Localization using Multidimensional Scaling (LMDS)

by

Duke Lee

B.S. (University of California at Berkeley) 1997
M.S. (University of California at Berkeley) 2001

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

in

Engineering - Electrical Engineering and Computer Science

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:
Professor Pravin Varaiya, Chair
Professor Raja Sengupta
Professor Kannan Ramchandran

Spring 2005
The dissertation of Duke Lee is approved:

__________________________   ____________________________
Professor Pravin Varaiya, Chair     Date

__________________________   ____________________________
Professor Raja Sengupta     Date

__________________________   ____________________________
Professor Kannan Ramchandran     Date

University of California, Berkeley

Spring 2005
Localization using Multidimensional Scaling (LMDS)

Copyright © 2005 by Duke Lee
Abstract

Localization using Multidimensional Scaling (LMDS)

by

Duke Lee

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Science
University of California, Berkeley
Professor Pravin Varaiya, Chair

We live in a time of unprecedented wireless connectivity as seen from the widespread use of wireless cellular phones and wireless LAN devices. Today, there is great interest in developing location estimation service in wireless networks such as wireless cellular networks and wireless sensor networks. The location estimation service opens the door to many applications that we call Location Based Services (LBS). The current technology, however, fails to meet the requirements for many applications including wireless 911 services, as highlighted by the failure of all major U.S. service operators to meet the October 1, 2001 Phase I deadline of E911 - a Federal Communications Commission (FCC) wireless 911 mandate. The service operators are currently under great pressure to meet the December 31, 2005 Phase II deadline. Localization is also important in wireless sensor networks, a technology on the rise. The momentum for commercialization of wireless sensor networks was increased recently when Walmart adopted Radio Frequency IDentification (RFID) for its inventory management.

In this thesis, we investigate location estimation. In particular, we study censored distance estimation algorithms and localization algorithms. A distance between a pair of nodes is censored if the distance cannot be reliably measured. We classify existing censored distance estimation algorithms into simple substitution, shortest path methods, and trigonometric resolution methods. Trigonometric $k$-clustering is a multiple trigonometric resolution method that uses geometric contraints to estimate censored distances. The second part of this thesis discusses localization algorithms. In surveying existing localization algorithms, we identify two major computational components: geometrical bounds and refinement. We classify and evaluate existing localization algorithms with respect to these components.
We introduce Localization using Multidimensional Scaling or LMDS, a location estimation algorithm, which takes an expanded set of measurements to increase reliability. The two algorithms introduced are compared with existing algorithms using both experimental and simulative data.

Professor Pravin Varaiya
Dissertation Committee Chair
To my parents and family, for their guidance, support, love, and enthusiasm.
Acknowledgments

First, I would like to thank Professors Pravin Varaiya and Raja Sengupta for their guidance. Without their wisdom, I could not have finished my doctorate program. They showed me by example what it means to be a great mentor, a scientist, and a person. I also would like to thank Professors Ramchandran and Sastry for their valuable input in my research. I would like to thank Tunc Simsek, Rahul Jain, SeongHwe Oh, Chao Chen, Mustafa Ergen, Barish Dundar, and Jeff Ko for their comraderie and countless coffee breaks. I would like to thank Kamin Whitehouse for providing experimental data that was crucial to this research. This research is funded by NSF EAR-0121693.
Contents

Abstract 1
Acknowledgments ii
List of Figures v
List of Tables vii

1 Introduction 1
  1.1 Wireless Networks and Location Estimation 3
    1.1.1 Cellular Networks 3
    1.1.2 Localization in Cellular Networks 4
    1.1.3 Wireless Sensor Networks 6
    1.1.4 Localization in Wireless Sensor Networks 7
  1.2 Dissertation Summary and Contributions 9
    1.2.1 Distance Estimation Algorithms 9
    1.2.2 Localization Algorithms 10
  1.3 System Model and Notation 11

2 Distance Measurements 12
  2.1 Network Connectivity 12
  2.2 Signal Strength 13
  2.3 RF Time of Flight 14
  2.4 Acoustic Time of Flight 14
  2.5 Calibration 15
  2.6 Distance Estimation 16
  2.7 Chapter Summary 16

3 Censored Distance Estimation 18
  3.1 Simple Substitution Method 19
  3.2 Shortest Hop Method 19
  3.3 Shortest Path Method 20
  3.4 Chapter Summary 21

4 Trigonometric Censored Distance Estimation 22
  4.1 Trigonometric Resolution Methods 23
    4.1.1 Random Resolution Method 23
List of Figures

1.1 RFID Tag .......................................................................................................................... 2
1.2 LG Terrestrial DMB Phone ............................................................................................... 3
1.3 Mica Motes .......................................................................................................................... 7

2.1 Signal Strength-Based Ranging .......................................................................................... 13

3.1 Two acceptable configurations specified by (3.1) ............................................................ 18
3.2 Inaccuracies in Simple Substitution .................................................................................. 19
3.3 Preference of Negative Error for Shortest Path Algorithm ............................................. 20

5.1 Node Placements of Signal Strength-Based Ranging Experiments ................................ 29
5.2 Accuracy of Signal Strength Ranging .................................................................................. 29
5.3 Accuracy of Censored Distance Estimation Algorithms - Outside5/1,2,3,4 .................. 31
5.4 Performance of Censored Distance Estimation Algorithms - Outside5/1,2,3,4 .............. 32
5.5 Performance of Censored Distance Estimation Algorithms - Lab2/1,3,4,5 ................... 33
5.6 Accuracy of Acoustic Ranging ........................................................................................... 34
5.7 Node Placements of Acoustic-Based Ranging Experiments ............................................ 35
5.8 Accuracy of Censored Distance Estimation Algorithms - Grasscups/Pavement ............ 37
5.9 Performance of Censored Distance Estimation Algorithms - Grasscups/Pavement ........ 38
5.10 Modeling of Ranging Error: Variance of Ranging Error vs. Distance ......................... 39
5.11 Modeling of Ranging Area: Successful Ranging vs. Distance ....................................... 39
5.12 Node Placements (8x8gvc: 8x8, Varied Spacing, ChipCon CC1000) ......................... 41
5.13 Distance Estimations (8x8gvc: 8x8, Varied Spacing, ChipCon CC1000) ..................... 42
5.14 Node Placements (9-5x9-5g1c: 9-5x9-5, one meter spacing, ChipCon CC1000) .......... 43
5.15 Distance Estimations (9-5x9-5g1c: 9-5x9-5, one meter spacing, ChipCon CC1000) ...... 44
5.16 Node Placements (10x10g3c: 10x10, three meter spacing, ChipCon CC1000) ............. 45
5.17 Distance Estimations (10x10g3c: 10x10, three meter spacing, ChipCon CC1000) ........ 46
5.18 Node Placements (8x8gva: 8x8, Varied Spacing, Acoustic) ........................................... 47
5.19 Distance Estimations (8x8gva: 8x8, Varied Spacing, Acoustic) ..................................... 48
5.20 Node Placements (10x10g1a: 10x10, one meter spacing, Acoustic) ......................... 49
5.21 Distance Estimations (10x10g1a: 10x10, one meter spacing, Acoustic) ...................... 50

6.1 Convex Sets ....................................................................................................................... 52
6.2 Limitation of Convex Bounds .......................................................................................... 52
6.3 Bounding Box Method ..................................................................................................... 53
6.4 Multilateration .................................................................................................................. 55
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>LMDS Overview</td>
<td>58</td>
</tr>
<tr>
<td>7.2</td>
<td>Law of Cosine</td>
<td>59</td>
</tr>
<tr>
<td>7.3</td>
<td>$M + 1$-dimensional placements from Classical MDS for two-dimensional physical space</td>
<td>61</td>
</tr>
<tr>
<td>7.4</td>
<td>Need of Reflection</td>
<td>63</td>
</tr>
<tr>
<td>7.5</td>
<td>Estimation of $pdcor_{ij}$</td>
<td>64</td>
</tr>
<tr>
<td>7.6</td>
<td>Estimation of $\theta(i)$</td>
<td>66</td>
</tr>
<tr>
<td>7.7</td>
<td>Intuition for Implosion</td>
<td>69</td>
</tr>
<tr>
<td>7.8</td>
<td>Implosion from Dimension Reduction (10x10g3c: 10x10, three-meter spacing, ChipCon signal strength-based ranging)</td>
<td>70</td>
</tr>
<tr>
<td>7.9</td>
<td>Correction from Refinement (10x10g3c: 10x10, three-meter spacing, ChipCon signal strength-based ranging)</td>
<td>70</td>
</tr>
<tr>
<td>8.1</td>
<td>Cellular Tower Installations in U.S. Highways</td>
<td>73</td>
</tr>
<tr>
<td>8.2</td>
<td>Position Estimation based on Received Signal Strength Ranging - Westgate (o: true positions, +: estimated positions)</td>
<td>75</td>
</tr>
<tr>
<td>8.3</td>
<td>Position Estimation based on Received Signal Strength Ranging - Lab (o: true positions, +: estimated positions)</td>
<td>76</td>
</tr>
<tr>
<td>8.4</td>
<td>Position Estimation based on the Acoustic Ranging - Grass/Pavement (o: true positions, +: estimated positions)</td>
<td>78</td>
</tr>
<tr>
<td>8.5</td>
<td>Position Estimation based on the Acoustic Ranging - Grass/Grasscups (o: true positions, +: estimated positions)</td>
<td>79</td>
</tr>
<tr>
<td>8.6</td>
<td>Position Estimation based on the Acoustic Ranging - Grasscups/Pavement (o: true positions, +: estimated positions)</td>
<td>80</td>
</tr>
<tr>
<td>8.7</td>
<td>Position Estimation based on the Acoustic Ranging - Grasscups/Grass (o: true positions, +: estimated positions)</td>
<td>81</td>
</tr>
<tr>
<td>8.8</td>
<td>Position Estimations (8x8vc: 8x8, Varied Spacing, ChipCon signal strength-based ranging)</td>
<td>83</td>
</tr>
<tr>
<td>8.9</td>
<td>Position Estimations (9-5x9-5g1c: 9-5x9-5, one-meter spacing, ChipCon signal strength-based ranging)</td>
<td>84</td>
</tr>
<tr>
<td>8.10</td>
<td>Position Estimations (10x10g3c: 10x10, three-meter spacing, ChipCon signal strength-based ranging)</td>
<td>85</td>
</tr>
<tr>
<td>8.11</td>
<td>Position Estimations (8x8va: 8x8, Varied Spacing, the Acoustic Ranging) (o: true positions, +: estimated positions)</td>
<td>87</td>
</tr>
<tr>
<td>8.12</td>
<td>Position Estimations (9-5x9-5g1a: 9-5x9-5, one-meter spacing, the Acoustic Ranging) (o: true positions, +: estimated positions)</td>
<td>88</td>
</tr>
<tr>
<td>8.13</td>
<td>Position Estimations (10x10g1a: 9-5x9-5, one-meter spacing, ChipCon signal strength-based ranging) (o: true positions, +: estimated positions)</td>
<td>89</td>
</tr>
</tbody>
</table>
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Chipcon Radio Signal Strength-Based Ranging Experimental Results ($disErr$ in meters)</td>
<td>30</td>
</tr>
<tr>
<td>5.2</td>
<td>Chipcon Radio Signal Strength-Based Ranging Experimental Results ($posErr$)</td>
<td>30</td>
</tr>
<tr>
<td>5.3</td>
<td>Acoustic-Based Ranging Experimental Results ($disErr$)</td>
<td>36</td>
</tr>
<tr>
<td>5.4</td>
<td>Acoustic-Based Ranging Experimental Results ($posErr$)</td>
<td>36</td>
</tr>
<tr>
<td>5.5</td>
<td>Signal Strength-Based Ranging Simulation Results ($disErr$ in meters)</td>
<td>40</td>
</tr>
<tr>
<td>5.6</td>
<td>Acoustic-Based Ranging Simulation Results ($disErr$)</td>
<td>43</td>
</tr>
<tr>
<td>8.1</td>
<td>Received Signal Strength-Based Ranging Experimental Results ($posErr$)</td>
<td>74</td>
</tr>
<tr>
<td>8.2</td>
<td>Acoustic Time of Flight-Based Ranging Experimental Results ($posErr$)</td>
<td>77</td>
</tr>
<tr>
<td>8.3</td>
<td>Simulation Results Using Received Signal Strength-Based Ranging ($posErr$)</td>
<td>82</td>
</tr>
<tr>
<td>8.4</td>
<td>Simulation Results Using the Acoustic Time of Flight-Based Ranging ($posErr$)</td>
<td>86</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

We live in a world of unprecedented wireless connectivity realized by cellular phones, satellite-based broadcasting, and wireless Local Area Networks (LAN) among others. This explosive growth can be attributed to both technological innovations and tremendous demand. We envision a future of pervasive wireless connectivity and computing in which people are connected to one another and have access to computing power with a simple wireless interface. Among the developments in wireless networks are the spread of cellular networks and the emergence of wireless sensor networks. Throughout this dissertation, concepts and results of location estimation will be explained in the context of these two systems.

The growth of cellular phone services has been truly astounding. Following sluggish growth at its inception during the 1970s, today there are about 1.6 billion wireless users worldwide [24] and the customer base is expected to reach 2.2 billion by 2010 [36]. Today, wireless cellular technology provides access to both voice and data communication. In the near future, increased computing power of the cellular phones and recent developments in location estimation technologies will enable a range of location-based services over cellular networks. Wireless 911 service is one such location-based service that will benefit from improvements in chip-based Global Positioning System (GPS) and ranging technologies. Differential GPS allows for position estimation to come within a 6.7 meters range of the true position 99.73% of the time [27], and a time difference of arrival ranging with respect to cellular base stations is reported to have accuracy of 50 meters in urban areas [43]. In some parts of the world, there is already significant penetration of location-based service-enabled phones. For example, in South Korea,
SK Telecom sold 3.8 million GPS phones while Korea Telecom Freetel (KTF) and LG Telecom supplied 400,000 and 200,000 units of GPS handsets as of November 2004 [37].

There have been exciting developments in wireless sensor networks as well. Recently, Walmart required its top one hundred suppliers to have all their cases and pallets tagged with Radio Frequency Identification (RFID) by January 1, 2005. An RFID tag (Figure 1.1), typically the size of a grain of rice, is a passive device activated by a radio energy pulse. The RFID tag absorbs this radio energy to power up a small chip that transmits an identification number, thus acting as a sophisticated bar code. According to Sanford C. Bernstein & Co., a New York investment research firm, Walmart could save up to $8.4 billion per year in inventory management with RFID [32]. As more companies adopt this technology, more applications for RFID will be available. For example, pharmaceutical companies will be able to ensure that their drugs are not counterfeited and farm products can be tracked to ensure freshness [32].

Moreover, wireless sensor networks that are much more sophisticated than RFID are being developed. There are proposals to deploy wireless sensor networks to monitor spatially distributed physical processes, such as building environments and highway traffic. In these applications, the network comprises of a set of nodes, each consisting of a processor, radio, and one or more sensors. The sensor measurements are locally processed and forwarded to a central server. To understand the received data it is necessary to associate the measurements from each node with its physical location.
1.1 Wireless Networks and Location Estimation

1.1.1 Cellular Networks

There has been tremendous growth in cellular networks owing to technological advances and regulatory reforms. On the technology side, advances in microprocessor fabrication and medium access control technologies have diminished cost, reduced size, and extended battery life of wireless devices. On the regulation side, deregulation and allocation of more bandwidth, such as the Federal Communications Commission (FCC)’s authorization of commercial cellular phone services in 1982 and similar allocation of bandwidth worldwide, have fostered intense global competition.

Essentially, cellular networks provide wireless telephony. The wireless service area is divided into cells and associated base stations. A cellular phone is connected to one or more base stations and is seamlessly handed off to a next set of base stations as it moves across the cells. In each cell, multiple users share the bandwidth using medium access control technologies such as CDMA and TDMA.

In addition to wireless telephony, cellular service providers are looking for ways to expand services. Additional services such as the voice mailbox and text messaging have already become standard. In many parts of the world, the second generation (2G) of medium access control technologies including GSM, CDMA, and TDMA is being replaced by the third generation (3G) including CDMA2000 and WCDMA. The third generation cellular networks support data transmission up to 2.4Mbps as well as voice transmission. For instance, terrestrial digital multimedia broadcasting (DMB) enabled cellular phones like the Samsung SCH-B100 and LG SB-100 (Figure 1.2) to provide sports, weather, and news in streaming video as well as high-speed Internet. Nevertheless, challenges remain. One of these challenges
Section 1.1. Wireless Networks and Location Estimation

is the implementation of a reliable location estimation service, which is essential in many emergency medical care services and security services. In the following section, we discuss localization in cellular networks.

1.1.2 Localization in Cellular Networks

During 2001 more than 139,000 wireless 911 calls were made nationwide [11]. In some regions in the United States twenty to sixty percent of all emergency calls are from wireless phones. Sparked by numerous wireless 911 tragedies [44], the FCC has mandated wireless Enhanced 911 (E911) - a guideline set to improve the effectiveness and reliability of wireless 911 services. In E911, the FCC requires wireless carriers to adopt technologies that would help locate emergency calls. According to the mandate, wireless phone carriers are required to achieve ninety-five percent penetration of Automatic Location Identification (ALI) capable handsets by December 31, 2005. The heterogeneous nature of devices, the diverse terrain of wireless phone networks, and the stringent requirements of the 911 services require both a robust and time efficient localization algorithm.

In the last few years, we have seen various attempts to create a localization system in cellular networks. For instance, now defunct U.S. Wireless developed a system in which the multipath signature of a phone is matched to a database of previously identified patterns and their locations in the Washington D.C. area [40]. A more successful effort is multilateration, which is sometimes referred to as triangulation. In multilateration the location of a mobile node is estimated by measuring its distance to known locations. For instance, Global Positioning System (GPS) estimates the location of a mobile node by measuring the time difference of the arrival of RF signals simultaneously broadcast by GPS satellites. The satellites are synchronized with each other with atomic clocks, and their positions are calculated based on their orbit and the time of RF transmission. While reliable and sufficiently accurate for most outdoor applications, GPS often fails to function indoors and is inaccurate in urban environments. To overcome these shortcomings, a significant portion of multilateration research today focuses on estimating the distance between mobiles and base stations to complement GPS technologies in urban areas. Among competing technologies are Enhanced Observed Time Difference (E-TOD) [5] and Uplink Time Difference of Arrival (U-TDOA) [14]. In U-TDOA, base stations compare the time difference of the arrival of cellular phone transmission. On the other hand, cellular phone in E-TOD keeps track of
Section 1.1. Wireless Networks and Location Estimation

arrival time of simultaneously-broadcasted transmissions from nearby base stations.

