi-target GF algorithm must also perform track management, deleting tracks for targets that have left the monitored region and updating the existing tracks, and adding new tracks for new targets. Finally, a multi-target GF algorithm's performance will have to be compared with its RTI and statistical inversion method counterparts.

Statistical inversion methods use a particle filter framework to accurately track the location of a device-free target. The use of a particle filter framework allows nonlinear dynamical models of the target motion to be utilized. However, particle filters typically require a large number of particles to accurately estimate the posterior distribution of the target's state. The large number of particles, coupled with complex observation models, results in high computational costs for these methods. A possible topic for future work is to hybridize geometric methods with statistical inversion methods for DFL. A low-complexity geometric method could be used initially to propose particles, which are then further processed by the particle filter. Instead of processing all the links as is typically done in statistical inversion method, a locational filter could also be used to remove outlier links and particles. A hybridized approach could potentially reduce the number of particles required for accurate tracking and improve the execution time of statistical

inversion methods.

Author's Publications on WSN Localization

- M. C. R. Talampas and K. S. Low, “A geometric filter algorithm for robust device-free localization in wireless networks,” *Industrial Informatics, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015, ISSN: 1551-3203. DOI: 10.1109/TII.2015.2433211.
- M. C. R. Talampas and K. S. Low, “An enhanced geometric filter algorithm with channel diversity for device-free localization,” *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 2, pp. 378–387, 2016, ISSN: 0018-9456. DOI: 10.1109/TIM.2015.2490818.
- M. C. R. Talampas and K. S. Low, “Geometry-based algorithms for device-free localization with wireless sensor networks,” in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2014 IEEE Ninth International Conference on*, 2014, pp. 1–6. DOI: 10.1109/ISSNIP.2014.6827625.
- M. C. R. Talampas and K. S. Low, “Integrating active and passive received signal strength-based localization,” in *Intelligent Signal Processing and Communication Systems (ISPACS), 2014 International Symposium on*, 2014, pp. 153–158. DOI: 10.1109/ISPACS.2014.7024443.

Other Publications

M. C. R. Talampas and K. S. Low, “Maximum likelihood estimation of ground truth for air quality monitoring using vehicular sensor networks,” in *TENCON 2012-2012 IEEE Region 10 Conference*, IEEE, 2012, pp. 1–6.

K. S. Low and M. C. R. Talampas, “Wireless sensor networks for intelligent transportation applications : A survey,” in *Industrial Wireless Sensor Networks: Applications, Protocols, and Standards*, V. C. Gungor and G. P. Hancke, Eds., CRC Press, 2013, ch. 3.

Bibliography

- [1] F. Chraim, Y. Erol, and K. Pister, “Wireless gas leak detection and localization,” *Industrial Informatics, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015, ISSN: 1551-3203. DOI: 10.1109/TII.2015.2397879.
- [2] K. Thongpull, N. Jindapetch, and W. Teerapabkajornet, “Wireless ESD event locator systems in hard disk drive manufacturing environments,” *Industrial Electronics, IEEE Transactions on*, vol. 60, no. 11, pp. 5252–5259, 2013, ISSN: 0278-0046. DOI: 10.1109/TIE.2012.2227911.
- [3] Z. Zhang, Z. Lu, V. Saakian, X. Qin, Q. Chen, and L.-R. Zheng, “Item-level indoor localization with passive UHF RFID based on tag interaction analysis,” *Industrial Electronics, IEEE Transactions on*, vol. 61, no. 4, pp. 2122–2135, 2014, ISSN: 0278-0046. DOI: 10.1109/TIE.2013.2264785.
- [4] S. Ivanov and E. Nett, “Localization-based radio model calibration for fault-tolerant wireless mesh networks,” *Industrial Informatics, IEEE Transactions on*, vol. 9, no. 1, pp. 246–253, 2013, ISSN: 1551-3203. DOI: 10.1109/TII.2012.2203311.

