ON LOCATION SUPPORT AND ONE-HOP DATA COLLECTION IN
WIRELESS SENSOR NETWORKS

by

Kiran Kumar Yedavalli

A Dissertation Presented to the
FACULTY OF THE GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(Electrical Engineering)

May 2007

Copyright 2007 Kiran Kumar Yedavalli
Epigraph

phi·los·o·phy

*n. pl.* phi·los·o·phies

1. Love and pursuit of wisdom by intellectual means and moral self-discipline.

2. Investigation of the nature, causes, or principles of reality, knowledge, or values, based on logical reasoning rather than empirical methods.

3. A system of thought based on or involving such inquiry.

4. The critical analysis of fundamental assumptions or beliefs.

5. The disciplines presented in university curriculums of science and the liberal arts, except medicine, law, and theology.

6. The discipline comprising logic, ethics, aesthetics, metaphysics, and epistemology.

7. A set of ideas or beliefs relating to a particular field or activity; an underlying theory.

8. A system of values by which one lives.

the·sis

*n. pl.* the·ses (-sz)

1. A proposition that is maintained by argument.

2. A dissertation advancing an original point of view as a result of research, especially as a requirement for an academic degree.
Dedication

To my Parents, Akka, and my wife Divya, for their love and support.
Acknowledgements

It has been my distinct honor and pleasure to work with Prof. Bhaskar Krishnamachari during my PhD studies at the University of Southern California. I am deeply thankful to him for being my advisor and for extending his much needed support and encouragement all through my stay at USC. He has been a great inspiration during difficult times and a true guide, not just in academic matters, but in matters of life in general. Through his compassion and enthusiasm he has been a true role model to be followed. I have learned much from him about being a good human being, which I believe is going to stay with me all through my life.

It is a special honor to me that Prof. Ramesh Govindan and Prof. Antonio Ortega have agreed to be on my thesis dissertation committee and I am very thankful to them for this.

I am eternally indebted to my parents and my sister for their love and support without which I wouldn’t have been what I am today. I am very fortunate to be able to share my happiness with my loving wife Divya who has been very patient and supportive of all my endeavors.

I am greatly thankful to my friends for being there for me when I needed them and for making my stay at USC a bouquet of many happy moments and rich experiences. Through my friends and colleagues in the Autonomous Networks Research Group (ANRG) I have had the opportunity to work with some remarkable and truly outstanding individuals and learn much from them. I am grateful to my friends outside of the group - Satya, Harish, Suraj, and others - for being there for me during moments of the greatest need. I would like to thank the Electrical Engineering department for making my stay at USC very comfortable through their support and services.
# Table of Contents

- Epigraph ii
- Dedication iii
- Acknowledgements iv
- List Of Tables viii
- List Of Figures ix
- Abstract xiv

## 1 Introduction 1
1.1 Wireless Sensor Networks 1
1.2 Location Support 2
1.3 Medium Access for One-Hop Data Collection 3
1.4 Research Contributions 4
1.5 Thesis Organization 7

## I Location Support for Mobile Nodes 9

## 2 Background on RF Localization 10
2.1 Introduction 10
2.2 Finger-Printing Techniques 11
2.3 Non-Finger-Printing Techniques 13
2.4 Chapter Summary 17

## 3 Accurate RF Localization 19
3.1 Introduction 19
3.2 Localization Technique I: Ecolocation 21
3.2.1 Ideal Scenario 22
3.2.2 Real World Scenario 24
3.2.3 Location Determination 25
3.2.4 Examples 27
3.3 Localization Technique II: Sequence-Based Localization 28
3.3.1 Location Sequences 29
3.3.2 Localization Procedure 31
3.3.3 Maximum Number of Location Sequences 32
### Fast & Fair Localization

#### 4.1 Introduction

#### 4.2 Motivation

#### 4.3 Assumptions and Terminology

#### 4.4 Definitions

#### 4.5 Problem Formulation

#### 4.6 Scheduling Algorithm

#### 4.7 Metrics

#### 4.8 Evaluation

##### 4.8.1 Analysis

- **4.8.1.1 Grid Deployment**
- **4.8.1.2 Random Deployment**

##### 4.8.2 Simulations

#### 4.9 Discussion

#### 4.10 Chapter Summary

### Medium Access for One-Hop Data Collection

#### 5.1 Introduction

#### 5.2 Application Space

#### 5.3 Medium Access Techniques

##### 5.3.1 Problem Description

- **5.3.1.1 Metrics**
- **5.3.1.2 Energy Model**

##### 5.3.2 Slotted Aloha Medium Access

##### 5.3.3 Carrier Sense Medium Access

- **5.3.3.1 IEEE 802.15.4**
List Of Tables

3.1 Constraints on the unknown node for the example in Figure 3.1(a)......... 23

3.2 Constraints for the example of Table 3.1 when the ranks of third and fourth
ranked reference nodes are interchanged due of multi-path effects............. 25

3.3 Progression of number of location sequences with number of reference nodes (n)
in the localization space. The last two columns compare the simulation and
analytical results for the maximum number of location sequences. Simulation
results are gathered from 1000 random trials (with 100 different random seeds)
in each of which n reference nodes were placed uniformly at random in a square
localization space......................................................... 40

3.4 Typical values and ranges for different simulation parameters............. 52

3.5 Comparison of worst-case computational complexities of SBL, LSE, Proximty and
3-Centroid................................................................. 59

4.1 Simulation parameters and their values........................................ 82

4.2 Comparison of analytical lower and upper bounds of M with simulation results for
grid deployment of different value of N, the network size. Note that the number of
reference nodes in the in-square of a cell is n = 2m^2 + 6m + 5, m = (\frac{R \cdot d}{4} - 1) where,
R is the radio range and d is the inter reference node distance (Proposition 4 in
Section 4.8.1.1)..................................................................... 84

4.3 Comparison of analytical lower and upper bounds of M with simulation results
for random deployment of different value of N, the network size. The simulations
results are average over 10 different random reference node network topologies. .. 85

8.1 Notation................................................................. 137

8.2 Performance comparison of Original and Enhanced IEEE 802.15.4 MAC for CD
in term of throughput (\Phi_{CD}(N)) in Kbps for Low density networks............. 157
List Of Figures

2.1 Classification of localization techniques. ................................................................. 10

3.1 The distance rank order of reference nodes (A, B, C, D) is different for different regions (X₁, X₂) in the localization space. ................................................................. 23

3.2 Real world experimental results: Reference nodes far from the unknown node may measure higher RSS values than closer reference nodes. Note that y-axis is reverse ordered. ................................................................. 24

3.3 Ecolocation location estimate (E) for the unknown node (P) at (1, 3) for a grid layout of 9 reference nodes (A). The number adjacent to a reference node is its corresponding rank. The location error is expressed in meters where the size of the square localization area is 12 × 12 sq. meters. (a) Sequence: 123456789 (no erroneous constraints) [Estimate: (1.25, 3.3); Error: 0.34 meters] (b) Sequence: 123568497 (13.9% erroneous constraints) [Estimate: (1.25, 1.95); Error: 1.07 meters] (c) Sequence: 125379486 (22.2% erroneous constraints) [Estimate: (1.95, 1.25); Error: 1.98 meters] (d) Sequence: 243976581 (44.4% erroneous constraints) [Estimate: (1.95, 1.25); Error: 1.98 meters]. ................................................................. 28

3.4 (a) The perpendicular bisector of the line joining two reference nodes divides the localization space into three distinct regions. (b) Illustration of arrangement of 6 bisector lines for 4 reference nodes placed uniformly randomly in a square localization space. ................................................................. 29

3.5 (a) Examples of location sequences for a four reference node topology. (b) All feasible location sequences for the topology of (a). ................................................................. 31
3.6 Addition of fourth reference node $D$ adds 3 new bisector lines to the localization space. (a) The first of the 3 new bisector lines, line 1, the perpendicular bisector of $CD$, creates 3 new vertices (equal to the number of pre-existing lines in the localization space), 4 new faces and 7 new edges at most. (b) The second line, line 2, the perpendicular bisector of $BD$, has to pass through the intersection point of the bisectors of $CD$ and $BC$ because, $\{BD, CD, BC\}$ form a triangle and the perpendicular bisectors of the three sides of a triangle intersect at a single point. Therefore line 2 creates 2 new vertices, 4 new faces and 6 new edges at most. (c) Similarly, line 3, the perpendicular bisector of $AD$ has to pass through the intersection points of perpendicular bisectors of $AB$, $BD$ and $AC$, $CD$ as $\{AD, AB, BD\}$ and $\{AD, AC, CD\}$ are two triangles with a common side $AD$. Therefore, line 3 creates 1 new vertex, 4 new faces and 5 new edges at most.

3.7 RF channel non-idealities could corrupt a location sequence from the feasible space either to another sequence in the feasible space or to a sequence in the infeasible space.

3.8 Robustness examples: Location estimate (E) for the unknown node (P) at (1, 3) for a grid layout of 9 reference nodes. The number adjacent to a reference node is its corresponding rank. The location error is expressed in meters where the side length of the square localization area is 12 meters. (a) $(T = 1, \tau = 1)$, Estimate (E): (1.33, 1.33), Location Error: 0.46 meters (b) $(T = 0.722, \tau = 0.783)$, Estimate (E): (2.0, 2.0), Location Error: 1.4 meters (c) $(T = 0.556, \tau = 0.667)$, Estimate (E): (2.0, 2.0), Location Error: 1.4 meters (d) $(T = 0.111, \tau = 0.278)$, Estimate (E): (2.0, 1.33), Location Error: 1.94 meters.

3.9 Overlap of Ecolocation scanning grid and regions created by arrangement of bisector lines in SBL.

3.10 Simulation results averaged over 1000 random trials (with 100 different random seeds) in each of which $n$ reference nodes were placed uniformly at random in a 2D square localization area of $S \times S$ sq. meters. (a) The average maximum, average and average minimum face areas as a function of the number of reference nodes. (b) The average maximum, average and average minimum edge lengths as a function of the number of reference nodes. $K1$, $K2$ and $K3$, $K4$ are scaling constants.

3.11 Average location error as measured using Spearman’s correlation and Kendall’s Tau as a function of the number of reference nodes.

3.12 Sequence corruption: Cumulative distribution function of Kendall’s Tau $T$ between the RSS location sequence and true location sequence for varying (a) standard deviation ($\sigma$) (b) path loss exponent ($\eta$) (c) number of reference nodes ($n$).

3.13 Performance: (a) Average location error as a function of RF channel parameters - standard deviation ($\sigma$) and path loss exponent ($\eta$). (b) Average location error as a function of node deployment parameters - number of reference nodes ($n$) and reference node density ($\beta$). (c) Average location error as a function of the location of the unknown node.
3.14 (a) Average location error as a function of the sequence corruption \((T)\) and as a function of the distance \((\tau)\) between the corrupted sequence and its nearest feasible sequence in the location sequence table. (b) Correlation between \(\tau\) and \(T\).

3.15 *Comparison:* Average location error due to SBL, LSE, Proximity and 3-Centroid as a function of standard deviation of RSS log-normal distribution \(\sigma\) for different values of path loss exponent \(\eta\). (a) \(\eta = 2, n = 10\) (b) \(\eta = 4, n = 10\) (c) \(\eta = 6, n = 10\) and for different values of number of reference nodes \(n\). (a) \(n = 4, \eta = 4\) (b) \(n = 7, \eta = 4\) (c) \(n = 10, \eta = 4\).

3.16 *Outdoor experiment:* 11 MICA 2 motes, placed randomly in a 144 sq.meters area, were used as reference nodes as well as unknown nodes. Consequently, each unknown node had 10 reference nodes. (a) Path loss exponent calculation, \(\eta = 2.9\). (b) Comparison between true locations and SBL location estimates. (c) Location error due to SBL, LSE, Proximity and 3-Centroid (the nodes are ordered in increasing error of SBL). (d) Corruption measure \(T\) and error indicator \(\tau\).

3.17 *Indoor experiment:* 12 MICA 2 motes, placed randomly in a 120 sq.meters area, were used as reference nodes. The location of the unknown node was estimated for 5 different locations using the 12 reference nodes. (a) Path loss exponent calculation, \(\eta = 2.2\). (b) Comparison between true path and SBL estimated path. (c) Location error due to SBL, LSE, Proximity and 3-Centroid (the nodes are ordered in increasing error of SBL). (d) Corruption measure \(T\) and error indicator \(\tau\).

4.1 Illustration of a cell, its in-square and out-square.

4.2 Example illustrating terminology.

4.3 Expression for localization delay.

4.4 Three different cases (1), (2) and (3) depending on the relative position of localization request arrival time with respect to the times slots of reference nodes in the cell.

4.5 Simulation Results: (a) Average localization delay \(D_{avg}(k)\), (b) Localization fairness \(F(k)\), (c) Average localizable speed \(V_{avg}(k)\), (d) Minimum localizable speed \(V_{min}(k)\), and (e) Maximum localizable speed \(V_{max}(k)\); as a function of number of reference nodes required for localization \(k\), for five reference node density values and for grid and random deployment of reference nodes. (f) \(V_{avg}(k), V_{max}(k),\) and \(V_{min}(k)\), (g) Average localization delay \(D_{avg}(k)\); as a function of reference node density \(\beta\) for number of reference nodes required for localization \(k = 8\), for grid and random deployments of reference nodes. (h) Localization fairness \(F(k)\) as a function of reference node density \(\beta\) for four different levels of location estimate accuracy \((k)\) for grid and random deployments of reference nodes.