Among location estimation design considerations, reliability is one of the most important issues in many safety critical applications such as wireless 911 service. Without an acceptable level of reliability, wireless 911 service may even be harmful since it can give users a false sense of security. FCC’s E911 mandate requires handset-based solutions to achieve fifty meters accuracy for sixty-seven percent of calls and 150 meters accuracy for ninty-five percent of calls. On the other hand, network-based solutions are to achieve 100 meters accuracy for sixty-seven percent of calls and 300 meters accuracy for ninty-five percent of calls.

To accomplish the reliability level specified by FCC’s E911 mandate, U.S. cellular phone carriers are developing robust location estimation service and supplying location-based service (LBS) enabled mobile phones. However, the implementation of location estimation service has proved to be difficult economically and technologically. Not only is upgrading cellular base stations and distributing cellular phone expensive, but reliable distance estimation is difficult as well. The failure of all major wireless cellular operators to meet E911’s Phase I deadline on October 1, 2001 highlights the implementation difficulty. Now, U.S. cellular operators are under even greater pressure to meet E911’s Phase II deadline on December 31, 2005.

Reliable ranging technology that measures distance is one of the biggest challenges for reliable location estimation. As already pointed out, GPS technology measures distance to orbiting satellites. The large distance, which GPS signal needs to travel, weakens the signal significantly, causing the signal to have difficulty penetrating obstructions such as walls. This impenetratability reduces performances of GPS systems indoors and near tall buildings. Technologies that seeks to complement GPS in urban areas are U-TDOA or E-TOD that measure distances to nearby base stations. Using dense deployment of base stations in urban areas, both U-TDOA and E-TOD supporters claim to have achieved the fifty meter accuracy mandated by E911 in wireless 911 services. However, there are still numerous challenges that need to be overcome for improved reliability. The RF transmissions from or to base stations suffer obstructions and multipath. Furthermore, the difference of time-of-arrival of the RF-signals used by U-TDOA and E-TOD are extremely small, leaving the system highly susceptible to error. This is because cellular towers are placed very close to each other compared to the immense distance that light can travel in a split second.

In wireless cellular networks, Localization using Multidimensional Scaling (LMDS) can help
increase robustness against ranging measurement error by incorporating useful measurements neglected by traditional multilateration. Those useful but unused information include proximity/non-proximity information to base stations/mobile phones, location estimation of nearby mobile phones, ranging measurements to nearby mobile phones, and ranging measurements obtained by nearby mobile phones. For example, an indoor mobile user without a reliable location estimation service can approximate its position if he/she can estimate the distance to and get coordinates from a GPS enabled cellular phone nearby. In another instance, a cellular user lacking line-of-sight connection to a base station can improve the distance estimation to the base station by comparing notes with other nearby mobile devices. In LMDS, all available information is forwarded to the central node for global computation.

### 1.1.3 Wireless Sensor Networks

Wired-line sensor networks have been available for decades. Sensors can be found in home and office security systems, monitoring systems for manufacturing processes, and environmental control systems for buildings. Recently, there has been a great impetus to make these devices wireless, exploiting the advantages of wireless networks. First, wireless sensor networks do not need wiring. Installing wires for wired sensor networks in an existing building often are prohibitively expensive. Second, wireless sensor networks can be deployed in dangerous areas such as battle grounds and radioactively contaminated sites. Sensors can be dropped from an airplane or shot as projectiles. Third, wireless sensors can be deployed in places where electrical power is not readily available. Overall, wireless sensor networks can be used in a wider range of applications. For instance, MICRO Seismic Intrusion Detector, or MICROSID - a Vietnam era troop movement vibration detector designed by the CIA - is a wireless sensor deployed in a dangerous place without a source of energy. Camouflaged as a tree twig or a stone, MICROSID transmits a radio burst on a determined frequency at around 140 MHz when triggered.

The surge of interest in wireless sensor networks is shared by both industry and research communities. Advances in MeMS sensors, low-power radios, wireless communication protocols, and tremendous opportunities for new applications are fueling this interest. On the industry side, a milestone was reached in October 2002, when the Zigbee Alliance was formed. The Zigbee Alliance is a wireless communication standard for wireless sensor networks. Currently, more than ninety companies support the technology. Crossbow is one the leading companies committed to wireless sensor development. On
the research side, centers such as Intel Research Lab in Berkeley, Berkeley Wireless Research Center (BWRC), and Center for Embedded Networked Sensing at UCLA are making progress both in hardware and software. Projects like habitat monitoring project at Great Duck Island [26] provided valuable insight for robust construction of sensors and the importance of energy conservation in sensors. Figure 1.3 is Mica Mote used in the habitat monitoring project.

The most ambitious experiment yet is the application of sensor networks in Loch Rannoch, an 885-foot oil tanker [21]. The tanker, one of a fleet operated by British Petroleum (BP), has been outfitted with 160 sensors called 'motes' that measure vibrations in the ship’s pumps, compressors, and engines and forward the readings to a central unit using wireless links. These sensors could potentially help to predict equipment failures. The wireless sensor networks has also been deployed at the Golden Gate Bridge in San Francisco to study structural integrity [34].

### 1.1.4 Localization in Wireless Sensor Networks

For the majority of wireless sensor network applications, sensor readings need to be associated with its location. The motivation of self-actuated localization is threefold. First, localization can reduce the initialization process of sensors, thus lowering cost; without a localization algorithm, an engineer may have to manually record the position of each sensors. Second, depending on the method of deployment, it may not be possible to know where the nodes will end up; as mentioned, sensors can be dropped from an airplane or shot as projectiles. Third, the nodes may be mobile; for instance RFID tags can be tracked from warehouse to warehouse for inventory management.

Location estimation remains a topic of research and discussion as seen from significant literature devoted to localization in wireless sensor networks. Bulusu et al. proposed localization using a grid of landmarks [4]. Niculescu et al. [9] and [10] introduced distributive censored distance estima-
Section 1.1. Wireless Networks and Location Estimation

tion with an information exchange strategy similar to the distant vector routing algorithm. Doherty et al. [19] developed localization based on convex optimization. Simic et al. [31] devised a simple, distributive method for localization by restricting the connectivity area to a rectangular box. Savvides [2], Savarese [6], and Whitehouse [42] among others called for iterative refinement of position estimates using gradient descent algorithms. Nguyen et al. [45] proposed the use learning theory to cope with noisy observations. MIT’s Cricket [23] is an indoor localization system in which each mobile estimates its location by estimating its distance to beacons mounted on walls and ceilings throughout the building.

Diverse applications of wireless sensor networks require diverse information collection and computations strategies in a localization system. Wireless sensor networks can vary from distributed ad hoc wireless sensor networks that monitor environmental information of a habitat to centralized networks in which magnetic sensors on the ground can alert the traffic signal switch for approaching cars. Localization systems of today focus mainly on the ad hoc system. In fact, all location estimation algorithms listed above, except Doherty’s algorithm, are distributed location estimation algorithms designed to address limited computational power, limited battery life, and difficulties of time synchronization of the sensors. However, in the case of centralized architecture, sensory readings are forwarded to a base station with higher computational and storage capacity. Using the routes set up for sensory data transmission, all measurements necessary for location estimation can be forwarded to the base station for global computation. Moreover, centralized solution can improve the accuracy of position estimation by incorporating all distance and position measurements.

LMDS is a centralized algorithm, which computes all unknown locations concurrently using all measurements collected at a central server. LMDS estimates censored distances, obtains initial positional estimates using multidimensional scaling and reference points, and refines the estimate using all measurements. Through experiments and simulations, we have shown that LMDS performs better than multilateration. In particular, LMDS performs significantly better under signal attenuation based ranging when the distance measurements are noisy.
1.2 Dissertation Summary and Contributions

1.2.1 Distance Estimation Algorithms

The first part of this dissertation (Chapter 2 to Chapter 5) deals with distance estimation. Distance estimation is an integral part of localization as it provides necessary geometrical constraints for location estimation. Distance Estimation begins with distance measurement or ranging. A node may measure signal attenuation of a signal to estimate its distance to the transmitter. If the receiver knows at what strength the signal has been transmitted, it can estimate the distance to a transmitter using the received signal strength and a signal attenuation model. Another ranging method is to measure the time-of-flight of an acoustic or radio frequency signal. Given that a receiver-transmitter pair is synchronized in time, the receiver is able to estimate the distance by multiplying the time-of-flight by the speed of the signal.

We also need to be concerned with estimation of distances too large to be measured reliably by available ranging devices, since many ranging devices for localization have very limited range. We call those large distances that cannot be measured reliably censored. In that case, one may be able to estimate the distance indirectly using distance measurements to other neighboring nodes. For example, if there are three nodes $A$, $B$, and $C$, and the distances between $A$ and $B$ and between $B$ and $C$ are not censored but the distance between $A$ and $C$ is censored, one can estimate the latter as the sum of the two previous distances.

Also called multi-hop distance estimation, censored distance estimation algorithms estimate censored distances using distance measurements of neighboring node pairs. Our contribution in the area of censored distance estimation is the development of a classification of censored distance estimations; the expansion of concepts based on trigonometric constraints; and the introduction of a robust, censored distance estimation algorithm, called Trigonometric $k$-clustering (TKC). TKC, examined in detail in Chapter 4, is a censored distance estimation algorithm that estimates the true distance by attempting to form a cluster of estimates around the true value using multiple trigonometric constraints.

Furthermore, we compared the performance of various censored distance estimation algorithms using real signal strength and ultrasonic wave time-of-flight based ranging data from a testbed composed of Berkeley Motes [13]. We also developed a simulation platform that models both ranging error and area. This simulation platform described in Section 5.2 is expected to be useful in future development.
Section 1.2. Dissertation Summary and Contributions

of location estimation in wireless sensor networks. In our experiments and simulations, TKC showed very little bias compared to Shortest Path Algorithm, in which short distances are overestimated and large distances are underestimated.

1.2.2 Localization Algorithms

In the second part of this dissertation (Chapter 6 to Chapter 8) we investigate location estimation algorithms. Our contribution includes organizing location estimation algorithms and introducing LMDS, a robust location estimation algorithm.

Chapter 6 surveys location estimation algorithms, which are conceptually divided into two groups: Geometrical Confinement and Iterative Refinement. Geometrical Confinement algorithms estimate target positions using geometry of connections (distance estimates) and nodes (position measurements). The Iterative Refinement algorithms estimate positions by minimizing error using a gradient search. Many localization algorithms use combinations of the two strategies. We discuss two examples of Geometrical Confinement algorithms: Bounding Box Method and Convex Position Estimation. In Bounding Box Method, node $i$ assumes that all neighboring nodes are within a bounding box around node $i$. Convex Position Estimation is a centralized algorithm that approximates the ranging area with convex shapes such triangles for directional ranging and circles for omni-directional ranging. Iterative Refinement algorithms attempt to find position estimates given an initial estimate by iteratively minimizing the error. Multilateration, also called triangulation, is an Iterative Refinement algorithm. Iterative Multilateration and Multi-hop Multilateration are variations of multilateration, which can overcome the scarcity of reference locations.

LMDS, introduced in Chapter 7, can be divided into three stages: Distance Estimation, Placement, and Coordinate Assignment. In Distance Estimation, LMDS estimates the distances between all involved nodes from position and distance measurements. In Placement, LMDS applies classical MDS to obtain a set of placements according to the distance measurements. The first step in Placement is classical MDS. Using the Law of Cosine, the classical MDS takes the distance estimations and returns $N - 1$ dimensional placements of the nodes, where $N$ is the number of nodes. The second step is to obtain low dimensional placements corresponding to the physical space; if we are interested in locating mobile nodes in a plane, we would want two dimensional placements. LMDS takes the projection by
taking principal components of the placements. In the third stage, Coordinate Assignment, LMDS takes the placements from Placement and changes the coordinate system of the placements according to positional estimates. Specifically, we first find a rigid transformation such that nodes are placed near their positional estimates. Lastly, we run a refinement algorithm that adjusts the placements to match more closely both position and distance estimates.

1.3 System Model and Notation

We classify nodes into anchor nodes and floating nodes. An anchor node can measure its position reliably. A floating node is a non-anchor node. We locate all nodes in the $M$-dimensional physical space, $\mathcal{Y}$. $\mathcal{V}$ is set of all nodes. $N$ is number of nodes ($N = |\mathcal{V}|$) $A$ is set of anchors. The true position of node $i$ is $p(i) \in \mathcal{Y}$; measurement for an anchor node $i$ is $\tilde{p}(i)$; and its measurement error is $e(i) = \tilde{p}(i) - p(i)$.

The true distance between node $i$ and $j$ is $d(i, j)$; its measurement is $\tilde{d}(i, j)$; and its measurement error is $e(i, j) = \tilde{d}(i, j) - d(i, j)$. Neighbors of node $i$, $N(i)$, is the set of all nodes whose distance from $i$ can be reliably measured. The distance between node $i$ and $j$ is censored if the distance cannot be reliably measured. We assume that $\tilde{d}(i, j) = \tilde{d}(j, i)$.

The following summarizes the notation for measurements.

$\tilde{p}(i) = \begin{cases} p(i) + e(i) & \forall i \in A \\ \text{NULL} & \text{otherwise.} \end{cases}$

$\tilde{d}(i, j) = \begin{cases} d(i, j) + e(i, j) & \forall j \in N(i) \\ \text{NULL} & \text{otherwise.} \end{cases}$

$\hat{d}(i, j) = \begin{cases} \| \tilde{p}(i) - \tilde{p}(j) \| & \forall i, j \in A \\ \text{NULL} & \text{otherwise.} \end{cases}$
Chapter 2

Distance Measurements

Ranging technologies are ways of measuring the distance between a pair of nodes. We investigate four popular ranging technologies - network connectivity, radio signal strength, RF time-of-flight, and acoustic time-of-flight; these differ in range, accuracy, directionality, and response to obstacles. We conclude this chapter with a strategy to combine various measurements.

2.1 Network Connectivity

Network connectivity can be used to approximate the distance between a pair of nodes. One way to do this is to model the ranging area. For instance, one can simplify the range of an omni-directional communication device as a circle centered at the device with radius $RANGE$; this yields,

$$d(i, j) \begin{cases} 
\leq RANGE & \text{if } (i, j) \in E \\
> RANGE & \text{otherwise}
\end{cases}$$

In many cases, no extra hardware is necessary for this ranging because the necessary network connectivity information can be readily obtained from onboard radios. However, network connectivity-based ranging do not usually give accurate results. Simplified shapes such as circles do not properly describe a ranging area since the area can be quite irregular in shape as shown in [42]. In addition, the range may be too coarse for reliable distance estimation - an IEEE 802.11 radio, for example, can communicate over one kilometer distance in an open area; in this case, the connectivity information is ineffective for finer-grained localization.
2.2 Signal Strength

A receiver can estimate its distance from a transmitter by measuring the signal attenuation. For instance, if a receiver knows at what strength the signal has been transmitted, it can estimate the distance to the transmitter using received signal strength and a signal attenuation model. In many situations, this method does not require extra hardware since RF-based communication devices are onboard most wireless devices. However, inaccuracies can result from unpredictable power attenuation due to multi-path, interference, fading, and shadowing. This is seen in Figure 2.1, a log-log plot of received signal strength between a pair of Cisco IEEE802.11b wireless cards with respect to the distance between the cards. The measurements were taken from a slow moving vehicle with respect to a fixed transmitter in an open area at the UC Berkeley Richmond Field Station. Furthermore, the direction and altitude of the antenna can affect signal strength significantly. According to [2], raising the antenna 1.5 meters above ground can increase the radio transmission range from 20 to 100 meters.
2.3 RF Time of Flight

Global Positioning System (GPS) [3], a widely available technology for localization, uses RF time-difference-of-arrival from four GPS satellites synchronized with atomic clocks. GPS technology is scalable and has a resolution of 2-3 meters with an error up to 10-20 meters. However, there are three main concerns when using GPS technology. First, each GPS receiver needs line of sight connections to at least four GPS satellites. Thus, localization solutions solely based on GPS readings may not work well indoors and near tall buildings. Second, each receiver needs an accurate clock - an inaccuracy of one microsecond corresponds to a 300-meter error. Third, GPS receivers are still too costly for many wireless applications and consume a great deal of power. A typical GPS receiver costs around 100 dollars (USD) and consumes power on the order of watts [4].

However, with single-chip GPS solutions, such as [35], cost and power consumption are expected to decrease dramatically. Qualcomm’s gpsOne technology is claimed to have an accuracy of 50 meters [25]. Furthermore, if the mobility of sensors is limited, sensors can further reduce the impact of high energy consumption by running localization algorithms sparingly. GPS is a viable position estimation solution in the future for outdoor wireless network applications with moderate accuracy requirement.

In wireless cellular networks, location estimation of a mobile user based on RF time-difference-of-arrival with respect to nearby base stations is proposed to complement GPS technologies indoors. This is to meet the coverage and reliability requirements of applications such as wireless 911 services. Among competing technologies are Enhanced Observed Time Difference (E-TOD) and Uplink Time Difference of Arrival (U-TDOA). In U-TDOA, base stations compare the time difference of the arrival of a cellular phone transmission. On the other hand, a cellular phone in E-TOD keeps track of arrival time of simultaneously broadcast transmissions from nearby base stations. These two technologies showed promising results in their limited deployment. True Position, a Massachusetts based firm, claims to have successfully tested their U-TDOA system with sub 50-meter accuracy[39].

2.4 Acoustic Time of Flight

Ranging technologies using acoustic time-of-flight are utilized in many wireless sensor networks such as MIT’s Cricket [23], ActiveBat [1], UCLA’s AHLoS [2] [20], and UC Berkeley’s Motes [42]. This
technology is robust against fluctuating received signal strength. Savvides [2] observed error less than 2 centimeters up to a range of 3 meters. Data collected by Kamin Whitehouse [16] also showed reliability of the measurement - less than a 10-centimeter error up to the distance of 5 meters in the best case. The error varied somewhat between calibration locations and measurement locations (Section 5.1). However, compared to RF technologies, acoustic time-of-flight technologies are more susceptible to atmospheric conditions, shorter in range, higher in reflection coefficients, and less able to penetrate solid objects such as walls.

Furthermore, acoustic time-of-flight technology needs time synchronization between transceivers. Because a global time synchronization is difficult to achieve, most wireless sensor network systems including [23], [1], [2], and [42] use a RF signal as a time synchronizing signal. A transmitter sends a RF signal and an acoustic signal at the same time; and a receiver measures the time-difference-of-arrival of the two signals. For this to be successful, the algorithm must correctly identify which acoustic signal corresponds to which RF signal.