- [5] H. Yan, H. Huo, Y. Xu, and M. Gidlund, “Wireless sensor network based e-health system : Implementation and experimental results,” *Consumer Electronics, IEEE Transactions on*, vol. 56, no. 4, pp. 2288–2295, 2010, ISSN: 0098-3063. DOI: 10.1109/TCE.2010.5681102.
- [6] D. Niculescu and B. Nath, “Ad hoc positioning system (APS),” in *Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE*, vol. 5, 2001, 2926 –2931 vol.5. DOI: 10.1109/GLOCOM.2001.965964.
- [7] K. S. Low, H. Nguyen, and H. Guo, “A particle swarm optimization approach for the localization of a wireless sensor network,” in *Industrial Electronics, 2008. ISIE 2008. IEEE International Symposium on*, 2008, pp. 1820–1825. DOI: 10.1109/ISIE.2008.4677205.
- [8] H. Guo, K. S. Low, and H. A. Nguyen, “Optimizing the localization of a wireless sensor network in real time based on a low-cost micro-controller,” *Industrial Electronics, IEEE Transactions on*, vol. 58, no. 3, pp. 741 –749, 2011, ISSN: 0278-0046. DOI: 10.1109/TIE.2009.2022073.
- [9] F. Reichenbach and D. Timmermann, “Indoor localization with low complexity in wireless sensor networks,” in *Industrial Informatics, 2006 IEEE International Conference on*, 2006, pp. 1018–1023. DOI: 10.1109/INDIN.2006.275737.
- [10] B.-F. Wu and C.-L. Jen, “Particle-filter-based radio localization for mobile robots in the environments with low-density WLAN APs,” *Industrial Electronics, IEEE Transactions on*, vol. 61, no. 12, pp. 6860–6870, 2014, ISSN: 0278-0046. DOI: 10.1109/TIE.2014.2327553.

- [11] H. Zou, H. Wang, L. Xie, and Q. S. Jia, "An rfid indoor positioning system by using weighted path loss and extreme learning machine," in *Cyber-Physical Systems, Networks, and Applications (CPSNA), 2013 IEEE 1st International Conference on*, 2013, pp. 66–71. DOI: 10.1109/CPSNA.2013.6614248.
- [12] F. Dwiya, M.-H. Lim, Y.-S. Ong, and B. Panigrahi, "Extreme learning machine for indoor location fingerprinting," *Multidimensional Systems and Signal Processing*, pp. 1–17, 2016, ISSN: 1573-0824. DOI: 10.1007/s11045-016-0409-0. [Online]. Available: <http://dx.doi.org/10.1007/s11045-016-0409-0>.
- [13] K. Kaemarungsi and P. Krishnamurthy, "Analysis of wlans received signal strength indication for indoor location fingerprinting," *Pervasive and Mobile Computing*, vol. 8, no. 2, pp. 292–316, 2012, Special Issue: Wide-Scale Vehicular Sensor Networks and Mobile Sensing, ISSN: 1574-1192. DOI: <http://dx.doi.org/10.1016/j.pmcj.2011.09.003>. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1574119211001234>.
- [14] N. Patwari and J. Wilson, "RF sensor networks for device-free localization: Measurements, models, and algorithms," *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1961–1973, 2010, ISSN: 0018-9219. DOI: 10.1109/JPROC.2010.2052010.
- [15] R. Rogers and W. Brown, "Radar observations of a major industrial fire," *Bulletin of the American Meteorological Society*, vol. 78, pp. 803–814, 1997.

- [16] M. C. R. Talampas and K. S. Low, “Integrating active and passive received signal strength-based localization,” in *Intelligent Signal Processing and Communication Systems (ISPACS), 2014 International Symposium on*, 2014, pp. 153–158. DOI: 10.1109/ISPACS.2014.7024443.
- [17] N. Patwari and P. Agrawal, “Effects of correlated shadowing: Connectivity, localization, and rf tomography,” in *Information Processing in Sensor Networks, 2008. IPSN ’08. International Conference on*, 2008, pp. 82–93. DOI: 10.1109/IPSN.2008.7.
- [18] J. Wilson and N. Patwari, “Radio tomographic imaging with wireless networks,” *Mobile Computing, IEEE Transactions on*, vol. 9, no. 5, pp. 621–632, 2010, ISSN: 1536-1233. DOI: 10.1109/TMC.2009.174.
- [19] J. Wilson and N. Patwari, “See-through walls: Motion tracking using variance-based radio tomography networks,” *Mobile Computing, IEEE Transactions on*, vol. 10, no. 5, pp. 612–621, 2011, ISSN: 1536-1233. DOI: 10.1109/TMC.2010.175.
- [20] Y. Zhao and N. Patwari, “Noise reduction for variance-based device-free localization and tracking,” in *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011 8th Annual IEEE Communications Society Conference on*, 2011, pp. 179–187. DOI: 10.1109/SAHCN.2011.5984895.
- [21] Y. Zhao, N. Patwari, J. M. Phillips, and S. Venkatasubramanian, “Radio tomographic imaging and tracking of stationary and moving people via kernel distance,” in *Proceedings of the 12th International Conference on Information Processing in Sensor Networks*, ser. IPSN ’13,