5.1 Application space spectrum for one-hop data collection in wireless sensor networks. The color transition from red to blue indicates the spectrum transition from infinite packets in the queues to single-packet in the queues of contending nodes.
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.5</td>
<td>Expected delay and energy consumption in an epoch with ( n ) nodes as a function of transmission probability, ( p ), for different values of packet length ( L )</td>
<td>148</td>
</tr>
<tr>
<td>8.6</td>
<td>Ratio of expected delays and energy consumptions for consecutive epochs.</td>
<td>149</td>
</tr>
<tr>
<td>8.7</td>
<td>Comparison of transmission probabilities for IEEE 802.15.4 and optimal ( p )-persistent CSMA for CD and OSD.</td>
<td>152</td>
</tr>
<tr>
<td>8.8</td>
<td>Flow chart for Enhanced IEEE 802.15.4 operation at a node.</td>
<td>155</td>
</tr>
<tr>
<td>8.9</td>
<td>Performance of Channel Feedback Enhanced IEEE 802.15.4.</td>
<td>155</td>
</tr>
</tbody>
</table>
Abstract

We consider two fundamental building blocks for many applications in wireless sensor networks - location support and efficient medium access for one-hop data collection.

In the first part of the thesis we identify two important problems of location support - accurate localization and fast & fair localization - and propose novel solutions. We address the problem of accurate localization by proposing two novel, light-weight RF localization techniques called *Ecoloration* and *Sequence-Based Localization*. We define constructs called *location constraints* and *location sequences* based on distance ranks of reference nodes from the location of the unknown node and use them for localization. We compare and contrast the two localization techniques and show their robustness to RF channel non-idealities through examples. Through extensive systematic simulations and a representative set of real mote experiments, we show that our light-weight RF localization techniques provide comparable or better accuracy than other state-of-the-art radio signal strength-based localization techniques over a range of wireless channel and node deployment conditions.

In addition to being accurate, the location support service should also be fast and fair. The response times of the reference nodes to localization requests from the unknown node should be minimized and multiple unknown nodes, at different locations, should not have widely varying response times. We identify this as a fast/fair localization problem and formulate it as a min-max optimization problem, show that it is related to the well-known, NP-hard, maximum broadcast frame length problem, and investigate a heuristic scheduling based solution. We study the attributes that determine the response times of the reference nodes, called the *localization*
delay, and derive closed-form expressions for it. We then investigate the heuristic solution’s performance in terms of localization delay, fairness and average and minimum localizable speeds.

In the second part of the thesis, we address the problem of medium access for one-hop data collection, which occurs frequently in many wireless sensor network applications. We consider a wide spectrum of one-hop data collection applications with continuous data collection at one end and one-shot data collection at the other. While in the continuous data collection problem the contending wireless nodes always have a packet to transmit, in the one-shot data collection problem each contending node has a single packet to transmit. Medium access mechanisms for continuous data collection have been studied extensively in the past by numerous researchers, but such mechanisms for one-shot data collection have received much less attention. In this thesis we address the medium access problem for this spectrum of application scenarios through three different pieces of work.

We model and analyze the performance of slotted Aloha medium access techniques for the one-shot data collection problem. Owing to the transient nature of the network in this problem we use non-ergodic Markov chain analysis and derive flow equations that accurately capture the temporal dynamics of the network. Using these equations we evaluate the medium access techniques’ performance in terms of delay and energy consumption.

We then present a novel location-aware medium access protocol for the one-shot data collection problem that uses the location information of contending nodes to reduce collisions and improve the overall performance. We evaluate the protocol in terms of delay and energy consumption and compare it with location-unaware medium access protocols using simulations. Results show that our protocol can take advantage of the location distribution of nodes to provide significantly lower delay and energy consumption compared to location-unaware protocols.

Finally, we model, analyze, and evaluate the performance of the IEEE 802.15.4 MAC protocol for both ends of the one-hop data collection application spectrum. We find that the IEEE 802.15.4 MAC protocol performs poorly for one-hop data collection in dense sensor networks,
showing a steep deterioration in both throughput and energy consumption with increasing num-
ber of transmitters. We propose a channel feedback-based enhancement to the protocol that
is significantly more scalable, showing a relatively flat, slow-changing total system throughput
and energy consumption as the network size increases. A key feature of the enhancement is
that the back-off windows are updated after successful transmissions instead of collisions. The
window updates are based on an optimality criterion we derive from mathematical modeling of
p-persistent CSMA.
Chapter 1

Introduction

In this thesis we study two key problems in wireless sensor networks - location support and efficient medium access for one-hop data collection.

1.1 Wireless Sensor Networks

Wireless sensor networks are an emerging paradigm that promise to change the way humans interact with their environments [59]. The sensor nodes used in a wireless sensor network are inexpensive, intelligent devices that run on batteries and can “sense” the environment in their vicinity for physical metrics such as temperature, humidity, light-intensity, etc. The sensor nodes can be programmed to communicate among themselves to form networks that could provide many mission critical and quality-of-life enhancing applications. Given these advantages, the emerging trend indicates the deployment of these wireless sensor nodes as an integral part of essential infrastructure such as homes, office buildings, roads, etc. Since wireless sensor networks are envisioned to work unassisted and uninterrupted for many years, energy savings are of critical importance for their operation.
1.2 Location Support

A location support service enables its users, mobile or static, to determine their location coordinates, by themselves, with respect to a reference provided by the service. Location support services not only enable many user-experience enhancing services but also support many mission critical applications. For example, location support services in venues such as museums, zoos, and airports can be used to provide experience-enhancing navigational services to guests. On the other hand, the same location support services can prove to be life saving in emergency situations such as fires or medical assistance to residents of old-age homes, or they could be a critical part of many businesses, working as productivity enhancers in factory floors and warehouses. In recent years, wireless sensor networks have emerged as key enablers of location support services.

The sensor nodes of location support enabling wireless sensor networks are programmed with their location information in terms of Cartesian coordinates with reference to a pre-defined coordinate system. Mobile devices that request location support services obtain the location coordinates of these nodes, via the wireless medium and use them to determine their own location. Thus, mobile devices, that are unknown nodes\(^1\), take the help of sensor nodes, that act as reference nodes, to determine their locations using efficient localization algorithms.

In any solution to an engineering problem the cost of the solution should be an essential consideration. During our interaction with wireless sensor networks engineers at Bosch Research in Palo Alto, California [3], it was concluded that the cost of deployment of sensor nodes in large numbers prohibits incorporation of special hardware for location support purposes in them. Thus wireless radios which are used by the sensor nodes for the essential task of communication have emerged as the key enablers for cost effective location support. Thus we focus only on radio frequency-based location support systems for wireless sensor networks. Also, the location

\(^1\)In this thesis, we use the terms “mobile devices” and “unknown nodes” interchangeably.
support system should use minimum possible energy of the reference nodes in the sensor network and at the same time it should be inexpensive to implement, deploy and operate.

We identify two key problems in location support systems for wireless sensor networks - (i) accurate localization and (ii) fast and fair localization - and provide efficient algorithms to solve them that comply with the above conditions for an effective location support system.

1.3 Medium Access for One-Hop Data Collection

The problem of one-hop data collection occurs frequently in many wireless sensor network applications such as location support, neighbor discovery, data-querying, etc. In this problem, a single data sink seeks interesting data from data sources in its radio range. For example, in location support, the unknown node (data sink) seeks the location coordinates of reference nodes (data sources) in its radio range. Similarly in neighbor discovery a node in a network (data sink) seeks to discover its neighboring nodes (data sources). The problem of neighbor discovery occurs as a building block of many wireless sensor network functions such as multi-hop routing of data.

One of the most important application areas for wireless sensor networks is sensor data collection. A mobile or static querier (data sink) seeks to pull interesting and relevant data from the sensor nodes (data sources) in the wireless sensor network that are in its one-hop radio range using specific data queries. Thus the problem of one-hop data collection opens itself to a broad spectrum of applications ranging from continuous data collection at one end and one-shot data collection at the other end of the spectrum. While in the continuous data collection problem the contending wireless nodes always have a packet to transmit, in the one-shot data collection problem each contending node has a single packet to transmit.

We review the applicability and performance of commonly known wireless medium access mechanisms, such as slotted Aloha multi-access schemes, carrier sensing medium access (CSMA)
mechanisms and the IEEE 802.15.4 MAC protocol, to this new spectrum of one-hop data-collection applications specific to wireless sensor networks through mathematical modeling and performance analysis. We also propose a novel medium access protocol that addresses the one-shot data collection end of the spectrum.

1.4 Research Contributions

In this section, we provide a high-level description of our research contributions in the areas of radio-frequency-based location support and medium access mechanisms for one-shop data collection.

1. Location Support for Mobile Nodes:

(a) **Accurate RF Localization**: Accurate localization is essential to provide fine-grained localization in feature-rich areas, such as office buildings and factories. This can be achieved by employing many different physical signal types such as radio signals, infra-red, ultra-sound etc. Depending on the environment, each type of signal behaves differently. For example, infra-red signals are sensitive to the ambient light intensity and ultra-sound signal behavior depends on the level of humidity in the environment. Significantly, radio signals are the most sensitive to the wireless channel. They are subjected to wireless channel non-idealities such as multi-path, shadowing, refraction, diffraction and thermal noise. Owing to this behavior of radio signals, our choice of using only radio signals for localization, poses a significant challenge to achieving the desired level of accuracy in indoor environments.

Many researchers have proposed many different solutions to address this challenge. However, in Chapter 2, we will show that the specific challenges posed by the condition of a cost and energy efficient location support system requires a fresh new approach. In this thesis, we propose a novel geometric-constraints based approach to address
the challenge of using radio signals for accurate localization. We present two light-weight localization techniques provide accurate localization using only radio signals and at the same time very inexpensive to implement, deploy and operate using the minimum possible energy resources.

(b) **Fast & Fair Localization**: The response time of the reference nodes to the unknown node’s localization request places limits on how fast it can move while obtaining accurate location estimates. The unknown node should be able to communicate with all the required number of reference nodes for localization before it moves to its next location. Conversely, the speed of movement of the unknown node determines the frequency of localization requests and the response time of the reference nodes should be able to match this frequency. In either case, the response time of the reference nodes to localization requests should be minimized in order to ensure fast localization. In addition to being fast, the response time of the reference nodes should not change drastically with the location of the unknown node in the localization area, i.e., the response-time-limited speed of movement of the unknown node should not be significantly lower or higher at some locations compared to others. Alternatively, if multiple unknown nodes at different locations request location support simultaneously, the response time for all requests should be similar. In order to achieve this, the variation in the reference node response time over all locations of the unknown node should be minimized. To the best of our knowledge, no one has looked at this problem till now. In this thesis, we formally define the fast/fair problem and propose solutions.

2. **Medium Access for One-Hop Data Collection:**

(a) **Modeling and Analysis of Slotted Aloha Multi-Access Techniques for One-Shot Data Collection**: The application of slotted Aloha multi-access techniques to the continuous data collection end of the one-hop data collection application spectrum
has been a very well studied area for over a quarter-of-a-century. However, the one-shot data collection problem has gained importance only due to the emergence of wireless sensor networks. To the best of our knowledge ours is the first attempt at modeling and analyzing the slotted Aloha multi-access technique for the one-shot data collection problem. This problem is characterized by the presence of a single packet in the transmission queue of each contending node. Once that packet is transmitted the node ceases to be in contention of the wireless channel. This leads to a non-steady state, transient behavior of the wireless networks. In this thesis we use non-ergodic Markov chain model to analyze the network dynamics for this problem and derive flow equations that accurately capture the temporal behavior of the network. Using these equations we evaluate the performance of the protocol for the one-shot data collection problem.

(b) **Location-Aware Medium Access for One-Shot Data Collection:** We add to our contributions to efficient medium access protocols for the one-shot data collection end of the one-hop data collection application spectrum by presenting a novel location-aware medium access protocol for the same. In this protocol the contending nodes make use of their location information to reduce collisions among themselves and improve the overall performance. A performance comparison with location-unaware medium access protocols shows that the location-aware protocol can take advantage of the location distribution of nodes to provide significant performance gains.

(c) **Enhancement of IEEE 802.15.4 MAC Protocol:** The IEEE 802.15.4 protocol is an IEEE standard for low-rate, low-power wireless embedded networks. The standard specifies both the physical and medium access layers of the protocol. In this thesis we model, analyze, and evaluate the performance of the IEEE 802.15.4 MAC protocol for both ends of the one-hop data collection application spectrum. Simulation results
show that the IEEE 802.15.4 MAC protocol performs poorly for one-hop data collection in dense sensor networks, showing a steep deterioration in both throughput and energy consumption with increasing number of transmitters. We propose a channel feedback-based enhancement to the protocol that is significantly more scalable, showing a relatively flat, slow-changing total system throughput and energy consumption as the network size increases. A key feature of the enhancement is that the back-off windows are updated after successful transmissions instead of collisions. The window updates are based on an optimality criterion we derive from mathematical modeling of p-persistent CSMA.

Next, we provide a brief overview of the thesis organization.

1.5 Thesis Organization

The rest of the thesis is organized as follows:

In the first part of the thesis, we study the problem of location support. In Chapter 2, we discuss the background work on RF localization and in Chapter 3, we address the problem of accurate RF localization. We address the problem of fast and fair localization in Chapter 4.

In the second part of the thesis, we study the problem of efficient medium access for one-hop data collection. In Chapter 5, we review the background work on medium access technique for one-hop data collection. In Chapter 6, we model and analyze slotted Aloha medium access techniques for the one-shot data collection problem and in Chapter 7, we present a novel location-aware medium access technique for the one-shot data collection problem. In Chapter 8, we model, analyze, and evaluate the performance of the IEEE 802.15.4 MAC protocol for both ends of the one-hop data collection application spectrum and propose enhancements that improve the protocols’ performance.
We present our conclusions from this work in Chapter 9 and discuss directions for future work in Chapter 10.
Part I

Location Support for Mobile Nodes
Chapter 2

Background on RF Localization

2.1 Introduction

Providing accurate indoor localization has been an active area of research for quite some time. Many solutions using different technologies have been proposed in the past. Figure 2.1 gives a classification from [44] of the various localization technologies based on signal types, signal metrics, processing methods and location estimate ends.