2.5 Calibration

In many applications in wireless sensor networks, sensors are designed and manufactured economically, resulting in units with varying specifications. For example, an uncalibrated wireless communication radio can transmit twice the power of another radio [17]. In addition, the characteristics of acoustic and RF signals vary significantly depending on the terrain. As noted earlier, raising the antenna 1.5 meters above ground can increase the radio transmission range from 20 to 100 meters.

Furthermore, the cost of each wireless sensor must be kept at a minimum; and in many applications, on-site calibration is difficult due to inaccessibility of the terrain. One approach is to predict the calibration coefficient from simulated settings. Whitehouse [42] advocates calibration of transmission and reception gains for each transceiver by linear regression using all transmission and reception information.
2.6 Distance Estimation

According to the model in Section 1.3, we have up to three measurements for \( d(i, j) \): \( \hat{d}(i, j) \), \( \tilde{d}(i, j) \), and \( \tilde{d}(j, i) \). If the joint probability density function, \( f \) is known, the maximum likelihood estimate \( \delta(i, j) \) of \( d(i, j) \) is

\[
\delta(i, j) = \arg \max_{\hat{d}(i, j)} f(\hat{d}(i, j), \tilde{d}(i, j), \tilde{d}(j, i) | d(i, j)). \tag{2.1}
\]

Suppose that \( e(i) \), \( e(i, j) \), and \( e(j, i) \) are independent Gaussian random variables, i.e., \( e(i) = N(0, \sigma_p) \), \( e(i, j) = N(0, \sigma_{dij}) \), and \( e(j, i) = N(0, \sigma_{dji}) \). Then \( \text{VAR}[\hat{d}(i, j)] \) is \( \text{VAR}[e(i)] + \text{VAR}[e_j] \). Thus, \( \hat{d}(i, j) = N(d(i, j), 2\sigma_p) \). The maximum likelihood estimate for \( d(i, j) \) is \( \delta(i, j) \) such that

\[
\frac{\partial}{\partial \delta(i, j)} \ln f(\hat{d}(i, j), \tilde{d}(i, j), \tilde{d}(j, i) | \delta(i, j)) = 0.
\]

This reduces to

\[
\frac{\hat{d}(i, j) - \delta(i, j)}{2\sigma_p} + \frac{\tilde{d}(i, j) - \delta(i, j)}{\sigma_{dij}} + \frac{\tilde{d}(j, i) - \delta(i, j)}{\sigma_{dji}} = 0,
\]

and we get an explicit expression for \( \delta(i, j) \)

\[
\delta(i, j) = \frac{\sigma_{dij}\sigma_{dji}\hat{d}(i, j) + 2\sigma_p\sigma_{dij}\tilde{d}(i, j) + 2\sigma_p\sigma_{dji}\tilde{d}(j, i)}{\sigma_{dij}\sigma_{dji} + 2\sigma_p\sigma_{dij} + 2\sigma_p\sigma_{dji}}.
\]

2.7 Chapter Summary

In this chapter, we investigated four popular ranging technologies - network connectivity, radio signal strength, RF time-of-flight, and acoustic time-of-flight. For network connectivity and radio signal strength-based algorithms, often no extra hardware is necessary since RF-based communication devices is onboard most wireless devices. However, the ranging methods do not usually give accurate results due to difficulties in the modeling of ranging area and the significant environmental effects on signal propagation.

Time-of-flight based algorithms are more reliable. The best known RF time-of-flight solution is Global Positioning System (GPS). The GPS system estimates locations based on the time-difference-of-arrival of simultaneously broadcasted RF signals by orbiting GPS satellites. Recent development of single chip GPS solutions, with reported accuracy up to 30 meters, is an important step toward its application in wireless cellular networks and wireless sensor networks. Yet, despite the development, GPS is not reliable indoors and in the vicinity of tall buildings. For wireless cellular networks in urban
areas, complementary technologies are being developed that use RF time of difference with respect to base stations.

Acoustic time-of-flight based algorithms have been used in wireless sensor networks. In both Calamari [42] and Cricket [23], RF signal was broadcast simultaneously with the acoustic signal. The RF signal, assumed to arrive at the receiver instantaneously, is used as a time synchronization signal between the receiver-transmitter pair. The acoustic-based algorithm is experimentally shown to be accurate with a resolution of a few centimeters. However, compared to RF signals, acoustic signals tend to be shorter in range and are less able to penetrate obstructions.
A ranging device for localization is useful within a limited range. Distance estimates outside this range are censored. The following example demonstrates the importance of censored distance estimation. Suppose the \((i, j)\)th element of matrix (3.1) is the estimated distance between nodes \(i\) and \(j\), \(1 \leq i, j \leq 4\). The censored distances are marked NULL in the matrix. If we ignore the knowledge that \(\delta(1, 2)\) is censored, it would be impossible to distinguish between the two configurations of Figure 3.1 using the distance estimates (3.1).

\[
\begin{bmatrix}
0 & NULL & \delta(1, 3) & \delta(1, 4) \\
NULL & 0 & \delta(2, 3) & \delta(2, 4) \\
\delta(3, 1) & \delta(3, 2) & 0 & \delta(3, 4) \\
\delta(4, 1) & \delta(4, 2) & \delta(4, 3) & 0
\end{bmatrix}
\]

(3.1)

![Figure 3.1: Two acceptable configurations specified by (3.1)](image)
Section 3.1. Simple Substitution Method

A censored distance between a pair of nodes can be replaced by a value based on prior knowledge about topology and ranging technologies. We call this the Simple Substitution Method. For instance, we can replace censored distances with an average of \( \{d(i,j) | d(i,j) > \text{RANGE}\} \). Suppose \( \text{RANGE} < \delta(1,2) < d(1,2) \). We estimate location of nodes in two dimensional physical space \( (M = 2) \). If all measurements are accurate, node placements that satisfy the censored distance estimate \( \delta(1,2) \) and (3.1) require a three dimensional space as in Figure 3.1. Projecting this three-dimensional placement onto the two-dimensional physical space reduces estimated distances between nodes: In the figure, node 1 is projected closer to node 3 and node 4 than indicated by \( \delta(1,3) \) and \( \delta(1,4) \) respectively. The results can be improved by incorporating more observations from neighboring nodes as we will discuss next.

3.2 Shortest Hop Method

In a network with evenly distributed nodes, the shortest hop count in \((V, E)\) multiplied by an estimated hop length can effectively estimate the distance between nodes. One challenge is to restrict the connectivity to achieve a good resolution while maintaining a set of core neighbors. To see the reason for restricting connectivity, consider a fully connected network. In that network, connectivity information is ineffective in discriminating between nearby nodes from distant nodes. Another challenge is to estimate the average hop length. DV-Hop [9] is a distributed algorithm in which anchors estimate the hop length by averaging hop length to other anchors; and nodes obtain average hop length from
Section 3.3. Shortest Path Method

For dense networks with relatively accurate ranging devices, the shortest path length (along the graph \((V, E)\)) between any two nodes can be used to approximate the distance between them. Because the shortest path length is greater than or equal to the straight line distance between them, there is a tendency for the Shortest Path Method to overestimate the distance. However, this tendency is partly balanced by the algorithm’s tendency toward negative error when minimizing over various paths. To illustrate, the shortest path estimate of \(d(1, 2)\) in Figure 3.3 is

\[
\delta(1, 2) = \begin{cases} 
\delta(1, 3) + \delta(3, 2) & \text{if } \delta(1, 3) + \delta(3, 2) < \delta(1, 4) + \delta(4, 2) \\
\delta(1, 4) + \delta(4, 2) & \text{otherwise.}
\end{cases}
\]

If \(\delta(1, 3), \delta(3, 2), \delta(1, 4), \text{ and } \delta(4, 2)\) are unbiased, the shortest path algorithm tends to underestimate path lengths, because it always chooses the shorter of the two equal-length paths. We observed in our experiments that \(Path\) tends to overestimate short distances due to a curved path but underestimates large distances when the tendency toward negative error dominates.

**Lemma 3.1.** If \(\delta(i, j)\) is an unbiased estimate of \(d(i, j)\) for all \(i, j \in V\), the expected value of the output of the shortest path algorithm is less than or equal to the shortest path length.

We denote \(p_k = [k_1 \ k_2 \ldots \ k_{M_k}]\) to be \(k\)-th path between two fixed nodes. Since \(\min : \mathbb{R}^2 \rightarrow \mathbb{R}\) is a concave function, Jensen's inequality implies \(E[\min_{p_k} (\sum_{t} \delta(k_t, k_{t+1}))] \leq \min_{p_k} (E[\sum_{t} \delta(k_t, k_{t+1})])\).
But since $E[\delta(i, j)] = d(i, j)$, we have $E[\min_p (\sum_l \delta(k_l, k_{l+1}))] \leq \min_p (\sum_l d(k_l, k_{l+1}))$.

3.4 Chapter Summary

Censored distance estimation algorithms estimate “multi-hop” distances using distance measurements of neighboring nodes. In this chapter, we discussed several censored distance estimation algorithms, including Simple Substitution Method, Shortest Hop Method, and Shortest Path Method. Simple Substitution Method replaces censored distance with a predetermined value. It is a simple algorithm, which can be useful with some prior knowledge of the placements. Nevertheless, it is hopelessly inadequate in a large area. Shortest Hop Method and the Shortest Path Method are better algorithms. These methods approximate the distance between two nodes with the shortest path between them. An important advantage of Shortest Hop and Shortest Path is that these algorithms could be implemented as distributed algorithms as a variant of distributed Bellman Ford algorithm. However, both Shortest Hop and Shortest Path suffer from their tendency to underestimate larger distances due to the preference of negative error.
Chapter 4

Trigonometric Censored Distance Estimation

In this chapter, we discuss strategies to estimate censored distances using trigonometric constraints from two adjoining triangles. Although, several concepts in this chapter (e.g. use of trigonometric constraints and clustering) have natural extensions to three dimensional space, we focus on techniques in two-dimensional physical space. A three-dimensional algorithm is a future topic of research.

As shown in Figure 3.1, $\delta(1,3)$, $\delta(1,4)$, $\delta(3,2)$, $\delta(4,2)$, and $\delta(3,4)$ in (3.1) form two adjoining triangles $\Delta(\delta(3,2), \delta(4,2), \delta(3,4))$ and $\Delta(\delta(3,4), \delta(1,3), \delta(1,4))$. Using the Law of Cosine, $\delta(1,2)$ can be identified up to the ambiguity shown in the figure. This trigonometric constraint of two adjoining triangles is the basis for the Trigonometric Censored Distance Estimation. To generalize, we relabel nodes 1, 2, 3, and 4 of Figure 3.1 as nodes $i$, $j$, $k$, and $l$, respectively. Using the new notation, the localization problem is restated: find $\delta(i,j)$ given $l$ and $k$, such that $\delta(i,k)$, $\delta(k,j)$, $\delta(i,l)$, $\delta(l,j)$, and $\delta(k,l)$ are defined. The following equations can be derived using the Law of Cosine.

\[
\theta_{jik} = \cos^{-1}\left(\frac{\delta(k,j)^2 + \delta(k,l)^2 - \delta(l,j)^2}{2\delta(k,j)\delta(k,l)}\right)
\]
\[
\theta_{ilk} = \cos^{-1}\left(\frac{\delta(i,k)^2 + \delta(k,l)^2 - \delta(i,l)^2}{2\delta(i,k)\delta(k,l)}\right)
\]
\[
\theta_{ilk}^\text{min} = \min(|\theta_{ilk} - \theta_{jik}|, |\theta_{ilk} + \theta_{jik}|)
\]
\[
\theta_{ilk}^\text{max} = \max(|\theta_{ilk} - \theta_{jik}|, |\theta_{ilk} + \theta_{jik}|)
\]
These equations produce two random variables, $\delta(i, j)^{\text{min}}$ and $\delta(i, j)^{\text{max}}$, corresponding to the two possible configurations as shown in Figure 3.1. One of the two possible configurations corresponds to the desired estimate. The following algorithms, called Trigonometric Resolution Methods, decide between $\delta(i, j)^{\text{min}}$ and $\delta(i, j)^{\text{max}}$ by attempting to find the true configuration.

### 4.1 Trigonometric Resolution Methods

Trigonometric Resolution Methods resolve the ambiguity between the two configurations shown in Figure 3.1.

#### 4.1.1 Random Resolution Method

The Random Resolution Method chooses between the two configurations at random. This method is simple and does not give preference to $\delta(i, j)^{\text{min}}$ or $\delta(i, j)^{\text{max}}$. In a well-connected network, where multiple pairs of node $l$ and node $k$ that satisfy such configuration can be found for each node pair $i$ and $j$, the Random Resolution Method often produces more accurate results than those of more sophisticated algorithms with a bias toward $\delta(i, j)^{\text{min}}$ or $\delta(i, j)^{\text{max}}$; this is because a tendency to overestimate or underestimate distances can result in significant error when estimating a large distance.

#### 4.1.2 Threshold Resolution Method

The Threshold Resolution Method uses the knowledge of the ranging device’s ability to measure the distance between a pair of nodes as a function of the distance between them. Suppose that $P(d)$ represents the probability that a pair of nodes at distance $d$ from each other can measure the distance between them. Using $P(d)$ we can choose between $\delta(i, j)^{\text{min}}$ and $\delta(i, j)^{\text{max}}$ by

\[
\delta(i, j) = \begin{cases} 
\delta(i, j)^{\text{max}} & \text{if } P(\delta(i, j)^{\text{min}}) > P(\delta(i, j)^{\text{max}}) \\
\delta(i, j)^{\text{min}} & \text{otherwise.}
\end{cases}
\]

In practice, one can expect $P(d)$ to decrease as $d$ increases because it is more difficult to measure the
Section 4.1. Trigonometric Resolution Methods

distance to faraway nodes for most ranging devices. One simple filtering method is to assume that

\[ P(d) = \begin{cases} 
1 & \text{if } d < \text{RANGE} \\
0 & \text{otherwise.} 
\end{cases} \]

Then, \( \delta(i, j) = \delta(i, j)^{\max} \) if \( \delta(i, j)^{\min} < \text{RANGE} \).

4.1.3 Euclidean Method

Euclidean Method of Niculescu et al. in [9] is based on the observation that choosing between the ambiguities is equivalent to choosing between one of two partitions of the plane separated by the line connecting node \( l \) and node \( k \); then, node \( j \) is likely to be in the same partition as node \( i \) if a majority of \( N_j \) is connected to node \( i \). More specifically,

\[ \delta(i, j) = \begin{cases} 
\delta(i, j)^{\max} & \text{if } 2|N_i \cap N_j| < |(N_i \cup N_j) - (N_i \cap N_j)| \\
\delta(i, j)^{\min} & \text{otherwise.} 
\end{cases} \]

We found from our experiments that Euclidean Method is robust when distance measurements are accurate. Euclidean Method slightly outperforms the Random Resolution Method in sparse networks, and slightly under-performs the Random Resolution Method in dense networks.

4.1.4 Degenerate Triangle Method

Suppose \( \Delta(\delta(k, j), \delta(l, j), \delta(k, l)) \) or \( \Delta(\delta(k, l), \delta(i, k), \delta(i, l)) \) is degenerate or nearly degenerate. If \( \Delta(\delta(k, j), \delta(l, j), \delta(k, l)) \) or \( \Delta(\delta(k, l), \delta(i, k), \delta(i, l)) \) approximately form a line, then the triangle is degenerate. From this, we can estimate the distance without ambiguity. If \( \delta(j, k), \delta(j, l), \) and \( \delta(l, k) \) form a line then,

if \( \text{abs}(\delta(l, k) - (\delta(l, j) + \delta(j, k))) \approx 0 \)

\[ \delta(i, j) = \min((\delta(l, j) + \delta(l, i)), (\delta(j, k) + \delta(i, k))) \]

elseif \( \text{abs}(\delta(l, j) - (\delta(l, k) + \delta(j, k))) \approx 0 \)

\[ \theta = \pi - \arccos(((\delta(l, k))^2 + \delta(i, k)^2 - \delta(l, i)^2)/(2 \times \delta(i, k) \times \delta(l, k))) \]
Section 4.2. Multiple Trigonometric Resolution Methods

\[ \delta(i, j) = \sqrt{\delta(i, k)^2 + \delta(j, k)^2} - 2 \cdot \delta(i, k) \cdot \delta(j, k) \cdot \cos(\theta) \]

else

\[ \theta = \pi - \arccos((\delta(i, k)^2 + \delta(i, l)^2 - \delta(l, k)^2)/(2 \cdot \delta(i, k) \cdot \delta(l, k))) \]

\[ \delta(i, j) = \sqrt{\delta(l, i)^2 + \delta(l, j)^2} - 2 \cdot \delta(l, i) \cdot \delta(l, j) \cdot \cos(\theta) \]

4.2 Multiple Trigonometric Resolution Methods

Let \( T(i, j) \) denote a set of all \((l, k)\) for each unknown \( \delta(i, j) \) for which \( \delta(i, k), \delta(k, j), \delta(i, l), \delta(l, j), \) and \( \delta(k, l) \) exist (i.e. are non-NULL). Multiple Trigonometric Resolution Methods estimate \( d(i, j) \) given \{\( \delta(i, k), \delta(k, j), \delta(i, l), \delta(l, j), \) and \( \delta(k, l)\)\( |(l, k) \in T(i, j) \} \).

4.2.1 Averaging

We can average estimates obtained from Trigonometric Resolution Algorithms to obtain \( \delta(i, j) \). Assuming that estimates obtained from Trigonometric Resolution Methods correspond to the true configuration, we hope to obtain a better estimate by averaging. However, there is no guarantee that each estimate is obtained from the true configuration. The next algorithm, Trigonometric \( k \)-clustering, attempts to correct the wrong guesses of the true configuration by a refinement process.

4.2.2 Trigonometric \( k \)-clustering

Trigonometric \( k \)-clustering (TKC) obtains \( \delta(i, j) \) by refining the choices between \( \delta(i, j)_{(i,k)}^{\text{min}} \) and \( \delta(i, j)_{(i,k)}^{\text{max}} \), \( \forall(l, k) \in T(i, j) \), using a variant of the \( k \)-clustering algorithm. Intuitively, we assume that either \( \delta(i, j)_{(i,k)}^{\text{min}} \) or \( \delta(i, j)_{(i,k)}^{\text{max}} \) corresponds to the true configuration. With small estimation errors, each ambiguity is likely to contain an element that is close to the unknown \( d(i, j) \). TKC attempts to identify clusters among the pairs, such that a cluster is formed near \( d(i, j) \). The following is the pseudo code for the variant of the \( k \)-clustering algorithm used by TKC. We start out with an estimate of the distance that may be obtained from any censored distance estimation algorithm.