- Philadelphia, Pennsylvania, USA: ACM, 2013, pp. 229–240, ISBN: 978-1-4503-1959-1. DOI: 10.1145/2461381.2461410.
- [22] O. Kaltiokallio, M. Bocca, and N. Patwari, “Enhancing the accuracy of radio tomographic imaging using channel diversity,” in *Mobile Adhoc and Sensor Systems (MASS), 2012 IEEE 9th International Conference on*, 2012, pp. 254–262. DOI: 10.1109/MASS.2012.6502524.
 - [23] O. Kaltiokallio, M. Bocca, and N. Patwari, “Follow @grandma: Long-term device-free localization for residential monitoring,” in *Local Computer Networks Workshops (LCN Workshops), 2012 IEEE 37th Conference on*, 2012, pp. 991–998. DOI: 10.1109/LCNW.2012.6424092.
 - [24] O. Kaltiokallio, M. Bocca, and N. Patwari, “A fade level-based spatial model for radio tomographic imaging,” *Mobile Computing, IEEE Transactions on*, vol. 13, no. 6, pp. 1159–1172, 2014, ISSN: 1536-1233. DOI: 10.1109/TMC.2013.158.
 - [25] M. Bocca, O. Kaltiokallio, and N. Patwari, “Radio tomographic imaging for ambient assisted living,” English, in *Evaluating AAL Systems Through Competitive Benchmarking*, ser. Communications in Computer and Information Science, S. Chessa and S. Knauth, Eds., vol. 362, Springer Berlin Heidelberg, 2013, pp. 108–130, ISBN: 978-3-642-37418-0.
 - [26] D. Zhang, J. Ma, Q. Chen, and L. Ni, “An RF-based system for tracking transceiver-free objects,” in *Pervasive Computing and Communications, 2007. PerCom '07. Fifth Annual IEEE International Conference on*, 2007, pp. 135–144. DOI: 10.1109/PERCOM.2007.8.

- [27] D. Zhang, K. Lu, R. Mao, Y. Feng, Y. Liu, Z. Ming, and L. Ni, “Fine-grained localization for multiple transceiver-free objects by using RF-based technologies,” *Parallel and Distributed Systems, IEEE Transactions on*, vol. 25, no. 6, pp. 1464–1475, 2014, ISSN: 1045-9219. DOI: 10.1109/TPDS.2013.243.
- [28] M. C. R. Talampas and K. S. Low, “Geometry-based algorithms for device-free localization with wireless sensor networks,” in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2014 IEEE Ninth International Conference on*, 2014, pp. 1–6. DOI: 10.1109/ISSNIP.2014.6827625.
- [29] Y. Li, X. Chen, M. Coates, and B. Yang, “Sequential Monte Carlo radio-frequency tomographic tracking,” in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 3976–3979. DOI: 10.1109/ICASSP.2011.5947223.
- [30] X. Chen, A. Edelstein, Y. Li, M. Coates, M. Rabbat, and A. Men, “Sequential Monte Carlo for simultaneous passive device-free tracking and sensor localization using received signal strength measurements,” in *Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on*, 2011, pp. 342–353.
- [31] J. Wilson and N. Patwari, “A fade-level skew-Laplace signal strength model for device-free localization with wireless networks,” *Mobile Computing, IEEE Transactions on*, vol. 11, no. 6, pp. 947–958, 2012, ISSN: 1536-1233. DOI: 10.1109/TMC.2011.102.
- [32] S. Nannuru, Y. Li, Y. Zeng, M. Coates, and B. Yang, “Radio-frequency tomography for passive indoor multitarget tracking,” *Mobile Comput-*