![Classification of Localization Techniques](image)

Figure 2.1: Classification of localization techniques.

In our work we focus on RF signals and RSS signal metric based localization techniques, in the setting of a location support system. Other metrics of RF signals such as TDOA, TOF and
AOA require special or extra hardware to be incorporated on sensor motes ([79], [41], [75], [80], [57]).

In this chapter, we discuss the background work on RF, RSS-based localization techniques, from the literature. Here, we discuss the background work from the point of view of location estimate accuracy and complexity of implementation. All the previously proposed localization techniques can be broadly classified into finger-printing techniques and non-finger-printing techniques. First, we present a detailed review of finger-printing techniques and then, review the non-finger-printing techniques.

2.2 Finger-Printing Techniques

Finger-printing techniques are characterized by two phases of localization. The first phase, which is the pre-configuration phase or the off-line phase, involves “profiling” the localization space. At each location point in the localization space, RSS values from carefully placed access-points are gathered. Access-points perform the role of reference nodes in this case. This set of RSS values is the “finger-print” for that location. A finger-print-map is then created that maps each RSS based finger-print to the corresponding location point. Since it’s physically not possible to measure finger-prints for all locations in the localization space, interpolation and extrapolation mechanisms are used to determine the map for the entire area. Many different devices have been proposed to be employed to determine as accurate a map as possible.

In the second phase, which is the localization phase or the on-line phase, the unknown node records RSS values from access points in its range and uses the finger-print-map from the first phase to map its set of RSS values to its corresponding location. Since RSS values are subjected to random variations in the wireless channel, the RSS finger-print of the unknown node might not directly correspond to any location in the map. Therefore, different techniques such as nearest-neighbor classification, neural-network based learning, probabilistic estimation,
statistical pattern classification, etc, have been proposed to obtain the best location estimate given the unknown node’s finger-print.

One of the earliest reported works on radio finger-printing based localization techniques, using wireless LANs is RADAR by Bahl et al in [15]. In this work, in the first phase, a database is created by collecting multiple RSS data samples from up to 3 access-points at multiple location points, and for multiple orientations at each location point. In the second phase, the location is estimated using the database created in the first phase, using the nearest neighbor rule. The authors report a median location error of 2 to 3 meters for office environments using 3 base stations, 40 location points, and at least 3 samples and many orientations at each location point. The location error is the Euclidean distance between the true location of the unknown node and its location estimate.

In a closely related work in [19], the authors propose to use training methods, such an neural networks and multi-layer perceptron algorithms, on the data in the database created in the first phase, to obtain an accurate location estimate for the unknown node. After many iterations of training, they report a best location estimate error of 1.9 meters. The authors of [18] state that “the functional dependence between the signal strength from a number of radio points and the physical position is not deterministic, but a statistical law connecting signal strength and position”. In view of this they use the method of support vector machines to determine the location estimate and report an average error of 3.4 meters. In [44], the authors use Kalman filtering algorithms to increase location estimate accuracy in the second phase. They use an Ekahau Positioning Engine (a commercially available localization system) with 5 access-points and report a mean error of 2.5 meters.

While the previous three references have focussed on improving the accuracy in the second phase, researchers have proposed many different methods to improve the finger-print-map in the first phase of localization. In [60], the authors use four different types of devices - access-points, sniffers, stationary emitters, and a location estimation engine - that are connected through
a wired network to create the finger-print-map. Through careful placement of sniffers and stationary emitters, and collected RSS data at 100 location points, the authors achieve a mean location error of 3.3 meters. The authors in [40], build on the work in the previous reference by using sniffers to monitor the unknown node behavior over a period of time and report a similar error for 40 data points only. They show that the error can be improved by using more sniffers.

In [45], the authors take a hybrid approach in which multiple branches of information sources, such as wireless LANs and Bluetooth devices, and multiple localization techniques, such as triangulation, k-nearest-neighbors and smallest M-vertex polygon, are combined to estimate the location of the unknown node. They selectively weigh and fuse the information from each of the multiple information sources and localization techniques to determine the location estimate and report a mean location error of 2.2 meters. They also report a mean error of 3.3 meters when information fusion is not used.

An important aspect of all the above finger-printing localization techniques is that all of them have been designed for and implemented on wireless LANs networks. This is the main advantage for these localization techniques, in that, few, freely available wireless access-points can be employed to provide localization services. However, the main drawbacks of these techniques is the costly, time consuming first phase of obtaining the finger-print-map for the localization space. Moreover, since this map heavily depends on the features of the localization space, such as office partitions, furniture etc., it has to be recalculated every time there is a change in the localization space.

Next, we review RF, RSS-based non-finger-printing localization techniques.

### 2.3 Non-Finger-Printing Techniques

Non-finger-printing localization techniques are characterized by a single phase of operation, in which the unknown node measures RSS values of RF signals from the reference nodes in its
radio-range and uses them along with location coordinates of reference nodes to estimate its location.

One of the earliest non-finger-printing based localization techniques is the centroid method proposed by Bulusu et al. in [26]. In this technique, the location estimate of the unknown node is the centroid of all reference nodes in its radio range. Even though this is a very simple technique, this provides very coarse grained location estimates.

In [20], the authors convert the RSS measurements from reference nodes to distance estimates using a assumed power law relationship between them, and use triangulation to localize the unknown node. They assume certain values for the different constants of the power law. They address the problem of Rayleigh fading of the RSS samples when the unknown node is mobile and propose to collect many RSS samples in a window of time and use the sample average for localization. They present location error as a function of the window size. Localization techniques that convert RSS values to distance range values are called range-based techniques.

Another range-based localization technique using the maximum likelihood estimator (MLE) has been proposed by [77] and [100]. At the heart of the MLE technique is the radio propagation model which is assumed to be the log-normal shadowing model [83]:

\[
P_R(d) = P_T - PL(d_0) - 10\eta \log_{10} \frac{d}{d_0} + X_\sigma
\]  

(2.1)

where, \( P_R \) is the received signal power, \( P_T \) is the transmit power and \( PL(d_0) \) is path loss for a reference distance of \( d_0 \). \( \eta \) is the path loss exponent and the random variation in RSS is expressed as a Gaussian random variable of zero mean and \( \sigma^2 \) variance, \( X_\sigma = N(0, \sigma^2) \). All powers are in \( dBm \) and all distances are in meters. The MLE estimator is equivalent to the least-squares estimator (LSE) for Gaussian random errors in RSS values [55], which is the case in the above propagation model. The LSE (or MLE) method uses the radio propagation model for localization as follows:

14
1. Measure the distance between each of the reference nodes and the unknown node using

\[ d_{mi} = 10^{\frac{P_T - P_{r}L(\theta_i) - 10\eta}{10\eta}} \tag{2.2} \]

where, \( d_{mi} \) is the measured distance and \( P_{r}L_i \) is the mean received signal power between a given reference node \( i \) and the unknown node. Accurate distance measurement requires accurate estimation of the path loss exponent (\( \eta \)) of the environment. This can be achieved by apriori, extensive pre-configuration surveys of the localization space or by a periodic exchange of messages by the reference nodes to measure RSS values and estimate the value of \( \eta \) for the environment.

2. For each grid point location in the localization space, determine the sum of the squares of differences in the measured distances and the true Euclidean distances of all the reference nodes from the grid point.

\[ \Sigma_{(x,y)} = \sum_{i=0}^{n-1} (d_{i}^{(x,y)} - d_{mi})^2 \tag{2.3} \]

where, \( d_{i}^{(x,y)} \) is the Euclidean distance between the grid location \( (x, y) \) and the reference node \( i \).

3. Choose the grid point location with the least value of the above sum, \( \Sigma_{(x,y)} \), as the location of the unknown node.

Using the MLE method, the authors in [100] have reported a mean error of 0.5 meters for 10 reference nodes and a single unknown node, using commodity 802.11 cards. This accuracy was obtained at the cost of extensive pre-configuration studies in the localization space to determine its path loss exponent \( \eta \).

Chakrabarty et al. in [30] and Ray et al. in [84] use identity-codes to determine the location of sensor nodes in grid and non-grid sensor fields respectively. In this, each grid point or region in the localization space is identified by a unique set of reference node IDs whose signals can
reach the point or region and this unique set is an identity-code for that point or region. The two main drawbacks of this approach are that (i) in order to uniquely identify all unknown node locations in the localization space, the reference nodes need to be placed carefully according to rules determined by an optimization algorithm and that (ii) for acceptable location accuracies, the number of reference nodes required is prohibitively expensive and for sparse networks of reference nodes the accuracy is coarse-grained, in the order of radio range. For example, the number of reference nodes required to uniquely identify the location of an unknown node using identity-codes is $O(p^m)$, where $m$ is the number of dimensions of the localization space and $p$ is the number of grid points per dimension [30]. For this technique, for an experiment using 802.11b devices, the authors report a maximum error of 13 meters.

In [48], the authors propose a RF-based localization technique in which the unknown node location is determined by the intersection of all triangles, formed by reference nodes, that are likely to bound it. The unknown node determines its existence inside a triangle by comparing its measured RSS values to that of its neighbors to detect a trend in RSS values in any particular direction. This technique depends on the weak assumption that signal strength decreases monotonically with distance, which is not true in real world scenarios. In this work, the authors do not report any experimental results.

In a recent work, Maroti et al in [69] have used the relative phase-offset between two receivers of RF signals to determine the distances between the nodes, which in turn could be used for localizing the unknown node. The authors use the standard MICA 2 mote [6] radios to obtain a best error of 3 centimeters. This technique is accompanied by extensive configuration such as careful placement of reference nodes and their antennas during run-time. Also, multiple-unknown nodes are required because localization is possible only through their collaboration. Another important draw-back of this work is that the implementation presented is for the case when all the nodes (reference and unknown) are in line-of-sight of each other. This is rarely
the case in indoor environments. Also, it may be difficult to determine the phase-offsets when barriers such as walls are present between the reference nodes and the unknown node.

A very useful comparison of the performance of many finger-printing and non-finger-printing localization techniques is presented in [36]. In this, the authors conduct experiments in two office buildings over an extended period of time using commodity 802.11 cards. They present a comparative study of 6 finger-printing and 5 non-finger-printing techniques and report that no particular RF, RSS-based localization technique has a significant advantages in terms of location estimate accuracy. Using 5 access-points in a 75 meters × 48 meters area they report a median error of 3.3 meters and a 97th percentile error of 10 meters. One major reason for this performance could be the fixed number and density of access-points used in the experiments. We conjecture that the location accuracy can be improved by increasing the access-points’ number and density.

2.4 Chapter Summary

From the discussion in the chapter, a trade-off can be discerned between accuracy of location estimates and the complexity of implementation. For instance, least squares estimation techniques ([77]) require accurate RF channel parameters such as the radio path loss exponent; finger-printing based techniques (such as [15]) require extensive pre-configuration studies that depend on the features of the localization space; other techniques require complex configuration procedure ([69]). On the other extreme, really simple techniques such as computing centroid of nearby reference nodes ([26]) provide low accuracy. Therefore, this leaves the design space of easy implementation accompanied with good accuracy in wireless sensor networks open for research on new localization techniques.

Also, localization based on wireless LANs (WLANs) is disadvantaged for density of reference nodes, as compared to wireless sensor networks. Since wireless sensor networks are made up
inexpensive devices, as compared to wireless LAN access-points, they are more amenable to higher density deployments. Even though much experimental analysis has been done to study the accuracy of localization techniques using WLANs, there is no experimental analysis yet of localization techniques using wireless sensor networks. Wireless sensor networks based localization techniques could provide higher resolution compared to Wireless LAN systems for the same cost.
Chapter 3

Accurate RF Localization

3.1 Introduction

In this chapter, we present two novel RF localization techniques that are lightweight, work with any RF hardware and provide accurate localization without requiring accurate channel parameters or any pre-configuration.

In both our techniques, called Ecolocation and Sequence-Based Localization (SBL), the unknown node examines the ordered sequence of nearby reference nodes to determine its location. The main idea in our techniques is that the distance-based rank order of reference nodes constitutes a unique signature for different regions in the localization space. We name the distance-based rank order as the location sequence of the region. The location sequence can be written as a set of rank constraints, which we call location constraints, on the region represented by the location sequence.

In Ecolocation, we obtain the ordered sequence of reference nodes by ranking them on one-way RSS measurements between them and the unknown node. This measured sequence is then written as a set of constraints on the location of the unknown node. This constraint set is then compared with the ideal distance-based constraint set for each location point to determine
how many order-constraints are satisfied. The location which maximizes the number of satisfied
order-constraints is then determined to be the best estimate of the unknown node’s location.

At the heart of our second technique, SBL, is the division of a two-dimensional (2D) localization
space into distinct regions by the perpendicular bisectors of lines joining pairs of reference
nodes. We show that each distinct region formed in this manner can be uniquely identified by
a location sequence that represents the distance ranks of reference nodes to that region. We
present an algorithm to construct the location sequence table that maps all these feasible location
sequences to the corresponding regions, using the locations of reference nodes. This table
is used to localize an unknown node as follows: The unknown node first determines its own
location sequence based on the measured strength of signals between itself and the reference
nodes. It then searches through the location sequence table to determine the “nearest” feasible
sequence to its own measured sequence. The centroid of the corresponding region is taken to be
its location.

Ideally, the ranks of the reference nodes based on RSS readings should follow their ranks
based on true Euclidean distance. Of course, this is not true in the real world because of
the presence of multi-path fading and shadowing in the RF channel. Reference nodes farther
from the unknown node might measure higher RSS values than those which are closer and this
introduces errors in the constraints for Ecolocation and corrupts location sequences for SBL.