1. Obtain \( c(i, j, 0) \), estimate of \( d(i, j) \), from any censored distance estimation algorithm. Set \( n = 1 \).
Section 4.3. Chapter Summary

2. We form a cluster $C(i, j, n)$ by selecting one from $\delta(i, j)^{\text{min}}_{(l,k)}$ and $\delta(i, j)^{\text{max}}_{(l,k)}$ for each $(l,k) \in T(i, j)$ that is closer to the current estimate $c(i, j, n - 1)$.

3. If $c(i, j, n) = c(i, j, n - 1)$, exit. Otherwise, continue.

4. Set the current estimate, $c(i, j, n)$, to be the mean of $C(i, j, n)$. Set $n = n + 1$, and go back to Step 2.

This algorithm is guaranteed to converge since (4.1) is monotonically decreasing until the end of the algorithm, and there are only finitely many possible values for (4.1),

$$\sum_{c \in C(i,j,n)} (c - c(i, j, n - 1))^2.$$  \hspace{1cm} (4.1)

4.3 Chapter Summary

Trigonometric Censored Distance Estimation algorithms estimate censored distances using trigonometric constraints from two adjoining triangles. The four Trigonometric Resolution Methods discussed were Random Resolution Method, Threshold Resolution Method, Euclidean Method, and Degenerate Triangle Method. Essentially, the four methods represent different ways of choosing between two configurations that correspond to the same set of trigonometric constraints between two nodes. Random Resolution Method chooses between the choices with equal chance. Threshold Resolution Method eliminates improbable configurations, where a node is within the ranging area of the other. In Euclidean Method, a node chooses between the configuration based on its neighbors’ distance to the other node. Degenerate Triangle Method eliminates configurations that violate the triangle inequality.

We also looked at Multiple Trigonometric Resolution Methods, which estimate censored distance given multiple trigonometric constraints. One way to estimate is to take the average of the results from Trigonometric Resolution Methods. A better method is to use clustering to find the true estimates over all ambiguous configurations. To this end, we developed Trigonometric $k$-clustering, a Multiple Trigonometric Resolution Method, that seeks to find a cluster around the true distance. Trigonometric $k$-clustering forms a cluster by taking one estimate from each ambiguous configuration, using a variant of $k$-clustering algorithm.
Chapter 5

Performance Comparison of Censored Distance Estimation Algorithms

5.1 Experimental Results

We tested various censored distance estimation algorithms with experimental data from wireless sensor networks using both received signal strength-based ranging and acoustic time-of-flight based ranging. All ranging data used below were collected by Kamin Whitehouse and presented in his paper [17] and [16]. The experimental results give us an insight into the performance of ranging devices and behavior of censored distance estimation algorithms.

First, we define $\text{disErr}$ - a performance metric for censored distance estimation algorithms:

$$\text{disErr} = \frac{1}{|V|^2} \sum_{i,j \in V} \frac{\| d(i, j) - \delta(i, j) \|_d}{d(i, j)}.$$  \hfill (5.1)

To justify the use of $\text{disErr}$, we also define $\text{posErr}$ - a performance metric for localization estimation algorithms:

$$\text{posErr} = \frac{1}{|V|} \sum_{i \in V} \frac{\| p(i) - \tilde{p}(i) \|}{R},$$  \hfill (5.2)

where

$$R = \frac{1}{|V|} \sum_{i \in V} \min_{j \in V, i \neq j} d(i, j).$$  \hfill (5.3)

$\text{posErr}$ is the average position estimation error normalized by the average distance to the closest neighbor. We found that $\text{disErr}$ is generally a reliable predictor for $\text{posErr}$ as will be shown in the
Section 5.1. Experimental Results

Using the experimental data, we compare the performance of Shortest Hop (Section 3.2), Shortest Path (Section 3.3), and Trigonometric Resolution Methods (Section 4). We denote Shortest Hop as \( \text{Hop} \), and the Shortest Path Algorithm as \( \text{Path} \). As for the Trigonometric Resolution Methods, we compare two configurations: \( \text{TKCEuclid} \), and \( \text{TKCRand} \). \( \text{TKCEuclid} \) applies the Trigonometric \( k \)-clustering (Section 4.2.2), starting from an initial estimate obtained from Euclidean Method (Section 4.1.3), and \( \text{TKCRand} \) obtains positional estimations using the Trigonometric \( k \)-clustering (Section 4.2.2), starting from the mean of estimates from Random Resolution Method.

5.1.1 Signal Strength-Based Ranging

We discuss a set of experiments carried out in an open, grassy field at the West Gate of the UC Berkeley campus and experiments in an office space at the East wing of the Intel Lab at Berkeley. Information about the experiment can be found in [15]. Berkeley Mica Motes [8] were placed on a 4 x 4 grid with three meter spacing in the field and a 3 x 6 grid with two meter spacing in the lab, as shown in Figure 5.1. They were intentionally placed carelessly to achieve random antenna orientation. Using ChipCon CC1000 radios onboard the Mica Motes, each node took turns transmitting ten messages, and the signal strengths of transmitted signals were collected at each node. The experiments were repeated five times; each time the motes were shuffled around on the grid.

We used four of the five experiments to find a mapping from received signal strength to distance and used the remaining experiment for measurements. More information on the mapping can be found in [17]. Outside5/1,2,3,4 means that outdoor experiments 1, 2, 3, and 4, were used to derive coefficients for distance measurements in experiment 5. Likewise, Lab2/1,3,4,5 means that lab experiments 1, 3, 4, 5, were used for calibration and signal-to-distance mapping for distance measurements in experiment 2. Figure 5.2 is a scatter plot of distance measurements against their true distances. In the figure, points below the true distance 45 degree line are underestimates; and points above the line represent overestimates. The standard deviations of errors are plotted as brackets around the true distance line. As one can see from the graph, the distance measurements from the ChipCon CC1000 radio are not very accurate - the standard deviations of distance error were more than 2 meters as shown in Figure 5.2. In particular, the distance measurements in Lab were too poor to be useful as shown in Figure 5.5. In
Section 5.1. Experimental Results

Figure 5.1: Node Placements of Signal Strength-Based Ranging Experiments

Figure 5.2: Accuracy of Signal Strength Ranging
### Section 5.1. Experimental Results

<table>
<thead>
<tr>
<th>Measured/Trained</th>
<th>Hop</th>
<th>Path</th>
<th>TKCRand</th>
<th>TKCEuclid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside5/1,2,3,4</td>
<td>0.477</td>
<td>0.236</td>
<td>0.229</td>
<td>0.261</td>
</tr>
<tr>
<td>Lab2/1,3,4,5</td>
<td>0.582</td>
<td>0.531</td>
<td>0.512</td>
<td>0.483</td>
</tr>
</tbody>
</table>

Table 5.1: Chipcon Radio Signal Strength-Based Ranging Experimental Results ($disErr$ in meters)

<table>
<thead>
<tr>
<th>Measured/Trained</th>
<th>Hop</th>
<th>Path</th>
<th>TKCRand</th>
<th>TKCEuclid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside5/1,2,3,4</td>
<td>0.255</td>
<td>0.466</td>
<td>0.427</td>
<td>0.477</td>
</tr>
<tr>
<td>Lab2/1,3,4,5</td>
<td>1.094</td>
<td>1.371</td>
<td>1.150</td>
<td>1.352</td>
</tr>
</tbody>
</table>

Table 5.2: Chipcon Radio Signal Strength-Based Ranging Experimental Results ($posErr$)

The figure, placements obtained by LMDS with various censored distance algorithms. For the rest of the analysis, we focus on Outside5/1,2,3,4.

Table 5.1 shows $disErr$ of various distance estimation algorithms in Outside5/1,2,3,4 and Lab2/1,3,4,5. In the table, it is seen that both $TKCRand$ and $Path$ perform well. However, the superiority of trigonometric $k$-clustering is apparent in Figure 5.3, a scatter plot of distance estimations against their true distances. In the figure, $TKCRand$ has the least bias among the distance estimation algorithms as well as low error variance.

Ultimately, we are interested in how various censored distance estimation algorithms affect location estimation. Table 5.2 summarizes $posErr$ obtained from LMDS using four corner nodes at (0,0), (0,9), (9,0), and (9,9) as anchor nodes. With respect to $posErr$ (?), $Path$ and $TKCRand$ performed equally well outdoors. However, $Path$ did poorly indoors compared to $TKCRand$ when the ranging errors were increased. Figure 5.4 shows the placements of nodes in Outside5/1,2,3,4 obtained by LMDS with various censored distance estimation algorithms. In the plot, one can see that the position errors of neighboring nodes are correlated. For instance, in Figure 5.4 (b), all nodes are moved to the right. This is because LMDS identifies a mesh structure of distance estimates around target nodes and locates all nodes in the mesh structure simultaneously. More detailed discussion on LMDS and location estimation in general is deferred to the following chapters.
Section 5.1. Experimental Results

![Graphs of True Distance vs Estimated Distance for different algorithms: (a) Hop, (b) Path, (c) TKCEuclid, (d) TKCRand. The graphs show the accuracy of censored distance estimation algorithms outside 5/1, 2, 3, 4.]

Figure 5.3: Accuracy of Censored Distance Estimation Algorithms - Outside 5/1, 2, 3, 4
Figure 5.4: Performance of Censored Distance Estimation Algorithms - Outside5/1,2,3,4
Figure 5.5: Performance of Censored Distance Estimation Algorithms - Lab2/1,3,4,5
Section 5.1. Experimental Results

Figure 5.6: Accuracy of Acoustic Ranging
Section 5.1. Experimental Results

5.1.2 Ultrasonic-Based Ranging

The ranging data were collected in three different environments: grass, grasscups, and pavement. In the grasscups environment, nodes were placed on top of cups on the grass to increase the ranging area. The node placements, shown in Figure 5.7, were the same for all three experiments. We focus on four configurations based on calibration environment and estimation environment: Grass/Grasscups, Grass/Pavement, Grasscups/Grass, and Grasscups/Pavement. For instance, Grass/Grasscups was a set of distance measurements taken in the grass environment with calibration coefficients obtained from the grasscups environment. Similarly, Grass/Pavement was a set of distance measurements taken in the Grass environment with calibration coefficient obtained from the Pavement environment, and so on. As shown in Figure 5.6, the accuracy of the acoustic-based estimation is significantly better than that of signal strength-based ranging from the ChipCon CC1000 radio.

The improvement in ranging accuracy is reflected in improvement in censored distance estimation. Table 5.3 summarizes the results for time-difference-of-arrival based distance estimation technique using concurrent ultrasonic and RF signals. Both Path and TKCRand perform well in terms of disErr, but TKCRand has slightly less error bias than Path. This is shown in the Figure 5.8, the scatter plot of estimated distances with respect to true distances for Grasscups/Pavement. In the figure, although not as significant as signal strength-based ranging, both Hop and Path algorithms display a tendency
Section 5.2. Simulation Results

<table>
<thead>
<tr>
<th>Measured/Trained</th>
<th>Hop</th>
<th>Path</th>
<th>TKCRand</th>
<th>TKCEuclid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grass/Pavement</td>
<td>0.474</td>
<td>0.035</td>
<td>0.035</td>
<td>0.042</td>
</tr>
<tr>
<td>Grass/Grasscups</td>
<td>0.479</td>
<td>0.043</td>
<td>0.043</td>
<td>0.049</td>
</tr>
<tr>
<td>Grasscups/Pavement</td>
<td>0.816</td>
<td>0.026</td>
<td>0.026</td>
<td>0.026</td>
</tr>
<tr>
<td>Grasscups/Grass</td>
<td>0.799</td>
<td>0.035</td>
<td>0.035</td>
<td>0.035</td>
</tr>
</tbody>
</table>

Table 5.3: Acoustic-Based Ranging Experimental Results ($\text{disErr}$)

<table>
<thead>
<tr>
<th>Measured/Trained</th>
<th>Hop</th>
<th>Path</th>
<th>TKCRand</th>
<th>TKCEuclid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grass/Pavement</td>
<td>0.558</td>
<td>0.299</td>
<td>0.305</td>
<td>0.435</td>
</tr>
<tr>
<td>Grass/Grasscups</td>
<td>0.583</td>
<td>0.328</td>
<td>0.323</td>
<td>0.456</td>
</tr>
<tr>
<td>Grasscups/Pavement</td>
<td>0.297</td>
<td>0.149</td>
<td>0.157</td>
<td>0.157</td>
</tr>
<tr>
<td>Grasscups/Grass</td>
<td>0.305</td>
<td>0.153</td>
<td>0.152</td>
<td>0.152</td>
</tr>
</tbody>
</table>

Table 5.4: Acoustic-Based Ranging Experimental Results ($\text{posErr}$)

to overestimate short distances and underestimate longer distances.

5.2 Simulation Results

We developed our simulation model based on the experimental ranging data discussed in the previous section. For convenience we assumed that additive measurement errors $e(i, j)$ are Gaussian. Using the experimental data, we modeled variance of $e(i, j)$ and success of ranging $p(i, j)$ with respect to the distance. Figure 5.10 (a) shows measurement error variance from Grass/Pavement, Grass/Grasscups, Grasscups/Pavement, and Grasscups/Grass with respect to the distance. Here we fit a square curve to model the sample variance. In the figure, square curves with various scaling constant $\alpha$ are plotted (In our simulation, $\alpha = 0.004$.) Figure 5.10 (b) shows measurement error variance from Outside5/1,2,3,4 and Lab2/1,3,4,5 with respect to the distance. As for the signal strength data, we focused on the outdoor experiment, which was far more reliable than the lab experiment. We fit constant function at 2.44896 to model the variance. As for ranging success probability, we first assumed the ranging device is omni-directional and without clutter. Then, we added a slight randomness at each try to reflect irregularity of ranging area as discussed in Section 2.1.

Figure 5.11 plots the probability of reliable distance measurement as a function of distance. Figure 5.11 (a) gives the results of Grass/Pavement and Grass/Grasscups. We took the results from
Section 5.2. Simulation Results

Figure 5.8: Accuracy of Censored Distance Estimation Algorithms - Grasscups/Pavement
Figure 5.9: Performance of Censored Distance Estimation Algorithms - Grasscups/Pavement
Section 5.2. Simulation Results

![Graph](attachment:image.png)

(a) Acoustic

![Graph](attachment:image.png)

(b) Signal Strength

Figure 5.10: Modeling of Ranging Error: Variance of Ranging Error vs. Distance

![Graph](attachment:image.png)

(a) Acoustic

![Graph](attachment:image.png)

(b) Signal Strength

Figure 5.11: Modeling of Ranging Area: Successful Ranging vs. Distance
Section 5.2. Simulation Results

<table>
<thead>
<tr>
<th></th>
<th>Hop</th>
<th>Path</th>
<th>TKCRand</th>
<th>TKCEuclid</th>
</tr>
</thead>
<tbody>
<tr>
<td>8x8gvc</td>
<td>0.719</td>
<td>0.370</td>
<td>0.266</td>
<td>0.277</td>
</tr>
<tr>
<td>9-5x9-5g1c</td>
<td>0.650</td>
<td>0.372</td>
<td>0.322</td>
<td>0.323</td>
</tr>
<tr>
<td>10x10g3c</td>
<td>0.276</td>
<td>0.336</td>
<td>0.123</td>
<td>0.106</td>
</tr>
</tbody>
</table>

Table 5.5: Signal Strength-Based Ranging Simulation Results (disErr in meters)

Grass/Pavement and approximated the success rate by an exponential function \( exp(-0.35 \times (x - 1.3725)^2) \). For the signal strength-based ranging, as shown in Figure 5.11 (b), the success rate of Outside5/1,2,3,4 is approximated by an exponential function, \( exp(0.0035 \times (x - 1.0607)^2) \). For both cases, \( x \) is distance in meters. Since we do not have data beyond twelve meters, we have censored all distance measurement beyond twelve meters for signal strength-based ranging.

The following are results from various simulations. With simulations we are able to increase the number of nodes and test localization algorithms under a diverse set of placements. In the following analysis, we will find that Trigonometric \( k \)-clustering is a reliable censored distance estimation algorithm. \( TKCEuclid \) tends to perform slightly better than \( TKCRand \) if ranging measurements are accurate or network is sparse. However, when network connectivity is high, \( TKCRand \) seems to outperform all other algorithms. The performance of \( Path \) is not consistent because its tendency to overestimate (path curvatures) and its tendency to underestimate (preference of negative error) need to be balanced. Low network connectivity, uniformly placed nodes, and large ranging error all increase the comparative performance of \( Hop \) against other algorithms. However, the performance of \( Hop \) falls below \( Path \) in most of the cases.

5.2.1 Signal Strength-Based Ranging Simulation

Table 5.5 summarize the results from simulations using the signal strength-based ranging model. The first experiment, 8x8gvc, is executed on an 8x8 grid with varying spacing. Node placements are shown in Figure 5.12. Anchors are located at (0,0), (0,11.2), (11.2,0), and (11.2,11.2). Figure 5.13 shows distance estimates from \( Hop \), \( Path \), \( TKCRand \), and \( TKCEuclid \). In these experiments, all nodes are within two distance measurements away as indicated by Figure 5.13 (a) - only one estimate of distance (two-hop distance) was evaluated. Figure 5.13 (b) shows that \( Path \) significantly underestimates large distances due to noisy distance measurements. \( TKCRand \) most accurately estimates distances.
Section 5.2. Simulation Results

TKCRand performs better than TKCEuclid in this experiment because nodes are well-connected. In such networks many trigonometric constraints are likely to exist for each unknown distance. Taking an average of a large set of randomly chosen configurations can lead to an unbiased and accurate estimate.