- ing, *IEEE Transactions on*, vol. 12, no. 12, pp. 2322–2333, 2013, ISSN: 1536-1233. DOI: 10.1109/TMC.2012.190.
- [33] J. Wang, Q. Gao, Y. Yu, P. Cheng, L. Wu, and H. Wang, “Robust device-free wireless localization based on differential RSS measurements,” *Industrial Electronics, IEEE Transactions on*, vol. 60, no. 12, pp. 5943–5952, 2013, ISSN: 0278-0046. DOI: 10.1109/TIE.2012.2228145.
 - [34] Y. Guo, K. Huang, N. Jiang, X. Guo, Y. Li, and G. Wang, “An exponential-Rayleigh model for RSS-based device-free localization and tracking,” *Mobile Computing, IEEE Transactions on*, vol. 14, no. 3, pp. 484–494, 2015, ISSN: 1536-1233. DOI: 10.1109/TMC.2014.2329007.
 - [35] Z. Wang, H. Liu, S. Xu, X. Bu, and J. An, “A diffraction measurement model and particle filter tracking method for RSS-based DFL,” *Selected Areas in Communications, IEEE Journal on*, vol. PP, no. 99, pp. 1–1, 2015, ISSN: 0733-8716. DOI: 10.1109/JSAC.2015.2430517.
 - [36] J. Wang, Q. Gao, M. Pan, X. Zhang, Y. Yu, and H. Wang, “Towards accurate device-free wireless localization with a saddle surface model,” *Vehicular Technology, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015, ISSN: 0018-9545. DOI: 10.1109/TVT.2015.2476495.
 - [37] C. Xu, B. Firner, R. Moore, Y. Zhang, W. Trappe, R. Howard, F. Zhang, and N. An, “SCPL : Indoor device-free multi-subject counting and localization using radio signal strength,” in *Information Processing in Sensor Networks (IPSN), 2013 ACM/IEEE International Conference on*, 2013, pp. 79–90. DOI: 10.1109/IPSN.2013.6917577.

- [38] M. Youssef, M. Mah, and A. Agrawala, "Challenges: Device-free passive localization for wireless environments," in *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, ser. MobiCom '07, Montreal, Quebec, Canada: ACM, 2007, pp. 222–229, ISBN: 978-1-59593-681-3. DOI: 10.1145/1287853.1287880.
- [39] M. Moussa and M. Youssef, "Smart devices for smart environments: Device-free passive detection in real environments," in *Pervasive Computing and Communications, 2009. PerCom 2009. IEEE International Conference on*, 2009, pp. 1–6. DOI: 10.1109/PERCOM.2009.4912826.
- [40] M. Seifeldin, A. Saeed, A. Kosba, A. El-Keyi, and M. Youssef, "Nuzzer: A large-scale device-free passive localization system for wireless environments," *Mobile Computing, IEEE Transactions on*, vol. 12, no. 7, pp. 1321–1334, 2013, ISSN: 1536-1233. DOI: 10.1109/TMC.2012.106.
- [41] B. Mager, P. Lundrigan, and N. Patwari, "Fingerprint-based device-free localization performance in changing environments," *Selected Areas in Communications, IEEE Journal on*, vol. PP, no. 99, pp. 1–1, 2015, ISSN: 0733-8716. DOI: 10.1109/JSAC.2015.2430515.
- [42] T. Rappaport, *Wireless Communications: Principles and Practice*, ., Ed. Englewood Cliffs, New Jersey: Prentice-Hall, 1996.
- [43] M. Bocca, O. Kaltiokallio, N. Patwari, and S. Venkatasubramanian, "Multiple target tracking with RF sensor networks," *Mobile Computing, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2013, ISSN: 1536-1233. DOI: 10.1109/TMC.2013.92.
- [44] D. Maas, J. Wilson, and N. Patwari, "Toward a rapidly deployable radio tomographic imaging system for tactical operations," in *Local*

- Computer Networks Workshops (LCN Workshops), 2013 IEEE 38th Conference on*, 2013, pp. 203–210. DOI: 10.1109/LCNW.2013.6758520.
- [45] J. Wang, Q. Gao, P. Cheng, Y. Yu, K. Xin, and H. Wang, “Lightweight robust device-free localization in wireless networks,” *Industrial Electronics, IEEE Transactions on*, vol. 61, no. 10, pp. 5681–5689, 2014, ISSN: 0278-0046. DOI: 10.1109/TIE.2014.2301714.
 - [46] A. Edelstein and M. Rabbat, “Background subtraction for online calibration of baseline RSS in RF sensing networks,” *Mobile Computing, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2012, ISSN: 1536-1233. DOI: 10.1109/TMC.2012.206.
 - [47] J. Wang, Q. Gao, H. Wang, P. Cheng, and K. Xin, “Device-free localization with multidimensional wireless link information,” *Vehicular Technology, IEEE Transactions on*, vol. 64, no. 1, pp. 356–366, 2015, ISSN: 0018-9545. DOI: 10.1109/TVT.2014.2318084.
 - [48] M. C. R. Talampas and K. S. Low, “A geometric filter algorithm for robust device-free localization in wireless networks,” *Industrial Informatics, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015, ISSN: 1551-3203. DOI: 10.1109/TII.2015.2433211.