However, for Ecolocation, we show that the inherent insensitivity to absolute RSS amplitudes
and the inherent redundancy present in the set of constraints make it very robust in practice.
The name Ecolocation is derived from the phrase “Error COntrolling LOCALizaTION” because
of the close analogy to controlling errors by redundancy in traditional error control coding
techniques in digital communication systems.

In the case of SBL, for \( n \) reference nodes in the localization space, the possible number of
combinations of distance rank sequences is \( O(n^n) \). However, we prove that the actual number of
feasible location sequences is much lower due to planar geometric constraints, only \( O(n^4) \). The
lower dimensionality of the number of location sequences enables the correction of errors in the measured sequence. Thus, the high density of location sequences coupled with the robustness of rank orders in the sequences to random errors help provide good location estimate accuracy.

The rest of the chapter is organized as follows: We define location constraints, introduce Ecolocation and describe its operation in Section 3.2. We define location sequences and describe the procedure of localization using them in Section 3.3. In the same section, we derive the maximum number of feasible location sequences in the localization space, illustrate the construction of the location sequence table, discuss the effect of RF channel non-idealities on unknown node location sequences and describe metrics to measure “distance” between sequences. In Section 3.4 we compare Ecolocation with SBL. In Section 3.5, we present an exhaustive systematic performance study of our localization techniques in addition to conducting a comparative study with other state-of-the-art localization techniques. We present the evaluation of our technique in real mote experiments in Section 3.6 and summarize the chapter in Section 3.7.

### 3.2 Localization Technique I: Ecolocation

In this section we describe Ecolocation and illustrate it for the ideal and real world scenarios through examples.

The localization process is initiated by the unknown node by broadcasting a localization request. The reference nodes in the network respond to this request with response packets containing their location coordinates. The unknown node measures the signal strength (RSS) of the received packets and uses the obtained location coordinates to determine its own location as follows:

1. Determine the ordered sequence of reference nodes by ranking them on the collected RSS measurements.
2. For each possible location grid-point in the location space determine the relative ordering of reference nodes and compare it with the RSS ordering previously obtained, to determine how many of the ordering constraints are satisfied.

3. Pick the location that maximizes the number of satisfied constraints. If there is more than one such location, take their centroid.

Radio frequency (RF) based localization techniques are inherently dependent on the RF channel whose multi-path fading and shadowing effects have a fundamental bearing on the accuracy of location estimate. Nevertheless, it helps to study the localization technique in isolation of these effects. We introduce Ecolocation for the ideal scenario of zero multi-path fading and shadowing effects and latter explain why it provides robust and accurate location estimate even in the presence of these effects.

### 3.2.1 Ideal Scenario

In the absence of multi-path fading and shadowing, RSS measurements between the reference nodes and the unknown node accurately represent the distances between them. If the reference nodes are ranked as a sequence in decreasing order of these RSS values then this order represents the increasing order of their separation from the unknown node. For a reference node ranked at position $i$ in the ordered sequence,

$$R_i > R_j \Rightarrow d_i < d_j, \forall i < j$$

where, $R_i$ and $d_i$ are the RSS measurement and distance of the $i^{th}$ ranked reference node from the unknown node, respectively.

The above relationship between two reference nodes is a *constraint* on the location of the unknown node and is dependent on it. An $i^{th}$ ranked reference node forms $(i - 1)$ constraints
with lesser ranked ones and for a total of \( n \) reference nodes there are \( \frac{n(n-1)}{2} \) constraints on the unknown node.

For fixed reference node locations, the sequence order and the constraints are completely determined by the unknown node location. Figure 3.1 illustrates this idea for a simple case of four reference nodes and one unknown node.

Figure 3.1: The distance rank order of reference nodes \((A, B, C, D)\) is different for different regions \((X_1, X_2)\) in the localization space.

Table 3.1 shows the constraints on the unknown node for the example in Figure 3.1(a).

<table>
<thead>
<tr>
<th>A:1</th>
<th>B:2</th>
<th>C:3</th>
<th>D:4</th>
</tr>
</thead>
<tbody>
<tr>
<td>( R_A )</td>
<td>( R_B &lt; R_A )</td>
<td>( R_C &lt; R_A )</td>
<td>( R_D &lt; R_A )</td>
</tr>
<tr>
<td>( R_C &lt; R_B )</td>
<td>( R_D &lt; R_C )</td>
<td>( R_D &lt; R_C )</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.1: Constraints on the unknown node for the example in Figure 3.1(a).

Each location grid-point in the location space has its own set of constraints based on its Euclidean distances to the reference nodes. The unknown node location estimate can be obtained by comparing the constraints obtained from RSS measurements to the constraint sets of each location grid-point and picking the location which satisfies the maximum number of constraints. If there are more than one such locations then their centroid is the location estimate.
3.2.2 Real World Scenario

In contrast to the ideal scenario, the real world is characterized by the presence of multi-path fading and shadowing in the RF channel. Ideally, reference nodes that are far from the unknown node should measure lower RSS values than reference nodes that are nearer, but due to multi-path effects this is not true in the real world.

Figure 3.2 shows the experimental RSS measurements at five MICA 2 receivers placed at different distances from a MICA 2 transmitter. It shows that the receiver at 5.69 meters measured a higher RSS value than the receiver at 5.37 meters. Evidently, RSS measurements do not represent distances accurately in the real world.

Therefore, if the reference nodes are ranked on their respective RSS measurements, the constraints on the unknown node location formed by these ranks will be erroneous. For example, if the ranks of third and fourth ranked reference nodes are interchanged due to multi-path effects in the RF channel, as in the experiment of Figure 3.2, for the example in Figure 3.1(a), then the new constraints are as shown in Table 3.2. As it can be seen, 10% of the constraints are erroneous in this case.

Figure 3.2: Real world experimental results: Reference nodes far from the unknown node may measure higher RSS values than closer reference nodes. Note that y-axis is reverse ordered.
Table 3.2: Constraints for the example of Table 3.1 when the ranks of third and fourth ranked reference nodes are interchanged due of multi-path effects.

The percentage of erroneous constraints depends on the RF channel condition, the topology of the reference nodes and the number of reference nodes. The unknown node location estimate accuracy in turn depends on the percentage of erroneous constraints.

Next, we discuss the implementation aspects of Ecolocation.

3.2.3 Location Determination

For ease of implementation, the constraint set is represented by a constraint matrix \( M_{n \times n} \), where

\[
M_{n \times n}(i, j) = \begin{cases} 
1 & \text{if } R_i < R_j \\
0 & \text{if } R_i = R_j \\
-1 & \text{if } R_i > R_j 
\end{cases}
\]

It is easy to see that \( M_{n \times n} \) is a skew-symmetric matrix and each element of the matrix represents a constraint in the constraint set. The pseudo code for the Ecolocation algorithm is presented below.

---

**Algorithm 1. ECOLOCATION**

Input: The number of reference nodes within the range of the unknown node \( n \), their locations \( (ax_i, ay_i)(i = 1 \ldots n) \), the RSS values of RF signals from the unknown node at each one of them \( R_i(i = 1 \ldots n) \), the localization area size \( (S \times S \text{ sq. length units}) \), and the area scanning resolution \( r \).
Output: The location estimate of the unknown node.

The reference nodes are sorted into an ordered sequence based on $R_i$'s and a constraint matrix $M_{n \times n}$ is derived from this sequence.

Calculate the number of matched constraints at each grid point $(i, j)$ and identify the maximum number of constraints matched over all the grid points.

0 $\text{maxConstrMatch} \leftarrow 0$;
1 for each grid point $(i, j)$ in the localization area
2   for each reference node $k (\rightarrow 1 \ldots n)$
3      $d_{ij}^k \leftarrow ((ax_k - i)^2 + (ay_k - j)^2)^{\frac{1}{2}}$;
4   generate constraint matrix $C_{n \times n}^{ij}$ based on $d_{ij}^k$.
5 for each element $(m, n) (n > m)$ in $C_{n \times n}^{ij}$
6   if $C_{n \times n}^{ij}(m, n) = M_{n \times n}(m, n)$
7      $\text{constrMatch}^{ij} \leftarrow \text{constrMatch}^{ij} + 1$;
8   else
9      $\text{constrMatch}^{ij} \leftarrow \text{constrMatch}^{ij} - 1$;
10   if $\text{constrMatch}^{ij} > \text{maxConstrMatch}$
11      $\text{maxConstrMatch} \leftarrow \text{constrMatch}^{ij}$;

Search for grid points where the maximum number of constraints are matched and return the centroid of those grid points as the location estimate.

12 $(eco_x, eco_y) \leftarrow (0, 0)$;
13 $\text{count} \leftarrow 0$;
14 for each grid point $(i, j)$
15   if $\text{constrMatch}^{ij} = \text{maxConstrMatch}$
16      $(eco_x, eco_y) \leftarrow (eco_x + i, eco_y + j)$;
17   $\text{count} \leftarrow \text{count} + 1$;
Theorem 1. Ecolocation takes at most $O\left(\frac{S^2n^2}{r^2}\right)$ time and $O\left(\frac{S^2}{r^2} + n^2\right)$ space to determine the location of the unknown node.

Proof. We should say first of all that this implementation of Ecolocation is meant only to be functional, it is not at all optimized for space or time complexity. Still, the following analysis provides an upper bound on the computational costs for implementing this technique. The initial sorting of reference nodes based on $R'_i$'s costs $\Theta(n \log(n))$ time and $O(n)$ space respectively. The corresponding constraint matrix generation costs $O(n^2)$ time and $O(n^2)$ space respectively. Calculating the number of constraints matched at each grid point and identifying the maximum number of constraints matched over all grid points (lines 1-11) costs $O\left(\frac{S^2n^2}{r^2}\right)$ time and $O\left(\frac{S^2}{r^2} + n^2\right)$ space respectively. Searching for grid points where maximum number of constraints are matched (lines 12 - 17) costs $O\left(\frac{S^2}{r^2}\right)$ time and $O(1)$ extra space. Finally, calculating the centroid of those grid points (line 18) costs $O(1)$ time and space. In total, the time and space complexities of Ecolocation are at most $O\left(\frac{S^2n^2}{r^2}\right)$ and $O\left(\frac{S^2}{r^2} + n^2\right)$ respectively.

3.2.4 Examples

Figure 3.3 shows a sample layout of nine reference nodes placed in a grid and a single unknown node. Figure 3.3(a) plots the location estimate for the ideal case when there are no erroneous constraints on the unknown node. Figures 3.3(b), 3.3(c) and 3.3(d) show the location estimates for varying percentages of erroneous constraints. The location estimate error increases with increasing percentage of erroneous constraints.

These examples suggest that Ecolocation is robust to multi-path effects of the RF channel up to some level. The inherent redundancy in the constraint set ensures that the non-erroneous
constraints help in estimating the unknown node location accurately. Also, the constraint construction inherently holds true for random variations in RSS measurements up to a tolerance level of (|R_i - R_j|).

Through the above examples, we have shown a proof of concept that localization using geometric constraints is robust to RF channel non-idealities. Next we conduct a deeper investigation into the geometric meanings of location constraints and location sequences and propose a localization technique based on location sequences.

3.3 Localization Technique II: Sequence-Based Localization

In this section, first, we discuss a deeper investigation of location sequences and then describe the procedure to use them for localization.
3.3.1 Location Sequences

Assume that a 2D localization space consists of \( n \) reference nodes. Consider any two reference nodes and draw a perpendicular bisector to the line joining their locations. This perpendicular bisector divides the localization space into three different regions that are distinguished by their proximity to either of the reference node, as illustrated in Figure 3.4(a). Similarly, if perpendicular bisectors are drawn for all \( \frac{n(n-1)}{2} \) pairs of reference nodes, they divide the localization space into many regions of three different types - vertices, edges and faces, as shown in Figure 3.4(b).

This subdivision of a 2D space into vertices, edges and faces by a set of lines is an arrangement induced by that set [32].

Now, for each region created by the arrangement induced by the set of perpendicular bisectors, determine the ordered sequence of reference nodes' ranks based on their distances from them. We define this ordered sequence of distance ranks as the location sequence. Consider the following theorem.

**Theorem 2.** The location sequence of a given region is unique to that region.

**Proof.** The proof is by contradiction. Assume that two different regions in the arrangement have the same location sequence. This implies that the distance ranks of reference nodes are the
same for both the regions. This further implies that there is no bisector line that separates the two regions. The implication applies to all possible combinations of regions such as two faces, two edges, two vertices, a face and an edge, an edge and a vertex and a face and a vertex, in their own different ways. Otherwise, if there was a bisector line of two arbitrary reference nodes that separated the two regions then it would rank those reference nodes differently for the two regions. But this is a contradiction, as by definition, two different regions in the arrangement are separated by at least a single bisector line.

Therefore, each region created by the arrangement has a unique location sequence. Further, we make the following observations:

- All locations inside a region have the same location sequence.

- If each region in the arrangement is represented by its centroid, there is a one-to-one mapping between a location sequence and the centroid of the region it represents. For a vertex, the centroid is the vertex itself; for an edge, the centroid is its midpoint and for a face, the centroid is the centroid of the polygon that bounds it.

- The total number of unique location sequences is equal to the sum of the number of vertices, the number of edges and the number of faces created by the arrangement in the localization space.