The topology for the second experiment, 9-5x9-5g1c, is a 9x9 grid with a 5x5 hole in the middle. Node placements are shown in Figure 5.14. Anchors are located at (0,0), (0,8), (8,0), and (8,8). Figure 5.15 shows the results from selected censored distance estimation algorithms. Again, Path greatly underestimates as shown in Figure 5.15 (b). In this case, the performance of Path is worsened by the uneven placements of nodes. The uneven placements result in a diverse set of neighbors for each node in terms of relatives distances. Therefore, the opposing forces to overestimate and underestimate distances for Path algorithm, as discussed in Section 3.3, work differently for each node. Both TKCEuclid and TKCRand performs decently. The third experiment, 10x10g3c, is on a 10x10 grid with three-meter spacing. Node placements are shown in Figure 5.16. The anchors are located at (0,0), (0,27), (27,0), and (27,27). Figure 5.17 shows results from selected censored distance estimation algorithms. Once again, underestimation of larger distances is shown in Figure 5.17 (b) for Path, exacerbated by increased size of the network. TKCRand performs slightly worse than TKCEuclid since the network is not as well-connected as in the first experiment - less trigonometric constraints are likely to exist for each unknown distance.
Section 5.2. Simulation Results

Figure 5.13: Distance Estimations (8x8gvc: 8x8, Varied Spacing, ChipCon CC1000)
Section 5.2. Simulation Results

Figure 5.14: Node Placements (9-5x9-5g1c: 9-5x9-5, one meter spacing, ChipCon CC1000)

<table>
<thead>
<tr>
<th></th>
<th>Hop</th>
<th>Path</th>
<th>TKCRand</th>
<th>TKCEuclid</th>
</tr>
</thead>
<tbody>
<tr>
<td>8x8gvc</td>
<td>0.719</td>
<td>0.370</td>
<td>0.266</td>
<td>0.277</td>
</tr>
<tr>
<td>10x10g3c</td>
<td>0.276</td>
<td>0.336</td>
<td>0.123</td>
<td>0.106</td>
</tr>
</tbody>
</table>

Table 5.6: Acoustic-Based Ranging Simulation Results (disErr)

5.2.2 Acoustic-Based Ranging Simulation

Table 5.6 summarizes the results from simulations using the acoustic ranging model. In the first experiment (Figure 5.19), 8x8gva, nodes are placed on a 8x8 grid with varied spacing. The node placements are shown in Figure 5.18. The anchors are located at (0,0), (0,5.6), (5.6,0), and (5.6,5.6). In this example, both TKCEuclid and TKCRand perform well due to the superiority of acoustic ranging compared to signal strength-based ranging. Path performs better with acoustic-based ranging than with signal strength-based ranging because the error from its preference of negative error is reduced. However, TKCEuclid and TKCRand both outperform Path, because the uneven density of nodes negatively affects the performance of Path. As explained, the strength of opposing forces to overestimate and underestimate distances for Path algorithm vary in an uneven network.

In the second experiment (Figure 5.21), 10x10g1a, nodes are placed on a 10x10 grid with one meter spacing and with anchors at (0,0), (8,0), (8,8), and (8,8). Node placements are shown in
Section 5.2. Simulation Results

Figure 5.15: Distance Estimations (9-5x9-5g1c: 9-5x9-5, one meter spacing, ChipCon CC1000)
Section 5.3. Chapter Summary

Figure 5.16: Node Placements (10x10g3c: 10x10, three meter spacing, ChipCon CC1000)

Figure 5.20. In this example, $TKCEuclid$ clearly performs the best. Euclidean Method is superior to the Random Resolution Method due to accuracy of measurements. Poorer performance of Path comes from its preference for negative error. As the number of hops increases, error due to the negative bias also increases.

5.3 Chapter Summary

Using experimental data from received signal strength-based ranging and acoustic-based ranging, we compared the performance of censored distance estimation algorithms. We observed that Shortest Path Method has tendencies to both overestimate short distances due to path curvature and underestimate larger distances due to the preference of negative error. The tendency to underestimate large distances was especially detrimental under signal strength-based ranging, since noisy measurements from signal strength-based ranging resulted in highly skewed estimations. Using Random Trigonometric Resolution Method, Trigonometric $k$-clustering had the least bias and performed the best.

Based on the experimental ranging data, we also developed a simulation model. We created three experiments to test the performance of censored distance estimation algorithms under various settings: 8x8gv, 9-5x9-5g, and 10x10g. 8x8gv is an 8x8 grid with varied spacing, 9-5x9-5g is a 9x9 grid
Figure 5.17: Distance Estimations (10x10g3c: 10x10, three meter spacing, ChipCon CC1000)
with a 5x5 hole in the center, and 10x10g is a 10x10 grid. We observed the reliability of Trigonometric $k$-clustering throughout the experiments.
Section 5.3. Chapter Summary

(a) Shortest Hop

(b) Shortest Path

(c) TKC Random

(d) TKC Euclidean

Figure 5.19: Distance Estimations (8x8gva: 8x8, Varied Spacing, Acoustic)
Figure 5.20: Node Placements (10x10g1a: 10x10, one meter spacing, Acoustic)
Section 5.3. Chapter Summary

(a) Shortest Hop

(b) Shortest Path

(c) TKC Random

(d) TKC Euclidean

Figure 5.21: Distance Estimations (10x10g1a: 10x10, one meter spacing, Acoustic)
Chapter 6

Location Estimation Algorithms

6.1 Bounding Method as a Geometrical Confinement Algorithm

Bounding Method relies on the assumption that measurements can be used to “bound” a node to a region. For instance, if node $i$ is 5 meters away from node $j$, then node $i$ is bounded by a circle with 5 meters radius around node $j$. Suppose $B_i(j)$ represents a bound on node $i$ with respect to measurement from node $j$, then

$$ p(i) \in \bigcap_{j \in V} B_i(j). \quad (6.1) $$

In particular, let $R(j)$ denote the ranging area of node $j$. Thus, $R(j)$ is the area in which the distance can be measured reliably from node $j$. Then, we can bound node $i$ by the following expression,

$$ p(i) \in \{ \bigcap_{j \in N(i)} R(j) \} \cap \{ \bigcap_{j \notin N(i)} R(j)^c \}. \quad (6.2) $$

It is often hard to model what the ranging area looks like since the shape of the ranging area is rarely regular as shown in Figure 3 in [42]. To make the the bounding problem computationally feasible, the ranging area $R(j)$ is often approximated by simple shapes. Both Bounding Box Method [31] and Convex Position Estimation [19] make the simplifying assumption that the ranging area is convex. Figure 6.1 shows convex sets used in [31] and [19]. If $C(j)$ is a convex range of node $j$, then

$$ p(i) \in \bigcap_{j \in N(i)} C(j). \quad (6.3) $$
Section 6.1. Bounding Method as a Geometrical Confinement Algorithm

Note that if $C(j)$ is a convex range of node $j$, then $C(j)^c$ cannot be convex. Thus, in this scheme, node $i$ is bounded only by its neighbors; and the information that non-neighboring nodes are likely to be far away cannot be used with this scheme. In Figure 6.2, node $i$ is in the neighborhood of nodes 1 and 2, but not node 3. Using convex constraints only, node $i$ is bounded by a shaded region in Figure 6.2(a) instead of tighter region in Figure 6.2(b). In the following two subsections, we will describe in more detail the two-dimensional ($M = 2$) bounding methods, Bounding Box Method and Convex Position Estimation. In Bounding Box Method, each node bounds its position by ranging area of neighboring nodes approximated by a bounding box. Although the approximation of a bounding box could be improved, the approximations enable each node to compute its position efficiently in a
Section 6.1. Bounding Method as a Geometrical Confinement Algorithm

Figure 6.3: Bounding Box Method

distributed fashion. This is because Bounding Box Method only uses location estimates of its neighbors. Convex Position Estimation, a centralized algorithm developed by Doherty et al, approximates the range with convex shapes such as triangles for directional ranging and circles for omni-directional ranging. In the case of omni-directional ranging, each circular constraints is translated into positive semi-definite programming and solved by a convex optimization algorithm.

6.1.1 Bounding Box Method

Bounding Box Method [31] uses a simple box-shaped ranging area shown in Figure 6.1(a), which can be specified by lower left extreme coordinates, \((x^l, y^l)\), and upper right extreme coordinates, \((x^u, y^u)\). Figure 6.3 shows an example where node \(i\) is bounded by the ranging area of its neighbors, nodes 1, 2, and 3. Thus, the bounding area for the node \(i\) can be specified by \((x^l(i), y^l(i)) = (\max(x^l(1), x^l(2), x^l(3)), \max(y^l(1), y^l(2), y^l(3)))\) and \((x^u(i), y^u(i)) = (\min(x^u(1), x^u(2), x^u(3)), \min(y^u(1), y^u(2), y^u(3)))\). For the node \(i\), the formula for the bounding box is

\[(x^l(i), y^l(i)) = (\max(\{x^l(j)\mid j \in N(i)\}), \max(\{y^l(j)\mid j \in N(i)\}))\]

\[(x^u(i), y^u(i)) = (\min(\{x^u(j)\mid j \in N(i)\}), \min(\{y^u(j)\mid j \in N(i)\}))\).

Bounding Box Method can be implemented as a distributed algorithm; Simic and Sastry [31] suggest anchors broadcast their respective position estimates periodically, and nodes broadcast any changes in their estimates upon reception of any broadcast.
Section 6.2. Multilateration as an Iterative Refinement Algorithm

6.1.2 Convex Position Estimation

Doherty et al. [19] model the directional and omni-directional ranging areas shown in Figure 6.1(b) and (c). Directional ranging technologies (Figure 6.1(b)) include ultrasound and infrared. Omni-directional ranging technologies (Figure 6.1(c)) include radio frequency. Convex Position Estimation [19] finds a feasible point using global convex optimization. Doherty et al. use linear programming (LP) and semi-definite programming (SDP) for computation. Both LP and SDP are studied extensively in the field of convex optimization and can be solved using fast heuristic algorithms [33]. The difference between the two methods is the expressiveness of constraints that form the feasible region. SDP is essentially an LP where the nonnegativity constraint is replaced by a semi-definite constraint on matrix variables. Denoting $\tilde{p}(i) = [\tilde{p}_{i1} \ldots \tilde{p}_{iM}]$ and $x = [\tilde{p}_{11} \ldots \tilde{p}_{1M} \tilde{p}_{21} \ldots \tilde{p}_{2M} \ldots \tilde{p}_{N1} \ldots \tilde{p}_{NM}]$, LP minimizes $c^T x$ subject to

$$Ax \leq b,$$

(6.4)

whereas SDP minimizes $c^T x$ subject to

$$F(x) = F_0 + x_1 F_1 + \ldots + x_{NM} F_{NM} \leq 0,$$

(6.5)

with $F_i = F_i^T$.

In LP, feasible regions are bounded by a set of linear inequalities as shown in (6.4), whereas in SDP, feasible sets are bounded by a linear matrix inequality (LMI) (6.5). The linear constraints of LP can describe the regions such as Figures 6.1(a) and (b). On the other hand, LMI constraints of SDP can describe the circular region shown in (Figure 6.1(c)). The circular constraints of $\|p(i) - p(j)\| \leq r$ can be converted into LMI using the Schur complements [28].

$$\|p(i) - p(j)\| \leq r \rightarrow \begin{bmatrix} I_{2r} & p(i) - p(j) \\ (p(i) - p(j))^T & r \end{bmatrix} \geq 0.$$

(6.6)

6.2 Multilateration as an Iterative Refinement Algorithm

Iterative Refinement Algorithms attempt to find position estimates that minimize a measure of error. We focus our discussion on algorithms that minimize the following class of cost functions:

$$\sum_{i \in V} \sum_{j \in V} \alpha_{ij} \|d(y(i), y(j)) - \delta(i, j)\| + \sum_{i \in A} \beta_i \|y(i) - \tilde{p}(i)\|.$$

(6.7)
Section 6.2. Multilateration as an Iterative Refinement Algorithm

According to the cost function, multilateration solves the following problem:

\[
y(i) = \begin{cases} 
\arg \min_{y(i)} \sum_{j \in N(i), j \in A} \alpha_{ij} \| d(y(i), \tilde{p}(j)) - \delta(i, j) \| & \text{if } i \notin A \\
y(i) = \tilde{p}(i) & \text{if } i \in A
\end{cases} \tag{6.8}
\]

Figure 6.4(a) depicts multilateration for node \(i\) located by its distance estimates \(\{\delta(i, j) | j \in \{1, 2, 3, 4, 5\}\}\) to the anchors with measured positions at \(\{\tilde{p}(j) | j \in \{1, 2, 3, 4, 5\}\}\).

As for the choice of the norm \(\| \cdot \| : \mathcal{R} \rightarrow \mathcal{R}\) in (6.8), most refinement algorithms use the 2-norms (square of the differences). 2-norms is a good choice if the errors are additive, independent, and Gaussian. Then, the maximum likelihood estimate of \(p(i)\) can be found using (6.8). However, the Gaussian assumption on error is disputed. In Resolution of Forces [42], Whitehouse claims that errors are Laplacian and minimizes the sum of the differences (1-norm).

Distributed algorithms exist for the following classes of multilateration techniques: Iterative Multilateration and Multi-Hop Multilateration.

6.2.1 Iterative Multilateration

In its basic form, Multilateration requires dense placements of anchors, because it locates each node by the geometrical constraints imposed by its neighboring anchors. Thus, there must be at least \(M + 1\) neighboring anchors for every target node. Recall that \(M\) was defined as the dimension of the physical space.

To ease the burden of deploying a large number of anchors, we allow multilateration to use
Section 6.3. Chapter Summary

Position estimates of neighboring floating nodes. In this scheme, target nodes without enough neighboring anchors can estimate their location after a sufficient number of neighboring floating nodes estimate their positions. There are several versions of Iterative Multilateration. Collaborative Multilateration [2], [29] extends multilateration to include cases like Figure 6.4(b), in which the location of the two floating nodes $i$ and $j$ are uniquely identified by constraints imposed by the anchors 1, 2, 3, and 4. Hop-TERRAIN/Refinement [7] assigns the scaling value, $\alpha_{ij}$, of (6.8) using estimated reliability of anchor locations.

6.2.2 Multi-Hop Multilateration

Instead of relying on neighboring floating nodes to become “anchors,” a target can directly estimate its distance to non-neighboring anchors. Multi-Hop Multilateration uses “multi-hop” distances to the anchors. Variations of Multi-Hop Multilateration include Ad-Hoc Positioning System, Resolution of Forces, and DV-Hop. Ad-Hoc Positioning System (APS) [9], [10] estimates the distance to non-neighboring anchors by finding the shortest paths to them. APS is a distributed algorithm in that each node maintains its position estimate and shortest path to all known anchors. Furthermore, whenever a broadcast is heard, a node updates its position. Whenever an update results in a change of its location estimate, a node broadcasts the new estimated position and shortest path. Resolution of Forces [42] is a variation of APS and minimizes the sum of the differences of (6.8). DV-Hop [9] approximates the distance between a pair of nodes by multiplying the minimum hop count by the average hop length. In DV-Hop, Anchors estimate the average hop length by taking the average of hop lengths to other anchors, and nodes obtain their hop length estimates from the nearest anchor.

6.3 Chapter Summary

We investigated the problem of localization based on distance and position measurements by identifying two major computational strategies of existing localization algorithms: Geometrical Confinement and Iterative Refinement. Furthermore, we discussed two Geometrical Confinement algorithms: Bounding Box Method and Convex Position Estimation. In Bounding Box Method, each node bounds its position by a ranging area of neighboring nodes approximated by a bounding box. Convex Position Estimation is a centralized algorithm that approximates the ranging area with convex shapes, such as...
triangles for directional ranging and circles for omni-directional ranging.

Iterative Refinement algorithms attempt to find position estimates given an initial estimate by iteratively minimizing error. Multilateration, also called triangulation, is often implemented as a Iterative Refinement algorithm. Iterative Multilateration and Multi-hop Multilateration are variations of multilateration that can overcome scarcity of reference locations.
Chapter 7

Localization using Multidimensional Scaling (LMDS)

LMDS, introduced in this chapter, is a centralized algorithm for two-dimensional physical space ($M = 2$). Conceptually, LMDS can be divided into four stages (Figure 7.1). As mentioned before, the three inputs or observations, $\bar{d}(i, j)$, $\hat{d}(i, j)$, and $\bar{p}(i)$ of Figure 7.1, are distance measurements, distance estimates from position measurements, and position measurements, respectively. The first stage, Distance Estimation, collects and combines position and distance measurements into a distance estimation matrix $(\delta(i, j))$. The second stage, Placement, produces a set of points $\bar{x}(i)$ according to $(\delta(i, j))$. The third stage, Coordinate Assignment, orients $\bar{x}(i)$ by using position measurements of anchors. Finally, the last stage, Refinement improves the position estimate, using a gradient method according to distance $(\delta(i, j))$ and position measurements $\bar{p}$.

Figure 7.1: LMDS Overview
7.1 Multidimensional Scaling

Multidimensional Scaling (MDS) refers to a set of techniques. One of the first well-known MDS techniques was proposed by Torgerson [38] in 1952. Since then MDS has been used in empirical studies in political science, psychology, sociology, anthropology, economics, and education [18]. Now called Classical MDS, Torgerson’s method uses the Law of Cosine to place objects or entities in a Euclidean space such that the distance between the objects correspond to measured distances, or dissimilarity. The placements reveal insight into important attributes of the objects. For example, MDS was used in a study of the 1968 United States presidential election by the Survey Research Center at the University of Michigan [41]. In the study, researchers sought to find perceptual differences among presidential candidates. Each subject was asked to rate twelve potential presidential candidates according to their emotional responses toward the presidential candidates. Dissimilarities among the candidates were computed based on their emotional response. MDS was used to plot the candidates on a plane such that those candidates who were perceived to be similar were placed near one another, and those candidates that were perceived to be dissimilar were placed far away from one another. Analyzing the output, the researchers observed that clusters of candidates were formed based on party affiliation. Other attributes such as the candidates’ view on Vietnam War or social welfare, two predominant issues during the 1968 election, proved less important.

7.1.1 Classical Multidimensional Scaling

Torgerson’s [38] Classical MDS formulation utilizes the Law of Cosine. Let us suppose we have a triangle formed by placing nodes $i$, $j$, and $k$ at each vertex as shown in Figure 7.2. Then, the
angle \( \theta_{jik} \) is constrained by

\[
\cos \theta_{jik} = \frac{\delta(i,j)^2 + \delta(i,k)^2 - \delta(j,k)^2}{2 \delta(i,j) \delta(i,k)}.
\]

Defining \( b_{jik} = \frac{1}{2}(\delta(i,j)^2 + \delta(i,k)^2 - \delta(j,k)^2) \) and placing node \( i \) at the origin, we have

\[
b_{jik} = \delta(i,j) \delta(i,k) \cos \theta_{jik}.
\]

Denoting \( B_i = (b_{jik}) \), we take the square root of \( B_i \) using eigen decomposition; \( B_i = U_i V_i U_i^T = X_N X_N^T \). The \( k \)-th row of \( X_N \), denoted by \( x(k) \), is the \( N \)-dimensional coordinates of node \( k \). Recall that \( N \) is the number of all nodes.

A few theoretical results are available in MDS. In [12], Young and Householder proved that if \( B_i \) is positive semi-definite, then \( d(x(i), x(j)) = \delta(i,j) \). We show in Lemma 7.2 that if \( \delta(i,j) = d(i,j) \), then \( X_N \) has a rank equal or less than \( M \). For the rest of this thesis, we simplify the notation by shifting the points so that the origin is at the centroid of all points. In this technique, called Double Centering Method [38], \( b_{ij} \) is defined as

\[
b_{ij} = \frac{1}{N} \sum_k^n \delta(i,k)^2 + \frac{1}{N} \sum_k^n \delta(j,k)^2 - \delta(i,j)^2 - \frac{1}{N} \sum_k^n \sum_l^n \delta(i,k)^2.
\]

Then we have \( B = (b_{ij}); B = UVU^T = X_N X_N^T; \) and \( X_N = UV^{1/2} \).

### 7.1.2 Dimension Reduction

The Classical MDS places the nodes in a \( N-1 \)-dimensional space according to \((\delta(i,j))\). If there were 100 nodes, then the nodes would be placed in 99-dimensional space! Dimensional Reduction takes the placements into a \( M \)-dimensional physical space \( X_M \) such that the distance relationships of the \( N \)-dimensional placements are preserved as much as possible. More precisely, each \( x(i) \in X_N \) is mapped to \( \bar{x}(i) \in X_M \) such that the \( d(\bar{x}(i), \bar{x}(j)) \) is as close as possible to \( d(x(i), x(j)) \).

To take an intuitive look on how to find such a reduction map, we establish through Lemmas 7.1 and 7.2 that if distance estimation is error-free, then there exists an \( M \)-dimensional subspace, \( X_M \), that contains all points \( x(i) \), and the distance between all pairs of nodes are preserved. In other words, if \( \delta(i,j) = d(i,j) \) for all \( i, j \in V \), then there exists \( M \)-dimensional subspace \( X_M \) of \( X_N \) such that \( x(i) \in X_M \) for all \( i \in V \) and \( d(x(i), x(j)) = d(i,j) \) for all \( i, j \in V \). In this case the dimension reduction map is found by identifying the subspace, \( X_M \). However, in practice, noisy measurements and distance
estimation errors force MDS to push points into “extra” dimensions. Assuming that random errors push placements into random directions, the problem is to find the configuration of points from its noisy image.

Adopting the language of principal component analysis, we find the projection from $x(i)$ to $\bar{x}(i)$ by taking the first $M$ components of $X_N$ obtained from the Classical MDS. Finding the components is simple because $X_N$ is obtained from the eigen decomposition of $B$. Recall that $B = UVU^T = X_NX_N^T$, where $U$ is a set of eigenvectors and $\text{diag}(V \frac{1}{2})$ is a set of eigenvalues. Arranging the elements of $\text{diag}(V \frac{1}{2})$ in non-increasing order, the first $M$ components are the first $M$ columns of the matrix $X_N$.

**Lemma 7.1.** Let $S_i = S(c_i, r_i, M_i)$ denote a sphere in $M_i$-dimensional space centered at $c_i$ with radius $r_i$: $S_i = \{ x = [x_1, x_2 \ldots x_{M_i}] | (x_1 - c_1)^2 + (x_2 - c_2)^2 + \ldots + (x_{M_i} - c_{iM_i})^2 = r_i \}$. Suppose $S_i$ is embedded $M_i$-dimensional subspace, $x_{M_i}$, of a $M$-dimensional Euclidean space, $x_{M}$. If $S_1 \neq S_2$, then $M_{S_1 \cap S_2} = \text{dim}(S_1 \cap S_2) < \text{min}(M_1, M_2)$ and $S_1 \cap S_2$ is a sphere embedded in $x_{M_1} \cap x_{M_2}$.

**Proof:** Because $S_1 \neq S_2$ and $\text{dim}(S_1 \cap S_2) \leq \text{min}(\text{dim}(S_1), \text{dim}(S_2))$, we have $M_{S_1 \cap S_2} < \text{min}(M_1, M_2)$. The rest of the proof shows that $S_1 \cap S_2$ is another sphere.

$S_1 = S(c_1, r_1, M_1)$ can be represented by $S(c_1, r_1, M) \cap x_{M_1}$, where $x_{M_1}$ is a $M_1$-dimensional subspace of $x_M$. $S_1 \cap S_2$ is equivalent to $S(c_1, r_1, M) \cap S(c_2, r_2, M) \cap x_{M_1} \cap x_{M_2}$. Since $x_{M_1} \cap x_{M_2}$ is an another subspace of $x_M$, it is sufficient to show that $S(c_1, r_1, M) \cap S(c_2, r_2, M) = \{ x = [x_1, x_2 \ldots x_M] | (x_1 - c_1)^2 + (x_2 - c_2)^2 + \ldots + (x_{M_1} - c_{1M_1})^2 = r_1^2 \text{ and } (x_1 - c_{21})^2 + (x_2 - c_{22})^2 + \ldots + (x_{M_2} - c_{2M_2})^2 = r_2^2 \}$ is a sphere.

Without loss of generality, we assume that $S_1$ and $S_2$ are centered at $(0, 0, \ldots, 0)$ and $(\|c_2 - c_1\|, 0, \ldots, 0)$, respectively. Then we have $S(c_1, r_1, M) \cap S(c_2, r_2, M) = \{ x = [x_1, x_2 \ldots x_M] | x_1^2 + x_2^2 + \ldots + (x_{M_1} - c_{1M_1})^2 = r_1^2 \text{ and } (x_1 - c_{21})^2 + (x_2 - c_{22})^2 + \ldots + (x_{M_2} - c_{2M_2})^2 = r_2^2 \}$.
... + x_M^2 = r_1^2 and (x_1 - \|c_2 - c_1\|)^2 + x_2^2 + ... + x_M^2 = r_2^2$. Thus, $S(c_1, r_1, M) \cap S(c_2, r_2, M) = \{x = [x_1 \ x_2 ... x_M] | x_1 = \frac{\|c_2 - c_1\|^2 + r_2^2 - r_1^2}{2\|c_2 - c_1\|} \text{ and } x_2^2 + ... + x_M^2 = r_1^2 - \left(\frac{\|c_2 - c_1\|^2 + r_2^2 - r_1^2}{2\|c_2 - c_1\|}\right)^2\}$. This clearly illustrates a sphere.

\[\text{Lemma 7.2.} \text{ If } \delta(i, j) = d(i, j), \forall i, j \in V \text{ where } p(i), \text{ the true location of node } i, \text{ spans } M \text{-dimensional space, } X_N \text{ has rank equal or less than } M.\]

**Proof:** If $X_N$ has rank greater than $M$, there exist $x(0), x(1), ..., x(M)$, and $x(M + 1)$ such that $\bar{x}(1) = x(1) - x(0)$, $\bar{x}(2) = x(2) - x(0)$, ..., and $\bar{x}(M + 1) = x(M + 1) - x(0)$ spans $M + 1$-dimensional subspace of $X_N$. Figure 7.3 shows an example in 2-dimensional physical space. We denote $\tilde{h} = \bar{x}(M + 1) - \bar{x}(M + 1)$, where $\bar{x}(M + 1)$ is a projection of $\bar{x}(M + 1)$ on a plane spanned by $\bar{x}(1)$, $\bar{x}(2)$, ... and $\bar{x}(M)$. Denoting $\bar{x}(M + 1)' = \bar{x}(M + 1) - 2\tilde{h}$, it is easy to see that $d(\bar{x}(M + 1)', 0) = d(\bar{x}(M + 1), 0)$, $d(\bar{x}(M + 1)', x(1)) = d(\bar{x}(M + 1), x(1))$, ..., and $d(\bar{x}(M + 1)', \bar{x}(M)) = d(\bar{x}(M + 1), \bar{x}(M))$.

Since $\delta(i, j) = d(i, j)$, $B$ has to be positive semi-definite. Using Young and Householder's theorem [12], $d(i, j) = d(x(i), x(j))$. Thus, there exists $\bar{x}(M + 1)'$ in the subspace spanned by $\bar{x}(1)$, $\bar{x}(2)$, ..., and $\bar{x}(M)$ and $d(\bar{x}(M + 1)', 0) = d(\bar{x}(M + 1), 0)$, $d(\bar{x}(M + 1)', \bar{x}(1)) = d(\bar{x}(M + 1), \bar{x}(1))$, ..., and $d(\bar{x}(M + 1)', \bar{x}(M)) = d(\bar{x}(M + 1), \bar{x}(M))$. This means that $M$ spheres, with a center at $\bar{x}(i)$ and radius $d(\bar{x}(i), \bar{x}(M + 1))$ embedded in the $M + 1$-dimensional subspace, intersect at no less than three points. However, this is not possible since $M$ intersections of the $M + 1$ spheres can only contain at most two points according to Lemma 7.1. This means that the $M$ intersections of $M + 1$ spheres are a sphere with a dimension less than or equal to one, and an one-dimensional sphere can consist of no more than two points.

### 7.2 Restoration of the Coordinate System

Obtained from the classical multidimensional scaling followed by dimension reduction, $\bar{x}(i)$ is solely based on $(\delta(i, j))$. We reestablish the "coordinate system" by changing the basis of $(\bar{x}(i))$ using the position measurements of anchors $\{\bar{p}(i)\}_{i \in A}$. We call this process the **Restoration of the Coordinate System**: a process of finding a change-of-basis map from $X_M$ to $Y$, such that each anchor


Figure 7.4: Need of Reflection

\( \bar{x}(i), i \in A \) is mapped to the corresponding \( y(i) \), and

\[
\sum_{i \in A} \| y(i) - \tilde{p}(i) \|
\]

is minimized. The following technique makes several simplifying assumptions to reduce computational complexity. Further optimization is left to \textit{Iterative Refinement}.

\textbf{Translation}

First, we find the translation offset between the two coordinate systems by finding a pair of corresponding points: the centroid of all projected anchors, \( \bar{x}^c \), and the centroid of all position measurements of anchors, \( \tilde{p}^c \). We use the following notation to represent translated points:

\[
\bar{x}^c(i) = \bar{x}(i) - \bar{x}^c, \quad \text{where} \quad \bar{x}^c = \frac{1}{|A|} \sum_{i \in A} \bar{x}(i),
\]

\[
\tilde{p}^c(i) = \tilde{p}(i) - \tilde{p}^c, \quad \text{where} \quad \tilde{p}^c = \frac{1}{|A|} \sum_{i \in A} \tilde{p}(i).
\]

For analysis we define \( p^c(i) \), the true position of node \( i \) translated so that the centroid of all anchors are at the origin:

\[
p^c(i) = p(i) - p^c, \quad \text{where} \quad p^c = \frac{1}{|A|} \sum_{i \in A} p(i).
\]

\textbf{Reflection}

First, we explain the process intuitively. For each pair of anchors \( i, j \in A \), there exist two triangles from vertices \( \bar{x}^c(i), \bar{x}^c(j), \text{ and } 0 \); and vertices \( \tilde{p}^c(i), \tilde{p}^c(j), \text{ and } 0 \). Let us suppose an ideal case, where \( \bar{x}^c(i) \) is placed according to \( d(i, j) \) and \( \tilde{p}(i) = p(i) \). Then, \( \bar{x}^c(i) \) needs to be reflected if and only
Section 7.2. Restoration of the Coordinate System

Figure 7.5: Estimation of $pdcor_{ij}$

if the direction of the rotation of vectors $\tilde{x}^c(i)$, $\tilde{x}^c(j) - \tilde{x}^c(i)$, and $-\tilde{x}^c(j)$ does not match the direction of the rotation of vectors $\tilde{y}^c(i)$, $\tilde{y}^c(j) - \tilde{y}^c(i)$, and $-\tilde{y}^c(j)$. This is shown in Figure 7.4. In this case $\tilde{x}^c(i)$ needs to be reflected.

Having found $\tilde{x}^c(i) = [\tilde{x}_{i1}^c \; \tilde{x}_{i2}^c]$ and $\tilde{y}^c(i) = [\tilde{p}_{i1}^c \; \tilde{p}_{i2}^c]$, we use the following formula to find an estimate $\tilde{f}$ of the true reflection $f \in \{+1, -1\}$:

$$\tilde{f}_{ij} = sgn \begin{pmatrix} \tilde{x}_{i1}^c & \tilde{x}_{i2}^c \\ \tilde{x}_{j1}^c & \tilde{x}_{j2}^c \end{pmatrix} \begin{pmatrix} \tilde{p}_{i1}^c & \tilde{p}_{i2}^c \\ \tilde{p}_{j1}^c & \tilde{p}_{j2}^c \end{pmatrix}. \quad (7.1)$$

In (7.1), $\tilde{f}_{ij}$ indicates an agreement between the direction of rotation, defined by the sequence of vectors $\tilde{y}^c(i)$, $\tilde{y}^c(j) - \tilde{y}^c(i)$, and $-\tilde{y}^c(j)$ and the direction of rotation, defined by the sequence of vectors $\tilde{x}^c(i)$, $\tilde{x}^c(j) - \tilde{x}^c(i)$ and $-\tilde{x}^c(j)$. If $pcor_{ij} = P[\tilde{f}_{ij} = f]$, then

$$pcor_{ij} = P \left[ f \ast \text{sgn} \begin{pmatrix} \tilde{x}_{i1}^c & \tilde{x}_{i2}^c \\ \tilde{x}_{j1}^c & \tilde{x}_{j2}^c \end{pmatrix} = \text{sgn} \begin{pmatrix} \tilde{p}_{i1}^c & \tilde{p}_{i2}^c \\ \tilde{p}_{j1}^c & \tilde{p}_{j2}^c \end{pmatrix} \right]. \quad (7.2)$$

Here, $pcor_{ij}$ is the probability that $\tilde{f}_{ij}$ coincides with $f$. We obtain $pcor_{ij}$ in two stages: obtaining $pdcor_{ij}$ by analyzing the reliability of distance estimates and obtaining $ppcor_{ij}$ by analyzing the reliability of position measurements. We first find $pdcor_{ij}$:

$$pdcor_{ij} = P \left[ f \ast \text{sgn} \begin{pmatrix} \tilde{x}_{i1}^c & \tilde{x}_{i2}^c \\ \tilde{x}_{j1}^c & \tilde{x}_{j2}^c \end{pmatrix} = \text{sgn} \begin{pmatrix} \tilde{p}_{i1}^c & \tilde{p}_{i2}^c \\ \tilde{p}_{j1}^c & \tilde{p}_{j2}^c \end{pmatrix} \right]. \quad (7.3)$$

We assume that $\tilde{p}^c_{i1}\tilde{p}^c_{j2} - \tilde{p}^c_{i2}\tilde{p}^c_{j1}$ is an unbiased estimate of $p^c_{i1}p^c_{j2} - p^c_{i2}p^c_{j1}$ with a symmetric probability density function. The assumptions allow us to focus on the case that $f > 0$ and $p^c_{i1}p^c_{j2} -
Section 7.2. Restoration of the Coordinate System

$p_{ij} > 0$. Thus,

$$pdcor_{ij} = P \left[ \begin{bmatrix} \tilde{x}_{i1} & \tilde{x}_{i2} \\ \tilde{x}_{j1} & \tilde{x}_{j2} \end{bmatrix} > 0 \right],$$

$$pdcor_{ij} = P \left[ \tilde{x}_{i1} \tilde{x}_{j2} - \tilde{x}_{j1} \tilde{x}_{i2} > 0 \right]. \quad (7.4)$$

To make (7.4) more intuitive, we note from Figure 7.5 that

$$\text{abs} \left[ \begin{bmatrix} p_{i1}^c & p_{i2}^c \\ p_{j1}^c & p_{j2}^c \end{bmatrix} \right] = h(p^c(i), p^c(i))d(p^c(i), p^c(i)).$$

Denoting $d(\tilde{x}^c(i), \tilde{x}^c(j)) = d(p^c(i), p^c(i)) - e_{ij}$ and $h(\tilde{x}^c(i), \tilde{x}^c(j)) = h(p^c(i), p^c(i)) - \rho_{ij}$, $pdcor_{ij}$ can be rewritten to be

$$pdcor_{ij} = P[h(p^c(i), p^c(i))e_{ij} + d(p^c(i), p^c(i))\rho_{ij} - e_{ij}\rho_{ij} < h(p^c(i), p^c(i))d(p^c(i), p^c(i))].$$

To make computation tractable, we assume that all random variables, $e_{ij}$, $\rho_{ij}$, and $e_{ij}\rho_{ij}$, are additive, independent, and Gaussian. We also assume that $\tilde{p}(i) \approx p(i)$. We then write

$$pdcor_{ij} \approx P[e < h(\tilde{p}(i), \tilde{p}(j))d(\tilde{p}(i), \tilde{p}(j))], \quad (7.5)$$

where $e$ is a zero mean Gaussian random variable with variance $d^2(p^c(i), p^c(i))\sigma^2_{\rho_{ij}} + h^2(p^c(i), p^c(i))\sigma^2_{e_{ij}} + \sigma^2_{p_{ij}} \sigma^2_{e_{ij}}$.

Now we find $ppcor_{ij}$, the second component of $pco_{ij}$:

$$ppcor_{ij} = P \left[ \text{sgn} \begin{bmatrix} p_{i1}^c & p_{i2}^c \\ p_{j1}^c & p_{j2}^c \end{bmatrix} = \text{sgn} \begin{bmatrix} p_{i1}^c + e_{i1} & p_{i2}^c + e_{i2} \\ p_{j1}^c + e_{j1} & p_{j2}^c + e_{j2} \end{bmatrix} \right],$$

$$ppcor_{ij} = P(p_{i1}^c e_{j2} + p_{j2}^c e_{i1} + e_{i1}e_{j2} - p_{i1}^c e_{j2} - p_{j2}^c e_{i1} - e_{i2}e_{j1} < |p_{i1}^c p_{i2}^c - p_{j1}^c p_{j2}^c|).$$

Again, we assume that all random variables, $e_{i1}$, $e_{i2}$, $e_{j1}$, $e_{j2}$, $e_{i1}e_{j2}$ and $e_{i2}e_{j1}$, are additive, independent, and Gaussian, and assume that $\tilde{p}(i) \approx p(i)$. We then write

$$ppcor_{ij} \approx P[e < |p_{i1}^c p_{i2}^c - p_{j1}^c p_{j2}^c|], \quad (7.6)$$
Section 7.2. Restoration of the Coordinate System

Figure 7.6: Estimation of $\theta(i)$

where $e$ is a zero mean Gaussian random variable with variance $\left((\tilde{p}_{i1}^e)^2 + (\tilde{p}_{i2}^e)^2 + (\tilde{p}_{j1}^e)^2 + (\tilde{p}_{j2}^e)^2 + 2\sigma_p^2\right)\sigma_p^2$.