The order in which the ranks of reference nodes are written in a location sequence is determined by a pre-defined order of reference node IDs. We illustrate the above ideas through examples. Figure 3.5(a) shows the location sequences four different regions. In the example the pre-defined order of reference node IDs is ABCD. Region 1 is a face and its location sequence is 1234, since the distance rank of A from it is 1 (A is the closest) and the respective distance ranks of B, C and D are 2, 3 and 4 (D is the farthest). Similarly, for Region 3 the location sequence is 4321 as the distance rank of A is the farthest (distance rank 4), D is the closest (distance rank...
1) and B is closer than C and A. For Region 4, which is a vertex, the distance ranks of A,B and C,D are equal in pairs as it lies on the intersection of perpendicular bisectors of those pairs of reference nodes. Also, the pair C,D is closer to it than the pair A,B. Therefore, its location sequence is 3311. Similarly, for Region 2, which is an edge, the distance ranks of A and B are the same and its location sequence is 1134. Figure 3.5(b) shows all feasible location sequences for the topology of reference nodes of Figure 3.5(a).

3.3.2 Localization Procedure

The procedure for localization of unknown nodes using location sequences is as follows:

1. Determine all feasible location sequences in the localization space and list them in a location sequence table.

2. Determine the location sequence of the unknown node location using received signal strength (RSS) measurements of localization response packets obtained from the reference nodes. The RSS based location sequence will be a corrupted version of the original location sequence.
3. Search in the location sequence table for the “nearest” location sequence to the unknown
node location sequence. The centroid mapped to by that sequence is the location estimate
of the unknown node.

The above procedure opens itself to the following questions: How many feasible location
sequences are there in a 2D localization space? How can we get them? How do random errors
in RSS measurements affect the unknown node location sequence? What is the meaning of
“nearest” location sequence and how do we measure distances between location sequences?

In the rest of this section we answer the above questions. We begin by determining the
maximum number of feasible location sequences in the localization space.

3.3.3 Maximum Number of Location Sequences

For $n$ reference nodes in the localization space, the number of possible combination sequences of
distance ranks is $O(n^n)$. However, we show that the actual number of feasible location sequences
is much lower, in the order of $O(n^4)$ at worst.

As stated previously, the number of feasible location sequences is equal to the sum of the
number of vertices, edges and faces created by the arrangement induced by the perpendicular
bisectors of reference nodes. Therefore, its upper bound can be obtained by determining the
maximum number of such vertices, edges and faces, given the locations of the reference nodes. In
[32], the authors show that the maximum number of vertices, edges and faces for an arrangement
induced by $n$ lines is $\frac{n(n-1)}{2}$, $n^2$ and $\frac{n^2}{2} + \frac{n}{2} + 1$ respectively. Using these results, for $\frac{n(n-1)}{2}$
perpendicular bisectors of $n$ reference nodes,

1. The number of vertices is at most $\frac{n^4}{8} - \frac{n^3}{4} - \frac{n^2}{8} + \frac{n}{4}$.

2. The number of edges is at most $\frac{n^4}{4} - \frac{n^3}{2} + \frac{n^2}{4}$.

3. The number of faces is at most $\frac{n^4}{8} - \frac{n^3}{4} + \frac{3n^2}{8} - \frac{n}{4} + 1$. 

32
Owing to the properties of perpendicular bisectors, it is possible to derive tighter upper bounds on the number of vertices, edges and faces.

**Theorem 3.** Let \( L \) be the set of bisector lines for \( n \) reference nodes, \( |L| = \frac{n(n-1)}{2} \). Let \( A(L) \) be the arrangement induced by \( L \). Then,

1. The number of vertices of \( A(L) \) is at most \( \frac{n^4}{8} - \frac{7n^3}{12} + \frac{7n^2}{8} - \frac{5n}{12} \).

2. The number of edges of \( A(L) \) is at most \( \frac{n^4}{4} - n^3 + \frac{7n^2}{4} - n \).

3. The number of faces of \( A(L) \) is at most \( \frac{n^4}{8} - \frac{5n^3}{12} + \frac{7n^2}{8} - \frac{7n}{12} + 1 \).

**Proof.** We make use of the property that the perpendicular bisectors of the sides of a triangle intersect at a single point. Assume that \((i-1)\) reference nodes have already been added, implying that the localization space already has \( \frac{(i-1)(i-2)}{2} \) bisector lines. When the \( i^{th} \) reference node is added, \((i-1)\) new bisector lines are added to the localization space.

**Vertices:** The first of the \((i-1)\) bisector lines intersects the already present lines in at most \( \frac{(i-1)(i-2)}{2} \) new vertices. The second new line is the perpendicular bisector of a side of the triangle in which the first new line is also a perpendicular bisector. Therefore, the second new line has to pass through at least one of the vertices created by the first new line, thus creating at most \( \frac{(i-1)(i-2)}{2} - 1 \) new vertices. Similarly the third new line creates at most \( \frac{(i-1)(i-2)}{2} - 2 \) new vertices. This is illustrated in Figure 3.6 for \( n = 4 \). Finally the \((i-1)^{th}\) new line creates at most \( \frac{(i-1)(i-2)}{2} - (i-2) \) new vertices. Therefore, the total number of new vertices added by the \( i^{th} \) reference node is at most

\[
\frac{(i-1)(i-2)}{2} + \frac{(i-1)(i-2)}{2} - 1 + \frac{(i-1)(i-2)}{2} - 2 + \cdots + \frac{(i-1)(i-2)}{2} - (i-2)
\]

\[
= (i-1) \frac{(i-1)(i-2)}{2} - (1 + 2 + \cdots + (i-2)) = (i-1) \frac{(i-1)(i-2)}{2} - \frac{(i-2)(i-1)}{2}
\]

\[
= \frac{(i-1)(i-2)^2}{2}
\]
Figure 3.6: Addition of fourth reference node $D$ adds 3 new bisector lines to the localization space. (a) The first of the 3 new bisector lines, line 1, the perpendicular bisector of $CD$, creates 3 new vertices (equal to the number of pre-existing lines in the localization space), 4 new faces and 7 new edges at most. (b) The second line, line 2, the perpendicular bisector of $BD$, has to pass through the intersection point of the bisectors of $CD$ and $BC$ because, $\{BD, CD, BC\}$ form a triangle and the perpendicular bisectors of the three sides of a triangle intersect at a single point. Therefore line 2 creates 2 new vertices, 4 new faces and 6 new edges at most. (c) Similarly, line 3, the perpendicular bisector of $AD$ has to pass through the intersection points of perpendicular bisectors of $AB$, $BD$ and $AC$, $CD$ as $\{AD, AB, BD\}$ and $\{AD, AC, CD\}$ are two triangles with a common side $AD$. Therefore, line 3 creates 1 new vertex, 4 new faces and 5 new edges at most.

The maximum number of vertices for $n = 3$ is 1. Therefore, for $n$ reference nodes, the maximum number of vertices is

\[
1 + \sum_{i=4}^{n} \frac{(i-1)(i-2)^2}{2} = 1 + \sum_{i=4}^{n} \left[ \frac{i^3}{2} - \frac{5i^2}{2} + 4i - 2 \right] = \sum_{i=1}^{n} \left[ \frac{i^3}{2} - \frac{5i^2}{2} + 4i - 2 \right] = \frac{n^4}{8} - \frac{7n^3}{12} + \frac{7n^2}{8} - \frac{5n}{12} \tag{3.4}
\]

Edges: As explained previously, the first new line intersects the already present lines in at most $\frac{(i-1)(i-2)}{2}$ vertices and creates at most $\frac{(i-1)(i-2)}{2} + 1$ new edges on the new line and at most $\frac{(i-1)(i-2)}{2}$ new edges on the old lines which add up to $\frac{(i-1)(i-2)}{2} \cdot 2 + 1$ new edges at most. Since the second new line passes through at least one of the vertices created by the first new line, it creates at most $\frac{(i-1)(i-2)}{2} + 1$ new edges on the second new line and it creates at most $\frac{(i-1)(i-2)}{2} - 1$ new edges on the old lines including the first new line. This adds up to at most $\frac{(i-1)(i-2)}{2} \cdot 2$ new
edges in the localization space. This trend is again illustrated in Figure 3.6 for four reference
nodes in the localization space. Finally, the \((i-1)^{th}\) new line adds \(\frac{(i-1)(i-2)}{2} \cdot 2 - (i-3)\) new
edges to the localization space. Therefore, the total number of new edges added by the \(i^{th}\)
reference node is at most

\[
\frac{(i-1)(i-2)}{2} \cdot 2 + 1 + \frac{(i-1)(i-2)}{2} \cdot 2 - 1 + \cdots + \frac{(i-1)(i-2)}{2} \cdot 2 - (i-3) \quad (3.6)
\]

\[
= 2 \cdot (i-1) \left( \frac{i-1}{2} \right) - 1 - (1 + 2 + \cdots + (i-3)) \quad (3.7)
\]

\[
= 1 + (i-1)^2(i-2) - \frac{(i-3)(i-2)}{2} \quad (3.8)
\]

\[
= i^3 - \frac{9i^2}{2} + \frac{15i}{2} - 4 \quad (3.9)
\]

The maximum number of edges for \(n = 3\) is 6. Therefore, for \(n\) reference nodes, the maximum
number of edges is

\[
6 + \sum_{i=4}^{n} \left[ i^3 - \frac{9i^2}{2} + \frac{15i}{2} - 4 \right] = \sum_{i=1}^{n} \left[ i^3 - \frac{9i^2}{2} + \frac{15i}{2} - 4 \right] = n^4 - n^3 + \frac{7n^2}{4} - n \quad (3.10)
\]

**Faces:** The number of new faces created by a new line is equal to the number of edges on the
new line. Therefore, the number of new faces created by the first new line among the \((i-1)\)
new lines is at most \(\frac{(i-1)(i-2)}{2} + 1\). Since the second new line has to pass through one of the
intersection points of the first line, it would also create \(\frac{(i-1)(i-2)}{2} + 1\) new faces and this trend
continues for all the \((i-1)\) new lines as illustrated in Figure 3.6. Therefore, the total number
of new faces added by the \(i^{th}\) reference node is at most

\[
(i-1) \left( \frac{(i-1)(i-2)}{2} + 1 \right) \quad (3.11)
\]

35
The localization space has one face when $n = 1$. Therefore, for $n$ reference nodes the maximum number of faces in the localization space is given by:

$$1 + \sum_{i=2}^{n} (i-1) \left( \frac{(i-1)(i-2)}{2} + 1 \right) = \frac{n^4}{8} - \frac{5n^3}{12} + \frac{7n^2}{8} - \frac{7n}{12} + 1 \quad (3.12)$$

**Corollary 1.** The maximum number of unique location sequences due to $n$ reference nodes is

$$\frac{n^4}{2} - 2n^3 + 7n^2 - 2n + 1.$$  

*Proof.* The maximum number of unique location sequences is the sum of the maximum number of vertices, edges and faces due to $n$ reference nodes, derived in Theorem 3.

$$\left( \frac{n^4}{8} - \frac{7n^2}{8} + \frac{5n}{12} \right) + \left( \frac{n^4}{4} - n^3 + \frac{7n^2}{4} - n \right) + \left( \frac{n^4}{8} - \frac{5n^3}{12} + \frac{7n^2}{8} - \frac{7n}{12} + 1 \right) \quad (3.13)$$

$$= \frac{n^4}{2} - 2n^3 + \frac{7n^2}{2} - 2n + 1 \quad (3.14)$$

Next, we illustrate how to obtain all these feasible location sequences in the localization space and store them in the location sequence table.

### 3.3.4 Location Sequence Table Construction

Below, we present the pseudo-code for an algorithm that constructs the location sequence table given the locations of the reference nodes and the boundaries of the localization space.
Algorithm 2. ConstructLocationSequenceTable$^1$.

Input:

1. Location coordinates of reference nodes, $\{(ax_i, ay_i) \mid i = 0 \rightarrow n - 1\}$.

2. Boundaries of the localization space $B$.

Output: Location Sequence Table.

\[
L = \{l_i \mid i = 0 \rightarrow (\text{\textfrac{n(n-1)}{2}} - 1)\} \leftarrow \text{BisectorLines}(\{(ax_i, ay_i) \mid i = 0 \rightarrow n - 1\}, B)
\]

\[
(FL, EL, VL) \leftarrow \text{ConstructArrangement}(L)
\]

\[\text{Get vertex sequences.}\]

\[
\text{for } i \leftarrow 0 \text{ to } (|VL| - 1)
\]

\[
\text{Centroid}[i] \leftarrow VL[i]
\]

\[
\text{Sequence}[i] \leftarrow \text{GetSequence}(\text{Centroid}[i])
\]

\[\text{Get edge sequences.}\]

\[
\text{for } i \leftarrow |VL| \text{ to } (|VL| + |EL| - 1)
\]

\[
\text{Centroid}[i] \leftarrow \text{GetEdgeCentroid}(EL[i])
\]

\[
\text{Sequence}[i] \leftarrow \text{GetSequence}(\text{Centroid}[i])
\]

\[\text{Get face sequences.}\]

\[
\text{for } i \leftarrow (|VL| + |EL|) \text{ to } (|VL| + |EL| + |FL| - 1)
\]

\[
\text{Centroid}[i] \leftarrow \text{GetFaceCentroid}(FL[i])
\]

\[
\text{Sequence}[i] \leftarrow \text{GetSequence}(\text{Centroid}[i])
\]

$^1$C++ code files that construct the arrangement of lines and the location sequence table are available for download at http://anrg.usc.edu/downloads.html
BISECTORLINES takes in the locations of the reference nodes and the boundaries of the localization space as input and returns the set \( L \) of all pair-wise perpendicular bisector lines within the boundaries of the localization space. Each line is represented by the intersection points on the left and right boundaries of the localization space.

CONSTRUCTARRANGEMENT constructs the arrangement given a set of lines as input and returns a doubly connected edge list that consists of a vertex list \( (V_L) \), an edge list \( (E_L) \) and a face list \( (F_L) \). Please refer to the book [32], Chapter 8, Section 3 for a detailed description of this algorithm.