Finally, we write an expression for $\text{pcor}_{ij}$ in terms of $\text{pdcor}_{ij}$ and $\text{ppcor}_{ij}$:

$$\text{pcor}_{ij} = \text{ppcor}_{ij}\text{pdcor}_{ij} + (1 - \text{pdcor}_{ij})(1 - \text{ppcor}_{ij}). \quad (7.7)$$

The maximum likelihood estimate of $\tilde{f}$ given $\{\tilde{f}_{ij} | i, j \in A\}$ is

$$\tilde{f} = \begin{cases} 1 & \text{if } P[\tilde{f}_{ij}, i, j \in A | f = 1] > P[\tilde{f}_{ij}, i, j \in A | f = -1] \\ -1 & \text{otherwise}. \end{cases}$$

Assuming $\{\tilde{f}_{ij} | i, j \in A, i < j\}$ are independent of each other,

$$P[\{\tilde{f}_{ij} | i, j \in A, i < j\} | f] = \prod_{i,j \in A, i < j} \frac{P[\tilde{f}_{ij}, f]}{P[f]},$$

and assuming that $f$ is equally likely to be 1 or -1, the maximum likelihood $\tilde{f}$ of $f$ is

$$\tilde{f} = \begin{cases} 1 & \prod_{i,j \in A, i < j}(\text{pcor}_{ij}1(\tilde{f}_{ij} = 1) + (1 - \text{pcor}_{ij})1(\tilde{f}_{ij} = -1)) > \\ \prod_{i,j \in A, i < j}(\text{pcor}_{ij}1(\tilde{f}_{ij} = -1) + (1 - \text{pcor}_{ij})1(\tilde{f}_{ij} = 1)) \\ -1 & \text{otherwise}. \end{cases}$$

From $\tilde{f}$ we obtain $\bar{x}^e(i)$, reflected $\bar{x}^c(i)$,

$$\bar{x}^e(i) = \begin{cases} (\bar{x}^e_{i1}, \bar{x}^e_{i2}) & \text{if } \tilde{f} = 1 \\ (\bar{x}^e_{i1}, -\bar{x}^e_{i2}) & \text{otherwise}. \end{cases} \quad (7.8)$$

Rotation

$$\tilde{\theta}(i) = \text{sgn} \begin{bmatrix} \bar{x}^e_{i1} & \bar{x}^e_{i2} \\ \tilde{p}^c_{i1} & \tilde{p}^c_{i2} \end{bmatrix} \cos\frac{-1}{2} \frac{||\bar{x}^e(i)||^2 + ||\tilde{p}^c(i)||^2 - ||\bar{x}^e(i) - \tilde{p}^c(i)||^2}{2||\bar{x}^e(i)||||\tilde{p}^c(i)||}. \quad (7.9)$$
Section 7.2. Restoration of the Coordinate System

\( \tilde{\theta}(i) \) (7.9) is an estimate of the rotation \( \theta \) based on the angle between vectors \( p(i) \) and \( \bar{x}^e(i) \).

Figure 7.6 shows that \( \|p^c(i)\| \tilde{\theta}(i) = \varepsilon(i) \), where the arc length \( \varepsilon(i) \) is a zero mean random variable. In the figure, \( \bar{x}^\theta(i) \) is obtained by rotating \( \bar{x}^e(i) \) by the true rotation \( \theta \). We observe the following,

\[
P[\tilde{\theta}(i) < a | \rho(i) < \|p^c(i)\|] = P[\varepsilon(i) < a \|p^c(i)\| | \rho(i) < \|p^c(i)\|].
\]

We assume that \( P[\rho(i) \geq \|p^c(i)\|] \) is small; \( \varepsilon(i) \) is a Gaussian random variable; and \( \bar{p}(i) \approx p(i) \). Then \( \tilde{\theta}(i) \) can be seen as a Gaussian random variable with variance of \( \text{VAR}\[\varepsilon(i)\]/\|\bar{p}(i)\|^2 \). We use the maximum likelihood estimator,

\[
\hat{\theta} = \frac{\sum_i w(i) \tilde{\theta}(i)}{\sum_i w(i)}, \tag{7.10}
\]

where \( w(i) = \prod_{j \neq i} \text{VAR}[\varepsilon(i)]/\|\bar{p}(i)\|^2 \).

Translation

Finally, we assigned the centroid of all nodes, located at the origin, its global location by \( y(i) = \bar{x}^\theta(i) + \bar{p}, \, i \in V \).

Refinement of Change of Basis

The above method of finding translational, reflective, and rotational coefficients in succession is a straightforward way of finding change of basis. Here, we discuss a way to improve on the change of basis. Borrowing ideas from a classical vision problem of pattern matching, we can find \( R \) and \( t \) such that the following quantity is minimized for each anchor node \( i \):

\[
|Ry(i) + t - \bar{p}(i)| \tag{7.11}
\]

or

\[
\left| \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} - \begin{bmatrix} \hat{p}_{i1} \\ \hat{p}_{i2} \end{bmatrix} \right|. \tag{7.12}
\]

Combining the relation for all anchors, we find \( a, b, c, d, t_1, \) and \( t_2 \) that minimize the following.
Section 7.3. Refinement

\[
\begin{bmatrix}
  y_{11} & y_{12} & 0 & 0 & 1 & 0 \\
  0 & 0 & y_{11} & y_{12} & 0 & 1 \\
  y_{21} & y_{22} & 0 & 0 & 1 & 0 \\
  0 & 0 & y_{21} & y_{22} & 0 & 1 \\
  y_{31} & y_{32} & 0 & 0 & 1 & 0 \\
  0 & 0 & y_{31} & y_{32} & 0 & 1 \\
  \cdots
\end{bmatrix}
\begin{bmatrix}
  a \\
  b \\
  c \\
  d \\
  t_1 \\
  t_2 \\
  \cdots
\end{bmatrix}
- 
\begin{bmatrix}
  \tilde{p}_{11} \\
  \tilde{p}_{12} \\
  \tilde{p}_{21} \\
  \tilde{p}_{22} \\
  \tilde{p}_{31} \\
  \tilde{p}_{32} \\
  \cdots
\end{bmatrix}
\right)
\]

(7.13)

\[ R \text{ and } t \text{ can be found using a gradient method starting from our initial estimates } y(i). \] The resulting \( R \) contains information on reflection and rotation coefficients since \( R \) is actually

\[
\begin{bmatrix}
  \cos \theta & -\sin \theta \\
  \sin \theta & \cos \theta
\end{bmatrix}.
\]

Calculating the reflection coefficient is straightforward: \( \text{sgn}(\text{det}(R)) \). However, obtaining the rotation coefficient \( \theta \) is slightly more involved since each entry of \( R, a, b, c, \) and \( d \) can give conflicting \( \theta \). Neglecting reflection, a simple way is to estimate \( \theta \) as the average of \( \cos^{-1}(a), -\sin^{-1}(b), \sin^{-1}(c), \) and \( \cos^{-1}(d) \). Other approaches can be found in computer vision literatures [46].

In our experiments and simulations, we did not see much improvement in location estimation using this gradient method. This is partly because this algorithm minimizes over sine or cosine of \( \theta \), not \( \theta \) itself. Furthermore, the correction we obtained from this method was minimal. Most importantly, this gradient method is eclipsed by the refinement algorithm we will discuss in the next section, incorporating all distance and position measurements.

7.3 Refinement

In this section we discuss ways to improve position estimates \( y(i) \) using both \( \delta(i, j) \) and \( \tilde{p}(i) \). One significant cause for error is a frequent implosion in Dimension Reduction. In Dimension Reduction we project a high dimensional configuration, \( X_N \), on a lower dimensional space. Often, this procedure results in an implosion, where nodes are placed closer to each other than indicated by distance estimates. This case is already shown in Figure 3.1 and is redrawn in this section as Figure 7.3. In the figure, the distance estimation error, \( \delta(1, 2) \), results in Node 1 projected closer to node 3 and node 4 than indicated
Section 7.3. Refinement

Figure 7.7: Intuition for Implosion

by $\delta(1, 3)$ and $\delta(1, 4)$ respectively. We define the implosion coefficient as

$$\frac{\sum d(i, j)}{\sum d(y(i), y(j))}. \tag{7.14}$$

Refinement seeks to address such causes for error. Conceptually, we find $y(i)$ that minimizes

$$(6.7),$$

which is restated below,

$$\sum \sum \alpha_{ij} \| d(y(i), y(j)) - \delta(i, j) \| + \sum \beta_i \| y(i) - ˜p(i) \|.$$

The scaling factors, $\alpha_{ij}$ and $\beta_i$ reflect reliability of various estimates. In our case, position measurements of anchors are assumed to be very small. Thus, in our approach $\beta_i$ dominates, providing a structure for our refinement technique. In the implementation, we improved $y(i)$ in steps. In each step, we move anchors $y(i) \in A$ followed by improvement of $y(i) / \in A$ using Nelder-Mead Method \[22].

Figure 7.8 illustrates LMDS placements without refinement. Nodes are placed in a 10x10 grid with three-meter spacing, and distance measurements are based on signal strength-based ranging. The arrows represent displacements from the true positions (denoted ‘o’) and estimated positions (denoted ‘+’). Peripheral nodes are placed closer to the center with respect to their true positions as indicated by the implosion coefficient of 1.0725.

Figure 7.9 illustrates improvement from refinement. The arrows represent correction from placements without refinement (denoted ‘x’) to placements with refinement (denoted ‘+’). In the figure, implosion at periphery is partially corrected, and nodes are placed closer to their true position. The implosion coefficient of refined placements is 1.0525.
Section 7.3. Refinement

Figure 7.8: Implosion from Dimension Reduction (10x10g3c: 10x10, three-meter spacing, ChipCon signal strength-based ranging) (o: true positions, +: estimated positions without refinement)

Figure 7.9: Correction from Refinement (10x10g3c: 10x10, three-meter spacing, ChipCon signal strength-based ranging) (x: estimated positions without refinement, +: estimated positions with refinement)
In this chapter, we introduced Localization using Multidimensional Scaling (LMDS), a centralized location estimation algorithm. Multidimensional Scaling (MDS) allows us to place nodes in a Euclidean space such that distance between the placed nodes are close the distance estimates. In particular, we used Torgerson’s Classical Multidimensional Scaling, which utilizes Law of Cosine to place objects in $N - 1$-dimensional space in which $N$ is the number of nodes. To obtain placements in $M$-dimensional physical space, we took $M$ principal components. To understand Classical MDS and subsequent Dimension Reduction intuitively, we showed in Lemmas 7.1 and 7.2 that if distance estimation is error-free, then there exists a low dimensional subspace that contains all points $x(i)$ and the distance between all pairs of nodes are preserved. These lemmas are important, because if random errors push placements into random directions, then the reduction map can be seen as finding a low dimensional configuration of points from its noisy image.

The next stage of LMDS called Coordinate Restoration, finds a change of basis so that the placements closely reflect position measurements. We developed a straightforward way of calculating the change of basis in two-dimensional space. Finally, the refinement algorithm makes adjustments to the placements to better reflect distance and position measurements and uncertainty associated with the measurements.
As introduced in Section 5.1, $posErr$ is a performance metric for positional estimation: average position estimation error normalized by average distance to closest neighbor,

$$posErr = \frac{1}{|V|} \sum_{i \in V} \frac{\| p(i) - \tilde{p}(i) \|}{R},$$  \hspace{1cm} (8.1)

where

$$R = \frac{1}{|V|} \sum_{i,j \in V, i \neq j} \min d(i,j).$$  \hspace{1cm} (8.2)

We mention other possibilities. For security systems, one may be interested in reducing the largest of the estimation errors,

$$posErr_{\text{max}} = \max_{i \in V} \| p(i) - \tilde{p}(i) \|.$$  \hspace{1cm} (8.3)

How location estimation error affects “exposure” as introduced by Meguerdichian [30] is also of interest in security systems.

We compare the performance of LMDS (Section 7) and multi-hop multilateration (Section 6.2) using the experimental data, denoted by LMDS and MLAT respectively. MLAT is similar to APS (Section 6.2.2) since MLAT estimates position by minimizing the square of the differences. LMDS uses Trigonometric $k$-clustering (TKC), a centralized censored distance estimation algorithm that is shown to be robust in various environments. MLAT uses Path to estimate multi-hop distance to anchors.
All experiments and simulations were tested under two anchor configurations: AnchorsIn and AnchorsOut. In the AnchorsIn option, four inner points were selected as the anchors. In the AnchorsOut option, four nodes at the outermost corners were chosen to be the anchors. AnchorsIn represents cases where anchors are located in close proximity or when the spatial diversity of anchors is low. We define spatial diversity coefficient of target node $i$ as

$$CH \parallel p(i) - p^c \parallel,$$

where $CH$ is the area of the convex hall of anchors and $p^c$ is centroid of all anchor nodes.

For instance, Figure 8.1 illustrates a typical case of rural cellular networks in the United States where base stations 1, 2 and 3 are located along highways. In the figure, the position of cellular user $i$ is being triangulated from position measurements of base stations, $\tilde{p}(1)$, $\tilde{p}(2)$ and $\tilde{p}(3)$, and its distance to the stations, $\tilde{d}(1, i)$, $\tilde{d}(2, i)$, and $\tilde{d}(3, i)$. The figure is an example where MLAT is vulnerable to a small measurement error due to insufficient distance between anchors. In the figure, a small error, $e$, added to $d(3, i)$ causes multilateration to determine node $i$ to be at $p'(i)$, the opposite side of the highway from the true position $p(i)$.

The following section summarizes results from the experiments. We will see that both LMDS and MLAT perform well when the distance measurements are accurate. However, when the distance measurements is not accurate, LMDS significantly outperforms MLAT.

Figure 8.1: Cellular Tower Installations in U.S. Highways
### Experimental Results

In this section, we discuss results from experiments carried out in wireless sensor networks. We used both received signal strength-based ranging and acoustic time-of-flight based ranging. Each experiment was repeated twice with the AnchorsIn and the AnchorsOut options. With the AnchorsIn option, four inner points were selected to be anchors. On the other hand, with the AnchorsOut option, four nodes at the outermost corners were chosen to be anchors. More detail on the experimental setup can be found in Section 5.1. Again, we would like to thank Kamin Whitehouse for making available the ranging data, which is presented in his paper [16].

#### 8.1.1 Received Signal Strength-Based Ranging

Table 8.1 summarizes the results for the received signal strength-based position estimation techniques using ChipCon radios for both outdoors (Westgate5/1,2,3,4) and indoors (Lab2/1,3,4,5). In Westgate5/1,2,3,4 anchor nodes were placed at (3,3), (3,6), (6,3), and (6,6) with the AnchorsIn option, and at (0,0), (0,9), (9,0), and (9,9) with the AnchorsOut option. In Lab2/1,3,4,5 anchors were placed at (0,4), (2,6) and (4,4) with the AnchorsIn option, and at (0,0), (0,8), (4,0), and (4,8) with AnchorsOut option.

*LMDS* performed better than *MLAT* outdoors. As indicated by lower posErr in Table 8.1, *LMDS* outperformed *MLAT* in both the AnchorsIn and AnchorsOut options. Figure 8.2 shows the position estimates from outdoor experiments, Outside5/1,2,3,4. The estimates from *MLAT* with the AnchorsIn option as shown in Figure 8.2 (b) were very inaccurate. For instance, the node at (6,9) was estimated to be near (-2,8).

On the other hand, *LMDS* and *MLAT* both fail to produce good results indoors as shown by Lab2/1,3,4,5 results in Table 8.1. Figure 8.3 shows results from the indoor experiment, Lab2/1,3,4,5.

<table>
<thead>
<tr>
<th>Measured/Trained-AnchorOption</th>
<th>LMDS</th>
<th>MLAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Westgate5/1,2,3,4-AnchorsIn</td>
<td>0.436</td>
<td>0.806</td>
</tr>
<tr>
<td>Westgate5/1,2,3,4-AnchorsOut</td>
<td>0.427</td>
<td>0.384</td>
</tr>
<tr>
<td>Lab2/1,3,4,5-AnchorsIn</td>
<td>1.628</td>
<td>2.084</td>
</tr>
<tr>
<td>Lab2/1,3,4,5-AnchorsOut</td>
<td>1.150</td>
<td>1.110</td>
</tr>
</tbody>
</table>

Table 8.1: Received Signal Strength-Based Ranging Experimental Results (posErr)
Figure 8.2: Position Estimation based on Received Signal Strength Ranging - Westgate (o: true positions, +: estimated positions)
Figure 8.3: Position Estimation based on Received Signal Strength Ranging - Lab (o: true positions, +: estimated positions)
### Section 8.2. Simulation Results

<table>
<thead>
<tr>
<th>Measured/Trained-AnchorOption</th>
<th>LMDS</th>
<th>MLAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grass/Pavement-AnchorIn</td>
<td>0.330</td>
<td>0.286</td>
</tr>
<tr>
<td>Grass/Pavement-AnchorOut</td>
<td>0.305</td>
<td>0.226</td>
</tr>
<tr>
<td>Grass/Grasscups-AnchorIn</td>
<td>0.319</td>
<td>0.330</td>
</tr>
<tr>
<td>Grass/Grasscups-AnchorOut</td>
<td>0.323</td>
<td>0.279</td>
</tr>
<tr>
<td>Grasscups/Pavement-AnchorIn</td>
<td>0.166</td>
<td>0.157</td>
</tr>
<tr>
<td>Grasscups/Pavement-AnchorOut</td>
<td>0.157</td>
<td>0.164</td>
</tr>
<tr>
<td>Grasscups/Grass-AnchorIn</td>
<td>0.163</td>
<td>0.181</td>
</tr>
<tr>
<td>Grasscups/Grass-AnchorOut</td>
<td>0.152</td>
<td>0.234</td>
</tr>
</tbody>
</table>

Table 8.2: Acoustic Time of Flight-Based Ranging Experimental Results \((\text{posErr})\)

In the figure, one can see many nodes displaced more than 5 meters - larger than the grid spacing of 2 meters. Both LMDS and MLAT do worse than if we placed all floating nodes at \((2, 4)\), at the center.

#### 8.1.2 Acoustic Time of Flight-Based Ranging

In all four environments, Grass/Pavement, Grass/Grasscups, Grasscups/Pavement, and Grasscups/Grass, the node placements were the same. Anchors were placed at \((3.08, 2.88)\), \((0.45, 0.11)\), \((4.24, 3.95)\), and \((4.08, 1.22)\) with the AnchorsOut option, and at \((3.2394, 3.6229)\), \((1.5252, 3.6590)\), and \((2.0738, 1.2479)\) with the AnchorsIn option. As explained in Section 5, the accuracy of the acoustic time of flight-based distance estimation is significantly better than that of the received signal strength-based ranging from the ChipCon CC1000 radio. Table 8.2 summarizes these results. Again, LMDS outperformed MLAT.

Figures 8.4, 8.5, 8.6, and 8.7 shows the position estimates from the four environments Grass/Pavement, Grass/Grasscups, Grasscups/Pavement, and Grasscups/Grass, respectively. Even though all tests were performed outdoors using the same node placements, differences of calibration environment and measurement environments made an impact on their performance. Grasscups/Grass gave the best performance as seen in Figure 8.7.

#### 8.2 Simulation Results

Our simulation model is based on the experimental results, as explained in Section 5.2. The following simulations allow us to test localization algorithms under a diverse set of placements with an
Section 8.2. Simulation Results

Figure 8.4: Position Estimation based on the Acoustic Ranging - Grass/Pavement (o: true positions, +: estimated positions)
Section 8.2. Simulation Results

Figure 8.5: Position Estimation based on the Acoustic Ranging - Grass/Grasscups (o: true positions, +: estimated positions)
Section 8.2. Simulation Results

Figure 8.6: Position Estimation based on the Acoustic Ranging - Grasscups/Pavement (o: true positions, +: estimated positions)
Figure 8.7: Position Estimation based on the Acoustic Ranging - Grasscups/Grass (o: true positions, +: estimated positions)
8.2. Simulation Results

<table>
<thead>
<tr>
<th>Experiment-AnchorOption</th>
<th>LMDS</th>
<th>MLAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>8x8gvc-AnchorsIn</td>
<td>0.675</td>
<td>4.491</td>
</tr>
<tr>
<td>8x8gvc-AnchorsOut</td>
<td>0.562</td>
<td>4.589</td>
</tr>
<tr>
<td>9-5x9-5g1c-AnchorsIn</td>
<td>0.474</td>
<td>1.858</td>
</tr>
<tr>
<td>9-5x9-5g1c-AnchorsOut</td>
<td>0.533</td>
<td>2.462</td>
</tr>
<tr>
<td>10x10g3c-AnchorsIn</td>
<td>0.369</td>
<td>3.551</td>
</tr>
<tr>
<td>10x10g3c-AnchorsOut</td>
<td>0.366</td>
<td>2.287</td>
</tr>
</tbody>
</table>

Table 8.3: Simulation Results Using Received Signal Strength-Based Ranging ($posErr$)

expanded set of nodes.

8.2.1 Received Signal Strength-Based Ranging Simulation

Using the received signal strength-based ranging model, we performed position estimation experiments using both LMDS and MLAT under three scenarios: 8x8vc, 9-5x9-5g1c, and 10x10g3c. In 8x8vc nodes were placed in a 8x8 grid with varied spacing as shown in Figure 8.8. With the AnchorsIn option, anchors were placed at (2.4, 2.4), (2.4, 4), and (4, 2.4), and (4, 4). With the AnchorsOut option, anchors were placed at (0, 0), (0, 11.2), (11.2, 0), and (11.2, 11.2). In 9-5x9-5g1c, nodes were placed in a 9x9 grid with one-meter spacing with a 5x5 hole in the center (Figure 8.9). With the AnchorsIn option, anchors were placed at (2, 2), (2, 7), (7, 2), and (7, 7). With the AnchorsOut option, anchors were placed at (0, 0), (0, 9), (9, 0), and (9, 9). In 10x10g3c nodes were placed in a 10x10 grid with three-meter spacing (Figure 8.10). With the AnchorsOut option, anchors were placed at (0, 0), (0, 27), (27, 0), and (27, 27). With the AnchorsIn option, anchors were placed at (12, 12), (15, 12), (12, 15), and (15, 15).

LMDS outperformed MLAT in all experiments. Surprisingly, LMDS was able to find reasonably accurate position estimates even when the measurements were inaccurate, as illustrated in Figure 8.8, 8.9, and 8.10. The reason for this is that LMDS is less vulnerable to any individual error since LMDS uses the measurements between all nodes instead of estimates to anchors only. LMDS creates a large mesh structure using all measurements and orients the mesh structure with respect to anchor measurements. In locating a node in the mesh structure, the effect of distance estimation error can be offset by other distance estimates in the mesh structure.

The importance of spatial diversity of anchors is displayed in Figure 8.13 (d). The type of error
Figure 8.8: Position Estimations (8x8vc: 8x8, Varied Spacing, ChipCon signal strength-based ranging) (o: true positions, +: estimated positions)
Figure 8.9: Position Estimations (9-5x9-5g1c: 9-5x9-5, one-meter spacing, ChipCon signal strength-based ranging) (o: true positions, +: estimated positions)
Section 8.2. Simulation Results

Figure 8.10: Position Estimations (10x10g3c: 10x10, three-meter spacing, ChipCon signal strength-based ranging) (o: true positions, +: estimated positions)
### Section 8.2. Simulation Results

<table>
<thead>
<tr>
<th>Experiment-AnchorOption</th>
<th>LMDS</th>
<th>MLAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>8x8gv5.6a-AnchorsIn</td>
<td>0.240</td>
<td>0.645</td>
</tr>
<tr>
<td>8x8gv5.6a-AnchorsOut</td>
<td>0.319</td>
<td>0.596</td>
</tr>
<tr>
<td>9-5x9-5g1a-AnchorsIn</td>
<td>0.156</td>
<td>0.270</td>
</tr>
<tr>
<td>9-5x9-5g1a-AnchorsOut</td>
<td>0.158</td>
<td>0.218</td>
</tr>
<tr>
<td>10x10g1a-AnchorsIn</td>
<td>0.162</td>
<td>0.437</td>
</tr>
<tr>
<td>10x10g1a-AnchorsOut</td>
<td>0.118</td>
<td>0.277</td>
</tr>
</tbody>
</table>

Table 8.4: Simulation Results Using the Acoustic Time of Flight-Based Ranging ($posErr$)

that can results from lack of spatial diversity in U.S. rural freeways (Figure 8.1) is shown in Figure 8.10 (d) among many edge nodes. For instance, the node at (0,15) is located near (24,15), and the node at (9,0) is located near (9,21), on the opposite side of the area. Again, LMDS is more robust against lack of the spatial diversity since it is more tolerant to any single measurement error. In 10x10g3c, the spatial diversity coefficient of outermost node is hundred times larger with AnchorsOut option.

We can also discuss how censored distance estimation algorithms influence the resulting position estimation. As explained in Section 3.3 and verified in Section 5, Path tends to underestimate large distances. This results in the implosions observed in Figure 8.10 (b). The underestimation results in all distance estimates to anchors to be similar. Thus, nodes are placed toward the center of the area. The implosion coefficient of Path in Figure 8.10 (b) is 2.2995 compared to 1.0526 of LMDS in Figure 8.10 (a).

#### 8.2.2 Acoustic Time of Flight-Based Ranging Simulation

Using the acoustic time of flight-based ranging model, we performed position estimation using LMDS and MLAT under three scenarios: 8x8va, 9-5x9-5g1a, and 10x10g1a. In 8x8va nodes were placed in a 8x8 grid with varied spacing as shown in Figure 8.11. With the AnchorsIn option, anchors were placed at (1.2, 1.2), (1.2, 2), and (2, 1.2), and (2, 2). With the AnchorsOut option, anchors were placed at (0, 0), (0, 5.6), (5.6, 0), and (5.6, 5.6). In 9-5x9-5g1a nodes were placed in a 9x9 grid with one-meter spacing with a 5x5 hole in the center (Figure 8.12). With the AnchorsIn option, anchors were placed at (2, 2), (2, 7), (7, 2), and (7, 7). With the AnchorsOut option, anchors were placed at (0, 0), (0, 9), (9, 0), and (9, 9). In 10x10g1a nodes were placed in a 10x10 grid with one-meter spacing (Figure 8.13). With the AnchorsOut option, anchors were placed at (0, 0), (0, 9), (9, 0), and (9, 9).
Figure 8.11: Position Estimations (8x8va: 8x8, Varied Spacing, the Acoustic Ranging) (o: true positions, +: estimated positions)
Figure 8.12: Position Estimations (9-5x9-5g1a: 9-5x9-5, one-meter spacing, the Acoustic Ranging) (o: true positions, +: estimated positions)
Figure 8.13: Position Estimations (10x10g1a: 9-5x9-5, one-meter spacing, ChipCon signal strength-based ranging) (o: true positions, +: estimated positions)
With the AnchorsIn option, anchors were placed at (4, 4), (5, 4), (4, 5), and (5, 5).

LMDS and MLAT both improved significantly better when ranging errors were reduced. This is further discussed in Section 5.1. Although less severe than seen in Figure 8.10 (b), implosion of MLAT is again observed in Figure 8.13 (b) with the implosion coefficient of 1.0684 (compared to 0.9958 of LMDS). Again, Path’s tendency to underestimate large distances is a source of error.

Lack of spatial diversity resulted in inaccuracies of MLAT in position estimations toward the edge of network in Figure 8.11 (b) and (d) and Figure 8.13 (b) and (d). Again, underestimation of larger distances moves the outer points inward. The spatial diversity coefficients for the outermost node for 8x8va were 9.6412 and 0.1131 for AnchorsOut and AnchorsIn options respectively. For 10x10g1a, they were 3.8891 and 0.1286.

8.3 Chapter Summary

We evaluated the performance of LMDS through experiments and simulations. More specifically, we compared the performance of LMDS to multilateration, using both signal strength-based ranging and acoustic time-of-flight based ranging. Since multilateration is a basic algorithm used by a majority of localization systems in wireless cellular networks and wireless sensor networks, we chose to compare LMDS with multilateration. The experiments employ wireless sensor motes [8] developed by the UC Berkeley Intel Lab. The motes are equipped with sensors, computational processors, wireless radio, and ranging devices. Ranging data is collected from each node and transmitted to a central node via radios. Each experiment was run several times to reduce the effect of random fluctuations, and nodes were shuffled around to discount any hardware dependencies. In addition to these experiments, we further tested the performance using simulations based on experimental data.

As expected, LMDS outperformed multilateration. LMDS performed significantly better under signal strength-based ranging. This is because signal strength-based ranging used in our experiments was quite noisy. Nevertheless, by incorporating more ranging information, LMDS produced reasonable results despite the noisy ranging data.
Chapter 9

Conclusion

9.1 Summary

In this dissertation, we explored two basic components of localization: distance estimation algorithms and location estimation algorithms. For distance estimation, we surveyed distance measurement techniques, examined censored distance estimation, expanded concepts in trigonometric constraints-based censored distance estimation, and introduced Trigonometric $k$-clustering (TKC). TKC, a robust censored distance estimation algorithm, estimates the true distance by taking the average of a cluster formed by all trigonometric constraints. We surveyed location estimation algorithms and introduced LMDS, a robust location estimation algorithm. LMDS is a centralized location estimation algorithm that uses all available distance and position estimates. LMDS obtains initial positional estimates using multidimensional scaling and refines the estimates using all distance and position estimation. All algorithms were tested using real radio signal strength and ultrasonic wave flight time ranging data. We also developed a simulation tool based on models of experimental ranging data. We expect this tool to be useful in evaluating future localization algorithms in wireless sensor networks. Using both the experiment and simulation data, we compared the algorithms under various conditions.

The first part of this dissertation (Chapter 2 to Chapter 5) discussed distance estimation. Distance estimation consists of distance measurement and censored distance estimation. In terms of distance measurement or ranging, we investigated four popular technologies: network connectivity, radio signal strength, radio frequency time-of-flight, and acoustic time-of-flight. In network connectivity-based ranging, a node attempts to bound the location of another node using a model of ranging area. For
instance, node \(i\) may assume that its omni-directional radio can reach all nodes within some distance from the node, so all nodes connected to \(i\) are assumed to be in a circular range of node \(i\). In radio signal strength-based ranging, a receiver measures the attenuation of a signal to estimate its distance to the transmitter. If the receiver knows at what strength the signal has been transmitted, it can estimate the distance to the transmitter using received signal strength combined with a knowledge of how the signal is attenuated. A strong reason to explore the feasibility of network connectivity and radio signal strength-based algorithms is that often those two algorithms do not require extra hardware, since most wireless devices also contain signal strength-based communication devices. However, these two ranging methods do not usually give accurate results due to difficulties in modeling the ranging area and significant environmental effects on signal propagation. In some of our experiments and simulations, network connectivity-based algorithm did not discriminate shortest distance from the longest distances. We also found that signal strength-based ranging was unreliable, especially indoors.

On the other hand, both radio frequency and acoustic time-of-flight based ranging are more reliable alternatives. The GPS system estimates locations based on time-difference-of-arrival of simultaneously broadcast radio frequency signals by orbiting GPS satellites. While reliable and sufficiently accurate for most outdoor applications, GPS is often unreliable indoors and near tall buildings. To complement GPS in urban areas, recently developed radio frequency time-difference-of-arrival technologies estimate the distance between mobiles and base stations. Among them are Enhanced Observed Time Difference (E-TOD) [5] and Uplink time-difference-of-arrival (U-TDOA) [14]. In U-TDOA, base stations compare the time-difference-of-arrival of a cellular phone transmission. In E-TOD, the cellular phone keep track of arrival times of simultaneously broadcast transmissions from nearby base stations. On the other hand in wireless sensor networks, many location estimation systems employ acoustic time-of-flight based algorithms. In both Calamari [42] and Cricket [23], a radio frequency signal is broadcast simultaneously with the acoustic signal as a time synchronizing signal. Since the radio frequency signal travels “instantly” compared to the acoustic signal, the receiver measures the time difference of arrival between the radio frequency and the acoustic signal to measure the time-of-flight. While accurate with a resolution of a few centimeters, acoustic-based ranging tends to be shorter range compared to radio frequency-based ranging.

Censored distance estimation algorithms estimate “multi-hop” distances using distance measurements of neighboring nodes. We discussed several censored distance estimation algorithms including
Section 9.1. Summary

Simple Substitution, Shortest Hop, Shortest Path, and Trigonometric Resolution. Substitution Method replaces censored distance with a predetermined value. Shortest Hop and Shortest Path algorithms approximate the distance between two nodes with the shortest path between them. Both Shortest Hop and Shortest Path suffer from their tendency to underestimate larger distances due to the preference of negative error. Trigonometric Resolution algorithms estimate censored distances using trigonometric constraints from two adjoining triangles. The four methods discussed were Random Resolution Method, Threshold Resolution Method, Euclidean Method, and Degenerate Triangle Method. Essentially, the four methods represent different ways to choose between two configurations, corresponding to the same set of trigonometric constraints between two nodes. Random Resolution Method chooses at random. Threshold Resolution Method eliminates improbable configurations where a node is within ranging area of the other. In Euclidean Method, a node chooses between the configuration based on what ratio of its neighbors are connected to the other node. Degenerate Triangle Method eliminates configurations that violate the triangle inequality.

We introduced TKC, a Multiple Trigonometric Resolution Method that estimates censored distance given multiple trigonometric constraints. TKC is based on the understanding that every ambiguous pair corresponds to a trigonometric constraint, containing a random variable that corresponds to the true configuration. With small estimation errors, each pair is likely to contain an element that is close to the true distance, \(d(i, j)\). TKC attempts to identify clusters among the pairs, such that a cluster is formed near \(d(i, j)\) using a variant of \(k\)-clustering method. In our experiments and simulations, TKC outperformed all other censored distance estimation methods.

In the second part of this dissertation (Chapter 6 to Chapter 8) we investigated location estimation algorithms, which can be classified into two algorithms: Geometrical Confinement and Iterative Refinement. Geometrical Confinement algorithms estimate target positions using geometry of connections (distance estimates) and nodes (position measurements). The Iterative Refinement algorithms estimate positions by minimizing error using a gradient search. Many localization algorithms use combinations of these two strategies. We discussed two Geometrical Confinement Algorithms: Bounding Box Method and Convex Position Estimation. In Bounding Box Method, each node bounds its position by ranging area of neighboring nodes approximated by a bounding box. Convex Position Estimation is a centralized algorithm that approximates the range area with convex shapes such as triangles for directional ranging and circles for omni-directional ranging. Iterative Refinement Algorithms attempt
to find position estimates given an initial estimate by iteratively minimizing error. Multilateration, also called triangulation, is an Iterative Refinement Algorithm. Iterative Multilateration and Multi-hop Multilateration are variations of multilateration, which allow localization based on distance reference locations.

LMDS, introduced in Chapter 7, can be divided into three stages: Distance Estimation, Placement, and Coordinate Assignment. In Distance Estimation, LMDS estimates the distances between all nodes from position and distance measurements. In Placement, LMDS applies classical MDS to obtain a set of placements according to the distance estimates. The first step in Placement is classical MDS. Using the Law of Cosine, the Classical MDS takes the distance estimates and returns $N-1$ dimensional placements of the nodes, where $N$ is the number of nodes. The second step is to obtain low dimensional placements corresponding to the physical space; if we are interested in locating nodes in a plane, we would want two-dimensional placements. LMDS takes the projection by taking principal components of the placements. In Coordinate Assignment, LMDS takes the placements from Placement and changes the coordinate system of the placements according to positional estimates. Specifically, we first find a change of basis such that nodes are placed near their position estimates. Lastly, we run a refinement algorithm that adjusts the placements to match more closely both position and distance estimates. In our experiments and simulations, we compared LMDS with multilateration, an algorithm used by a majority of localization systems. LMDS performed better than multilateration. LMDS performed significantly better under signal strength-based ranging. This is because signal strength-based ranging data used for our experiments was quite noisy.

9.2 Future Research

In our research we attempted to create a robust location estimation algorithm in the presence of noise in measurements. Our research was motivated primarily by the inherent difficulties in reliably measuring distances in wireless cellular networks and wireless sensor networks, due to unpredictable environmental effects on ranging technology. Recognizing that both distance and position measurements support a geometrical mesh structure, we assumed if all measurements were collected at a centralized node, we could rectify conflicting noisy observations. The enriched set of observations at the centralized node can enhance each measurement, improve censored distance estimation, allow tighter construction
of a larger mesh structure, and strengthen the refinement based on a larger data set. We have seen from the performance analysis that TKC and LMDS improve the reliability of location estimation.

Moreover, there can still be much improvement in TKC and LMDS. For example, TKC’s requirement for a certain set of trigonometric constraints may be too restrictive. To estimate the distance between nodes $i$ and $j$, TKC requires at least two nodes $l$ and $k$, such that $l, k \in N(i)$, $l, k \in N(j)$, and either $l \in N(k)$ or $k \in N(j)$. In sparse networks, it is possible that node $i$ has only one other neighbor $k$, shutting out the possibility for TKC. Shortest Path Method, which is more resilient than TKC, can still return distance estimates: $d(i, j) = d(i, k) + d(k, j)$, $\forall k$. Our implementation of TKC relies on Shortest Path Method for such a case. We would like to explore other possibilities.

TKC can benefit from centralized location estimation by choosing which censored distance should be estimated first. This ordering is important because any reliable estimate could be referenced in future estimation. Naturally, estimation should be ordered such that the distance with the highest chance of success is estimated first. In our implementation, TKC first estimates the shortest estimated distance, which is obtained from Shortest Path Method. We found the ordering to be useful. TKC with the ordering performed remarkably better than TKC with a random order. However, our ordering scheme is solely based on the intuition that shorter distances should be easier to estimate. We believe that a better ordering strategy is possible if distance measurements can be reliability estimated.
Bibliography