• Vertex List, \( V_L \): Contains pointers to all vertices of the arrangement induced by the set \( L \).

• Edge List, \( E_L \): Contains pointers to all edges of the arrangement induced by the set \( L \).

• Face List, \( F_L \): Contains pointers to all faces of the arrangement induced by the set \( L \).

GETEDGECENTROID takes in an edge pointer as the input and returns the centroid of the edge. The centroid of an edge \((c_x, c_y)\) is its mid point given by:

\[
(c_x, c_y) \leftarrow \left( \frac{o_x + d_x}{2}, \frac{o_y + d_y}{2} \right)
\]

where, \((o_x, o_y)\) and \((d_x, d_y)\) are the origin and destination vertices of the edge.
• GetFaceCentroid takes in a face pointer as the input and returns the centroid of the face. The centroid of a face \((c_x, c_y)\), given its vertices \(\{(x_i, y_i) | 0 \leq i \leq p - 1\}\), is calculated as follows:

\[
c_x \leftarrow \frac{1}{6A} \sum_{i=0}^{p-1} (x_i + x_{i+1})(x_iy_{i+1} - x_{i+1}y_i)
\]

\[
c_y \leftarrow \frac{1}{6A} \sum_{i=0}^{p-1} (y_i + y_{i+1})(x_iy_{i+1} - x_{i+1}y_i)
\]

where, \(p\) is the number of vertices that bound a given face and \(A\) is its area given by

\[
A \leftarrow \frac{1}{2} \sum_{i=0}^{p-1} (x_iy_{i+1} - x_{i+1}y_i) ; (x_p, y_p) = (x_0, y_0)
\]

• GetSequence takes in the coordinates of a point in the localization space and returns the location sequence for that point with respect to the locations of the reference nodes.

Theorem 4. Algorithm 2 takes \(O(n^5 \log(n))\) worst-case time and \(O(n^5)\) worst case space to construct the location sequence table.

Proof. The function BisectorLines in line 0 takes \(O(n^2)\) time and space. The algorithm ConstructArrangement that constructs the arrangement of lines takes \(O(n^4)\) time, which is optimal, as proven in Theorems 8.5 and 8.6 of [32]. Since this algorithm returns the vertex list \(VL\), the edge list \(EL\) and the face list \(FL\), it requires \(O(n^4)\) space to store all the three lists. The functions GetFaceCentroid and GetEdgeCentroid in lines 3 and 7 respectively take \(O(1)\) time and space each. The function GetSequence involves sorting \(n\) reference nodes based on their distances from the centroid of the region in consideration. This takes \(O(n \log n)\) time and \(O(n)\) space. Since the number of faces, edges and vertices is \(O(n^4)\) the worst case time requirement for lines 2-13 in the above algorithm is \(O(n^5 \log(n))\) and the worst case space
requirement is $O(n^5)$. Therefore, in total, Algorithm 2 takes $O(n^5 \log(n))$ worst-case time and $O(n^5)$ worst case space to construct the location sequence table.

Table 3.3 compares simulation results for the number of location sequences obtained using the above algorithm with analytical values from Corollary 1. The simulation results are gathered over 1000 random trials (with 100 different random seeds) in each of which $n$ reference nodes were placed uniformly at random in the localization space. From the last two columns of the table it can be seen that the simulation results match the analytical results very closely. Note that for higher number of reference nodes the probability of occurrence of the arrangement that would produce the maximum of location sequences is less than 1 in 1000 i.e., 0.001. Also, for increasing number of reference nodes, the average number of location sequences is increasingly smaller than the maximum number. Next, we discuss the effect of RF channel random errors on the unknown node location sequence.

<table>
<thead>
<tr>
<th>Number of Reference Nodes ($n$)</th>
<th>Number of Bisector Lines ($\frac{n(n-1)}{2}$)</th>
<th>Average Number of Location Sequences (Simulations)</th>
<th>Minimum Number of Location Sequences (Simulations)</th>
<th>Maximum Number of Location Sequences (Simulations)</th>
<th>Maximum Number of Location Sequences (Analytical)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3</td>
<td>12.3</td>
<td>7</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>44.0</td>
<td>23</td>
<td>49</td>
<td>49</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>117.3</td>
<td>51</td>
<td>141</td>
<td>141</td>
</tr>
<tr>
<td>6</td>
<td>15</td>
<td>274.8</td>
<td>217</td>
<td>331</td>
<td>331</td>
</tr>
<tr>
<td>7</td>
<td>21</td>
<td>548.4</td>
<td>441</td>
<td>653</td>
<td>673</td>
</tr>
<tr>
<td>8</td>
<td>28</td>
<td>988.6</td>
<td>840</td>
<td>1147</td>
<td>1233</td>
</tr>
<tr>
<td>9</td>
<td>36</td>
<td>1663.9</td>
<td>1447</td>
<td>1881</td>
<td>2089</td>
</tr>
<tr>
<td>10</td>
<td>45</td>
<td>2630.2</td>
<td>2321</td>
<td>2933</td>
<td>3331</td>
</tr>
</tbody>
</table>

Table 3.3: Progression of number of location sequences with number of reference nodes ($n$) in the localization space. The last two columns compare the simulation and analytical results for the maximum number of location sequences. Simulation results are gathered from 1000 random trials (with 100 different random seeds) in each of which $n$ reference nodes were placed uniformly at random in a square localization space.
3.3.5 Unknown Node Location Sequence

The unknown node determines its location sequence using RSS measurements of RF localization packets exchanged between itself and the reference nodes. The RSS measurements are subjected to random errors due to RF channel non-idealities such as multi-path and shadowing. In the absence of such non-idealities, the RSS measurements accurately represent the distances between the unknown node and the reference nodes. If reference nodes are ranked in a decreasing order of these RSS values then this order represents the increasing order of their separation from the unknown node.

This is not true in reality. Reference nodes that are farther from the unknown node might measure higher RSS values than reference nodes that are closer. If the reference nodes are ranked on their respective RSS measurements, the location sequence formed by these ranks will be a corrupted version of the original sequence. Corruption in unknown node location sequence results in erroneous estimation of its location. In the ideal case, when there is no corruption, the unknown node location would be the centroid of the region represented by its location sequence. However, corruption in its location sequence could erroneously estimate its location to be the centroid of some other region.

For example, if the ranks of reference nodes C and D are interchanged because of corruption due to RF channel non-idealities for Region 1 of Figure 3.5(a), the new location sequence would be 1243 instead of 1234. And 1243 represents a region that is adjacent to the original region as shown in Figure 3.5(b).

3.3.6 Feasible and Infeasible Sequences

As discussed previously, combinatorially, \( n \) reference nodes produce \( O(n^n) \) location sequences. But as shown in the previous section, a localization space with \( n \) reference nodes has only \( O(n^4) \) distinct regions and consequently only \( O(n^4) \) feasible location sequences in the worst
case. For given reference node locations, the location sequence table includes all feasible location sequences. All other sequences are infeasible. The non-idealities of the RF channel could corrupt a feasible location sequence either to another feasible sequence or an infeasible sequence as illustrated in Figure 3.7. If the corrupted sequence is infeasible, then it would be possible to detect the corruption in the sequence, whereas, if the corrupted sequence is feasible, corruption detection is not possible.

Here, we would like to emphasize the importance of low density of location sequences compared to the full sequence space. The low density of location sequences implies that many infeasible sequences are mapped to a single feasible sequence and this in turn could provide robustness to location estimation against RF channel non-idealities.

![Sequence space of size $O(n^4)$](image)

Corruption due to wireless channel non-idealities

Space of feasible location sequences (Size: $O(n^4)$)

Space of infeasible location sequences

Figure 3.7: RF channel non-idealities could corrupt a location sequence from the feasible space either to another sequence in the feasible space or to a sequence in the infeasible space.

Next, we present metrics to measure distance between two location sequences.

### 3.3.7 Distance Metrics

The distance between two location sequences is essentially the difference in rank orders of different reference nodes. Fortunately, statistics [78] offers two metrics that capture this difference in rank orders - Spearman’s Rank Order Correlation Coefficient, and Kendall’s Tau.
Given two location sequences \( U = \{u_i\} \) and \( V = \{v_i\} \), \( 1 \leq i \leq n \), where \( u_i \)'s and \( v_i \)'s are the ranks of reference nodes, the above two metrics are defined as follows.

1. **Spearman’s Rank Order Correlation Coefficient** [78]: It is defined as the linear correlation coefficient of the ranks and is given by

\[
\rho = 1 - \frac{6 \sum_{i=1}^{n} (u_i - v_i)^2}{n(n^2 - 1)} \tag{3.19}
\]

2. **Kendall’s Tau** [78]: In contrast to Spearman’s coefficient in which the correlation of exact ranks is calculated, this metric calculates the correlation between the relative ordering of ranks of the two sequences. It compares all the \( \frac{n(n-1)}{2} \) possible pairs of ranks \( (u_i, v_i) \) and \( (u_j, v_j) \) to determine the number of matching and non-matching pairs. A pair is matching or concordant if \( u_i > u_j \Rightarrow v_i > v_j \) or \( u_i < u_j \Rightarrow v_i < v_j \) and non-matching or discordant if \( u_i > u_j \Rightarrow v_i < v_j \) or \( u_i < u_j \Rightarrow v_i > v_j \). The correlation between the two sequences is calculated as follows:

\[
\tau = \frac{(n_c - n_d)}{\sqrt{n_c + n_d + n_{tu}} \sqrt{n_c + n_d + n_{tv}}} \tag{3.20}
\]

where, \( n_c \) is the number of concordant pairs, \( n_d \) is the number of discordant pairs, \( n_{tu} \) is the number of ties in \( u \)'s and \( n_{tv} \) is the number of ties in \( v \)'s.

The range of both \( \rho \) and \( \tau \) is \([-1, 1]\). Next, we describe the procedure to determine locations of unknown nodes using their location sequences.

### 3.3.8 Location Determination

The location of the unknown node is determined as follows:

1. Calculate distances between the unknown node location sequence and all location sequences in the location sequence table using the above distance metrics.
2. Choose the centroid represented by the location sequence that is closest to the unknown node location sequence as its location estimate.

Mathematically,

\[
\text{LocationEstimate} = \text{Centroid}(\arg \min_{1 \leq i \leq O(n^4)} \tau_i) \quad (3.21)
\]

where, \(\tau_i\) is the Kendall’s Tau or Spearman’s correlation between the unknown node location sequence and the \(i^{th}\) location sequence in the location sequence table.

Due to RF channel non-idealities, the unknown node location sequence could be a feasible sequence different from its uncorrupted version or an infeasible sequence. In any case, the above procedure maps it to the centroid of the nearest feasible location sequence in the location sequence table that represents a different region in the arrangement than the original uncorrupted version.

We measure the amount of corruption in the unknown node location sequence by calculating its distance from the uncorrupted version, using the above metrics, and denote it by \(T\). We denote the distance between the unknown node location sequence and the nearest feasible sequence in the location sequence table by \(\tau\).

Calculating the Spearman’s coefficient and Kendall’s Tau between two sequences are \(O(n)\) and \(O(n^2)\) operations respectively. Since the location sequence table is of size \(O(n^4)\), searching through it takes \(O(n^5)\) and \(O(n^6)\) operations respectively for the above two metrics. Later in the chapter, in Section 3.5, we compare the performance of the two distance metrics in terms of error in the unknown node location estimate.

### 3.3.9 Localization Scenarios

Here, we illustrate two localization procedures for two different scenarios that are determined by the localization space size.
1. *Entire localization space is within the radio range of the unknown node:* In this case, the location sequence table remains constant for all locations of the unknown node in the localization space. Therefore, the localization procedure is as follows:

(a) Pre-construct and store the location sequence table using the locations of the reference nodes.

(b) When the unknown node initiates the localization process by broadcasting a localization packet, provide the stored location sequence table along with the RSS measurements from the reference nodes.

(c) The unknown node determines its location sequence using the RSS measurements and determines its location by searching through the provided location sequence table for the nearest feasible location sequence.

Here, the time cost incurred by the unknown node to estimate its location is equal to the sum of the time to determine its location sequence, an $O(n \log n)$ operation, and the time to search through the location sequence table, a $O(n^6)$ operation. The amount of memory space required is of the order of $O(n^5)$ bytes.

2. *Localization space is much larger than the radio range of the unknown node:* In this case, the location sequence table changes with the location of the unknown node as a different set of reference nodes are encountered at each location. Therefore, the localization procedure is as follows:

(a) The unknown node collects the locations and RSS measurements of the reference nodes in its radio range.

(b) It constructs the location sequence table, using Algorithm 2, using the locations of the reference nodes and calculates its location sequence using the RSS measurements.
(c) It determines its location by searching for the nearest sequence in the location sequence table.

In this case, the time cost incurred by the unknown node to estimate its location is equal to the sum of the time to calculate its location sequence, an $O(n \log n)$ operation, the time to construct the location sequence table, an $O(n^5 \log n)$ operation, and the time to search through it, a $O(n^6)$ operation. The memory requirement is $O(n^5)$ in this case also.

A wireless device that is typically used as an unknown node is of the form factor of an IPAQ [2] (that can communicate with the reference node devices, usually of the form factor of Berkeley MICA 2 motes [6]) which typically has a 300MHz processor and 128MB of RAM. In real application scenarios, a typical value for the number of reference nodes ($n$) is less than 15 after which there is only very marginal gain in location accuracy of the unknown node. Therefore, for a typical value of $n = 10$ reference nodes, the time and space requirements for the unknown node to construct the location sequence table are approximately 0.3 milliseconds and 32 KB respectively. And the time required to search through it is approximately 0.4 milliseconds. Thus, including the associated overhead, the total localization time taken by sequence-based localization is in milliseconds in typical application scenarios, which is very efficient. Next, we illustrate the robustness of our localization technique to RF channel non-idealities through some examples.

### 3.3.10 Examples

Figure 3.8 shows the sample layout of nine reference nodes placed in a grid and a single unknown node (P) considered in Figure 3.3. Figure 3.8(a) plots the location estimate (E) for the ideal case when there are no erroneous ranks i.e., the location sequence is uncorrupted or $T = 1$. In these examples we use Kendall’s Tau to measure the distance between sequences. Figures 3.8(b), 3.8(c) and 3.8(d) show the location estimates for increasing corruption in unknown node location
Figure 3.8: Robustness examples: Location estimate (E) for the unknown node (P) at (1, 3) for a grid layout of 9 reference nodes. The number adjacent to a reference node is its corresponding rank. The location error is expressed in meters where the side length of the square localization area is 12 meters. (a) \((T = 1, \tau = 1)\), Estimate (E): (1.33, 1.33), Location Error: 0.46 meters (b) \((T = 0.722, \tau = 0.783)\), Estimate (E): (2.0, 2.0), Location Error: 1.4 meters (c) \((T = 0.556, \tau = 0.667)\), Estimate (E): (2.0, 2.0), Location Error: 1.4 meters (d) \((T = 0.111, \tau = 0.278)\), Estimate (E): (2.0, 1.33), Location Error: 1.94 meters.

sequences. Similar to the example of Figure 3.3, the location estimate error increases with increasing corruption or decreasing correlation, \(T\), between the RSS location sequence and the true location sequence of P. These examples suggest that sequence-based localization is robust to multi-path and shadowing effects of the RF channel up to some level. Intuitively, the three main reasons to which this robustness can be attributed to are:

1. The low density, \(O(n^4)\), of location sequence space relative to the entire sequence space of \(O(n^n)\).

2. The inherent redundancy of comparing \(\frac{n(n-1)}{2}\) rank pairs in calculating the distance between two sequences using Kendall’s Tau.

3. The rank order in the location sequence of the unknown node due to two reference nodes with RSS readings \(R_i\) and \(R_j\) is robust to random errors in them up to a tolerance level of \(|R_i - R_j|\).

Having presented two localization techniques, one based on location constraints called Ecolocation and the other based on location sequences called SBL, we investigate their difference and similarities in the next section.
3.4 SBL Vs. Ecolocation

In Ecolocation, the location estimate of the unknown node is the centroid of all grid points that have maximum number of matched constraints, whereas, in SBL the centroid of the region represented by a location sequence is the location estimate. Figure 3.9 shows the overlap of Ecolocation scanning grid points and the regions created by bisector lines in SBL. It can be observed that if the scanning resolution of Ecolocation is high enough then the centroid of the highest-constraint-matching grid points is the same as the centroid of the region represented by the corresponding location sequence. This results from the fact that the constraint set in Ecolocation is equivalent to the location sequence in SBL.

The main difference between Ecolocation and SBL is in their complexity. While Ecolocation takes $O\left(\frac{n^2S^2}{r^2}\right)$ time and $O(n^2)$ space, SBL takes $O(n^6)$ time and $O(n^5)$ space. Clearly, while Ecolocation depends on the number of reference nodes in the radio range $n$, localization space size $S^2$ and the scanning resolution $r$, SBL depends only on the number of reference nodes $n$. This suggests that for large localization spaces and high location resolutions SBL is the preferred method and for small localization spaces and low location resolutions Ecolocation is preferred.
Since Ecolocation is equivalent to SBL for high scanning resolutions, for the rest of the chapter we focus on SBL.

3.5 Evaluation

In this section, we present a complete performance evaluation of sequence-based localization (SBL). First, we discuss its inherent location error characteristics and then using simulations, we study its performance as a function of RF channel and node deployment parameters. We also present a comparative study with three other state-of-the-art localization techniques.

3.5.1 Location Error Characteristics

Each location sequence maps to the centroid of the region it represents. Representing all locations in a region by its centroid comes at the cost of error in the location estimate of the location sequence. If the region is a face, then the location error is of the order of the square-root of the area of the face and if the region is an edge then it is of the order of the length of the edge. Figure 3.10 plots the average, average maximum and average minimum face areas and edge lengths gathered over 1000 random trials in each of which $n$ reference nodes were placed uniformly randomly in a square localization space of size $S \times S$ sq. meters. The main error characteristics obtained from curve fitting can be summarized as follows:

- The average face area varies proportional to $\frac{1}{n^2}$. Since the location estimate error of locations in a face region is proportional to the square-root of its area, the average location estimate error for locations in a face region reduces proportional to $n^2$.

- The average maximum face area varies proportional to $\frac{1}{n^2}$. Therefore, the maximum location estimate error in a face region reduces proportional to $n$ which is slower than the reduction in average location estimate error.
Figure 3.10: Simulation results averaged over 1000 random trials (with 100 different random seeds) in each of which $n$ reference nodes were placed uniformly at random in a 2D square localization area of $S \times S$ sq. meters. (a) The average maximum, average and average minimum face areas as a function of the number of reference nodes. (b) The average maximum, average and average minimum edge lengths as a function of the number of reference nodes. $K_1$, $K_2$ and $K_3$, $K_4$ are scaling constants.

- The average edge length varies proportional to $\frac{1}{n^{2.5}}$. Since, the location estimate error for locations on an edge is proportional to its length, the average location estimate error for locations on an edge reduces proportional to $n^{2.5}$ which is faster than that for locations in a face region.

- The maximum edge length varies proportional to $\frac{1}{(n+1.5)}$. Therefore, the maximum location estimate error for locations on an edge reduces proportional to $n$ which is slower than the reduction of average location estimate error.

Apart from the above location errors, the performance of sequence-based localization is affected by random errors in RSS measurements due to multi-path and shadowing effects of the RF channel. In the rest of this section, we present results from simulation studies that capture the effect of these random errors on the performance of SBL.
3.5.2 Simulation Model

The most widely used simulation model to generate RSS samples as a function of distance in RF channels is the log-normal shadowing model [83]:

$$P_R(d) = P_T - PL(d_0) - 10\eta\log_{10}\frac{d}{d_0} + X_\sigma$$  \hspace{1cm} (3.22)

where, $P_R$ is the received signal power, $P_T$ is the transmit power and $PL(d_0)$ is path loss for a reference distance of $d_0$. $\eta$ is the path loss exponent and the random variation in RSS is expressed as a Gaussian random variable of zero mean and $\sigma^2$ variance, $X_\sigma = N(0, \sigma^2)$. All powers are in dBm and all distances are in meters.

In this model we do not provision separately for any obstructions like walls. If obstructions are to be considered an extra constant needs to be subtracted from the right hand side of the above equation to account for the attenuation in them (the constant depends on the type and number of obstructions).

3.5.3 Simulation Parameters

The accuracy of radio frequency based localization techniques depends on a number of parameters. Chief among this the accuracy of RSS measurements. In an ideal world, in which RSS values are not affected by multi-path fading effects, they represent true distances between nodes, which can lead to very accurate localization of unknown nodes. The ideal world is represented by $\sigma = 0$ in Equation 3.22. However in the the real world RSS values are corrupted by multi-path fading effects. This has a profound influence on the accuracy of RF localization techniques. According to the above propagation model RSS values are defined by $\eta$ and $\sigma$ values for the given environment. Since every RF environment can be characterized by $\eta$ and $\sigma$ values ([47],[68]) it is necessary to study the accuracy of RF localization techniques as a function of these two parameters.
In addition the density and number of reference nodes available to the unknown node has a significant influence on the number of reference nodes ([100],[48], etc). Thus the location estimate of any RF-based localization technique depends on a fundamental set of parameters which can be broadly categorized as follows:

1. **RF Channel Characteristics** ([47], [83])

   (a) Path loss exponent ($\eta$): Measures the power attenuation of RF signals relative to distance.

   (b) Standard deviation ($\sigma$): Measures the standard deviation in RSS measurements due to log-normal shadowing.

   The values of $\eta$ and $\sigma$ change with the frequency of operation and the obstructions and disturbance in the environment.

2. **Node Deployment Parameters**:

   (a) Number of reference nodes ($n$).

   (b) Reference node density ($\beta$).

Table 3.4 lists the typical values and ranges for different parameters used in our simulations.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Typical Value</th>
<th>Typical Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_T$</td>
<td>4dBm (max.)</td>
<td>NA</td>
</tr>
<tr>
<td>$PL(d_0)$</td>
<td>55dB ($d_0 = 1m$) [68]</td>
<td>NA</td>
</tr>
<tr>
<td>$\eta$</td>
<td>4 (indoors)</td>
<td>1 – 7 [47]</td>
</tr>
<tr>
<td></td>
<td>4 (outdoors)</td>
<td></td>
</tr>
<tr>
<td>$\sigma$</td>
<td>7 (indoors)</td>
<td>2 – 14 [47]</td>
</tr>
<tr>
<td></td>
<td>4 (outdoors)</td>
<td></td>
</tr>
<tr>
<td>$n$</td>
<td>10</td>
<td>3 – 10</td>
</tr>
<tr>
<td>$\beta$</td>
<td>0.1 (one node in 10 sq.m)</td>
<td>{0.01,0.04,0.1,1}</td>
</tr>
</tbody>
</table>

Table 3.4: Typical values and ranges for different simulation parameters
3.5.4 Simulation Procedure

We assume that all reference nodes are in radio range of each other and also that of the unknown node. A 48 bit arithmetic linear congruential pseudo random number generator was used and results were averaged over 100 random trials. In each trial, $n$ reference nodes were placed uniformly randomly in a square localization space of size $S \times S$ sq. meters and the unknown node was placed at 100 different locations on a grid of $\frac{S}{10}$ separation. In total, the results presented are averaged over 10000 different scenarios. In our simulations we use $S = 100$ meters.

The performance of sequence-based localization is measured in terms of location error for a wide range of RF channel conditions and node deployment parameters. Location error is defined as the Euclidean distance between the location estimate and the actual location of the unknown node. The location error is averaged over 100 random trials as described previously.

Figure 3.11 plots the two distance metrics described in the previous section as a function of the number of reference nodes ($n$) or in other words the length of the location sequence. There is a growing difference, however small, between the two metrics with increasing length of the sequence, with Kendall’s Tau performing increasingly better than Spearman’s correlation in terms of the location estimate error.

---

![Figure 3.11](image)

Figure 3.11: Average location error as measured using Spearman’s correlation and Kendall’s Tau as a function of the number of reference nodes.
Figure 3.12: *Sequence corruption*: Cumulative distribution function of *Kendall’s Tau* $T$ between the RSS location sequence and true location sequence for varying (a) standard deviation ($\sigma$) (b) path loss exponent ($\eta$) (c) number of reference nodes ($n$).

### 3.5.5 Simulation Results: Sequence Corruption

Figure 3.12 plots the corruption in location sequences, represented by $T$, due to RF channel and node deployment parameters. According to these results, the corruption in location sequences

- increases with increasing randomness in the RF channel represented by standard deviation in RSS, $\sigma$. (Figure 3.12(a))
- decreases with increasing path loss exponent, $\eta$. (Figure 3.12(b))
- is independent of the number of reference nodes in the localization space, $n$. (Figure 3.12(c))

### 3.5.6 Simulation Results: Performance Study

Figure 3.13 plots the average location error due to SBL as a function of RF channel and node deployment parameters. The main results are:

- Location error due to SBL is higher for RF channels with higher standard deviation ($\sigma$) values (Figure 3.13(a)). This is due to higher levels of corruption in location sequences at higher values of $\sigma$. 

Figure 3.13: Performance: (a) Average location error as a function of RF channel parameters - standard deviation ($\sigma$) and path loss exponent ($\eta$). (b) Average location error as a function of node deployment parameters - number of reference nodes ($n$) and reference node density ($\beta$). (c) Average location error as a function of the location of the unknown node.

Figure 3.14: (a) Average location error as a function of the sequence corruption ($T$) and as a function of the distance ($\tau$) between the corrupted sequence and its nearest feasible sequence in the location sequence table. (b) Correlation between $\tau$ and $T$. 
• Location error due to SBL is lower for RF channels with higher path loss exponent ($\eta$) values (Figure 3.13(b)). This is due to lower levels of corruption in location sequences at higher $\eta$ values.

• Location error due to SBL reduces with increasing number of reference nodes ($n$) suggesting that longer sequences are more robust to RF channel non-idealities than shorter sequences. (Figure 3.13(b))

• Location error due to SBL reduces with increasing reference node density $\beta$ according to Figure 3.13(b).

• Location error due to SBL depends on the location of the unknown node. Figure 3.13(c) plots the average location error for all possible unknown node locations in the localization space. It shows that unknown node locations that are closer to the center of the localization space have lower location error than unknown node locations closer to the boundaries of the localization space. This can be verified from the observation (Eg. Figure 3.4(b)) that for any arrangement of bisector lines, the faces and edges towards the center of the localization space have smaller areas and lengths respectively compared to that of at its boundaries. Consequently, for unknown node locations towards the center of the localization space, the location to which the nearest feasible sequence of the corrupted sequence maps will be closer to the true location of the unknown node than for locations towards the boundaries. This results in lower location errors for unknown node locations towards the center of the localization space than for locations towards its boundaries.

• Figure 3.14(a) plots average location error as a function of Kendall’s Tau values $T$ and $\tau$ and Figure 3.14(b) plots $\tau$ as a function of $T$. The figures suggest that:

  – The location error is correlated to $T$, the corruption due to RF channel.
– The location error is correlated to $\tau$, the distance between the corrupted sequence and the nearest feasible sequence.

– A correlation exists between $\tau$ and $T$.

This suggests that, $\tau$, which is a measurable quantity, as apposed to $T$, could be used as a quantitative indicator of the location error due to sequence-based localization. Also, owing to its correlation to $T$, it could also be used as an approximate indicator of the state of the RF channel.

### 3.5.7 Simulation Results: Comparative Study

We compare SBL with three other localization techniques - least squares estimator, proximity localization and 3-centroid.

The least squares estimator method has been discussed in detail in Section 2.3 of Chapter 2. In proximity localization, the location of the closest reference node by RSS value is chosen as the location of the unknown node. This is an extreme special case of SBL in which the sequence is of length 1. In Centroid technique, the centroid of all the reference nodes in the radio range of the unknown node is chosen as its location ([26]). Since, in our case, all reference nodes are in the radio range of the unknown node the location error would be independent of the RF channel characteristics. In order to measure the effect of these characteristics on the centroid technique we choose the centroid of the closest three reference nodes by RSS values as the location of the unknown node, called 3-centroid.

Figure 3.15 plots the average location error due to SBL, LSE, Proximity and 3-Centroid as a function of the standard deviation in RSS log-normal distribution $\sigma$ for different values of path loss exponents $\eta$ and for different values of number of reference nodes $n$. The main results of the comparison are:
Figure 3.15: Comparison: Average location error due to SBL, LSE, Proximity and 3-Centroid as a function of standard deviation of RSS log-normal distribution $\sigma$ for different values of path loss exponent $\eta$. (a) $\eta = 2, n = 10$ (b) $\eta = 4, n = 10$ (c) $\eta = 6, n = 10$ and for different values of number of reference nodes $n$. (a) $n = 4, \eta = 4$ (b) $n = 7, \eta = 4$ (c) $n = 10, \eta = 4$.

- SBL performs better than Proximity and 3-Centroid over a range of RF channel and node deployment parameters.

- SBL performs better than LSE for higher values of $\sigma$, whereas LSE performs better than SBL for lower values of $\sigma$. There is a crossover value of $\sigma$ between the error due to SBL and LSE and this value of $\sigma$ is higher for environments that have more attenuation i.e., higher values of path loss exponent $\eta$. There is no significant change in the value of crossover $\sigma$ with changing number of reference nodes $n$.

- For lower values of $\sigma$, the location error due to SBL decreases faster than location error due to LSE for increasing values of $n$. This can be seen in Figures 3.15(a)(b)(c) in which the difference between the location error due to SBL and LSE reduces with increasing values of $n$.

- LSE is out performed by all other localization techniques after some value of $\sigma$ and this value is the lowest for SBL.
It should be noted that, in the above simulations LSE operates at a considerable advantage over other techniques as the exact value of the path loss exponent $\eta$ is known. This advantage vanishes in real world scenarios where the value of $\eta$ is very difficult to estimate accurately owing to its dependence on the area features such as walls, furniture, etc. Thus, LSE may not perform as well in real world scenarios. Table 3.5 compares the time and space complexities of SBL with that of the other three localization techniques. We believe that the efficiency of SBL can be increased significantly by using more efficient location sequence table search algorithms as opposed to a naive search.

<table>
<thead>
<tr>
<th></th>
<th>SBL</th>
<th>LSE</th>
<th>Proximity</th>
<th>3-Centroid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>$O(n^6)$</td>
<td>$O(nr^2)$</td>
<td>$O(n \log n)$</td>
<td>$O(n \log n)$</td>
</tr>
<tr>
<td>Space</td>
<td>$O(n^5)$</td>
<td>$O(r^2)$</td>
<td>$O(n)$</td>
<td>$O(n)$</td>
</tr>
</tbody>
</table>

Table 3.5: Comparison of worst-case computational complexities of SBL, LSE, Proximity and 3-Centroid.

### 3.6 Real World Experiments

The performance of sequence based localization in real systems is studied through two experiments, representing different RF channel and node deployment parameters, conducted using Berkeley MICA 2 motes [6]. The first experiment was conducted in a parking lot which represents a relatively obstruction free RF channel and the second experiment was conducted in an office building with many rooms and furniture that represents a typical indoor environment. For comparison, the locations of the unknown nodes were also estimated using the three localization techniques - least squares estimator (LSE), proximity localization, 3-centroid - described in the previous Section.
3.6.1 Outdoor Experiment: Parking lot

The RF channel in an outdoor parking lot represents a class of relatively obstruction free channels. Eleven MICA 2 motes were placed randomly on the ground as shown in Figure 3.16. All motes were in line of sight of each other and all of them were programmed to broadcast a single packet without interfering with each other\(^2\). The motes recorded the RSS values of the received packets and stored them in their EEPROMs which were later used off-line for location estimation.

The locations of all the motes were estimated and compared with their true locations. Since all motes were in radio range of each other each mote had ten reference nodes. For the LSE method, to estimate the distances between the motes, the RSS model described by Equation 3.22 in Section 3.5.2 was used as there were no obstructions between motes in this experiment. The performance of the LSE technique depends on the value of the path loss exponent \(\eta\), for the area in which the experiment was conducted. For this experiment, we used the true distances and the corresponding RSS values between the reference nodes and the unknown node to estimate the value of \(\eta\). Figure 3.16(a) plots RSS values as a function of distance and the least-squares (best fit) line obtained from linear regression analysis.

If \((x_k, y_k), 1 \leq k \leq m\) are the data points, the slope (which is \(-\eta\)) and intercept of the least-squares best fit line are given by ([1]):

\[
\text{slope} = -\eta = \frac{\left(\sum_{k=1}^{m} x_k\right)\left(\sum_{k=1}^{m} y_k\right)}{(\sum_{k=1}^{m} x_k)^2} - m\left(\sum_{k=1}^{m} x_k^2\right) \\
\text{intercept} = \frac{\left(\sum_{k=1}^{m} x_k\right)\left(\sum_{k=1}^{m} y_k\right)}{(\sum_{k=1}^{m} x_k)^2} - m\left(\sum_{k=1}^{m} x_k^2\right)
\]

\(\text{(3.23)}\)

\(\text{(3.24)}\)

\(^2\)We had actually measured RSS of 100 packets in one minute and observed that their standard deviation was less than 0.5dBm. Therefore, we decided to use only a single packet for localization. In real application scenarios this would help in conserving energy at the mote and reducing the delay in localization without affecting its accuracy.
Figure 3.16: Outdoor experiment: 11 MICA 2 motes, placed randomly in a 144 sq.meters area, were used as reference nodes as well as unknown nodes. Consequently, each unknown node had 10 reference nodes. (a) Path loss exponent calculation, $\eta = 2.9$. (b) Comparison between true locations and SBL location estimates. (c) Location error due to SBL, LSE, Proximity and 3-Centroid (the nodes are ordered in increasing error of SBL). (d) Corruption measure $T$ and error indicator $\tau$. 
We applied the above expressions to the RSS Vs. distance data and calculated the value of \( \eta \) to be 2.9. We used this value of \( \eta \) to evaluate the LSE technique.

Figure 3.16(b) compares the true mote locations with SBL location estimates for all the motes. The Figure also shows the arrangement induced by the perpendicular bisectors between all pairs of reference nodes. Figure 3.16(c) plots the error at each mote location in meters due to all the four techniques. Evidently, SBL performs better than Proximity and 3-Centroid in ten out of eleven cases and it performs better than LSE in all the eleven cases.

Figure 3.16(d) plots the sequence corruption \((T)\) at each mote location and the distance \((\tau)\) between the corrupted sequence and the nearest feasible sequence in the location sequence table for all the 11 nodes. The correlation between \(T\) and \(\tau\) can be clearly seen from the Figure. Comparing Figure 3.16(c) and Figure 3.16(d), broad correlations between \(T\) and location error and between \(\tau\) and location error can be observed for SBL. For example, the location error is highest for node IDs 1 and 9, in that order, and \(\tau\) is the lowest for the same node IDs in the same order. Also, the location error is almost equal for nodes 8, 2, 7 and 10. This trend is also reflected in the values of \(\tau\) for those nodes.

### 3.6.2 Indoor Experiment: Office building

Office buildings with features such as rooms, corridors, furniture and other obstructions represent a distinct class of RF channels. Twelve MICA 2 motes (reference nodes) were placed on the ground randomly in a corner of the Electrical Engineering building at USC spanning different rooms and corridors. Figure 3.17 shows a schematic of the experimental setup. In this experiment, an unknown node was placed at five different locations and these locations were estimated using all the twelve motes as reference nodes. As in the outdoor experiment, the unknown node was programmed to broadcast a single packet from each location and the reference nodes recorded the RSS values of this packet in their respective EEPROMs which were later used off-line for location estimation.
Figure 3.17: Indoor experiment: 12 MICA 2 motes, placed randomly in a 120 sq.meters area, were used as reference nodes. The location of the unknown node was estimated for 5 different locations using the 12 reference nodes. (a) Path loss exponent calculation, $\eta = 2.2$. (b) Comparison between true path and SBL estimated path. (c) Location error due to SBL, LSE, Proximity and 3-Centroid (the nodes are ordered in increasing error of SBL). (d) Corruption measure $T$ and error indicator $\tau$.

Unlike in the outdoor experiment not all motes were in line of sight of each other even though they were in each other’s radio range. A subset of the motes had obstructions in between them in the form of walls. For this experiment, we calculated the value of $\eta$ be 2.2 by applying the same linear regression analysis used for the outdoor experiment, to the indoor RSS Vs. distance data. Figure 3.17(a) shows the data and least-squares line.

Figure 3.17(b) compares the SBL location estimates of the five unknown node locations with their true locations. It can be seen that the path of the location estimates closely follows the
true path of the unknown node. Figure 3.17(c) plots the location estimate error due to SBL, LSE, Proximity and 3-Centroid techniques for each unknown node location. It can be observed that SBL performs better than LSE and 3-centroid in four out of the five cases and better than Proximity in two out of five cases. A possible reason why proximity is performing well is the relatively dense distribution of the reference nodes.

Figure 3.17(d) plots the sequence corruption \((T)\) at each mote location and the distance \((\tau)\) between the corrupted sequence and the nearest feasible sequence in the location sequence table for all the 5 unknown node locations. Comparing this Figure and Figure 3.16(d) shows that sequences are more corrupted in the indoor experiment than the outdoor experiment, which was expected. Also, as in the outdoor experiment, there is a clear correlation between \(T\) and \(\tau\) for the indoor experiment also. But the correlations between \(T\) and location error and between \(\tau\) and location error are not as clear as that in the outdoor experiment.

### 3.6.3 Discussion

Experimental results show that localization techniques are more accurate for relatively clutter free RF channel environments (outdoors with line of sight) than RF channels with many obstructions (indoor environment). Also, the performance of LSE in real world scenarios is worse than in simulations, as was conjectured in Section 3.5.7. This is mainly because the radio propagation model of Equation 3.22 is an approximate model and the location estimate accuracy for the LSE technique depends heavily on the accuracy of \(\eta\) estimate. The RSS measurements in the experiments depend on antenna orientations, antenna height and transmitter/receiver non-determinism. For simulations, these issues can be captured within the log-normal random term in Equation 3.22.
3.7 Chapter Summary

In this chapter we presented a simple and novel localization technique based on location sequences called Sequence-Based Localization (SBL). In Sequence Based Localization location sequences are used to uniquely identify distinct regions in the localization space. The location of the unknown node is estimated by first determining its location sequence using RSS measurements of RF signals between the unknown node and the reference nodes. And then searching through a pre-determined list of all feasible location sequences in the localization space, called the location sequence table, to find the region represented by the “nearest” one. In this chapter, we derived expressions for the maximum number of location sequence and presented an algorithm to construct the location sequence table. We described distance metrics that measure the distance between location sequences and used them to determine the corruption in location sequences due to RF channel non-idealities. We identified an approximate indicator of the extent of location estimation error using the same distance metrics. Through examples we demonstrated the robustness of sequence-based localization to RF channel non-idealities. Through exhaustive simulations and systematic real mote experiments we evaluated the performance of our localization system and presented a comparison with other state-of-the-art localization techniques for different RF channel and node deployment parameters. Results showed that SBL performs well and better than other state-of-the-art localization techniques in both indoor and outdoor environments.
Chapter 4

Fast & Fair Localization

4.1 Introduction

In this chapter, we focus on the problems of fast and fair localization of unknown nodes or mobile devices. Related work in the literature has studied the speed of the movement of the mobile device versus the accuracy of tracking [91] and has proposed techniques to balance the speed of tracking and energy consumption [94]. But to the best of our knowledge ours is the first attempt to study the problem of fast and fair localization. We separate the problems of fast and fair localization from the problem of accurate localization so that the solutions for the former are compatible with any solution for the latter.

Fast localization requires minimizing the response time of reference nodes, called the localization delay, to localization requests from mobile devices. This requires avoiding collisions and retransmissions of the reference node responses. This can be achieved by splitting time into slots and scheduling reference nodes to transmit one in each time slot. Fair localization requires minimizing the variation in response time over all locations of the mobile device in the localization area.

The problems of fast and fair localization can be formulated as a single problem of minimizing the maximum localization delay in the localization area. We show that this problem is closely
related to the NP-complete \textit{minimum length broadcast frame} problem ([82]) in which, the total number of time slots required to schedule all the reference nodes in the localization area is minimized. We investigate a polynomial time heuristic algorithm for this problem and study its performance in terms of localization delay and fairness. A comparative study is also presented for two different reference node deployment distributions - grid and random - for different reference node density values and different levels of location estimate accuracy.

The rest of the chapter is organized as follows: In Section 4.2 we motivate the problem of fast localization. We state our assumptions and introduce terminology, define the different terms used, formulate the problem of fast/fair localization, present the heuristic time slot scheduling algorithm and define metrics to study its performance in Sections 4.3, 4.4, 4.5, 4.6 and 4.7, respectively. In Section 4.8, analytical and simulation based evaluations of this algorithm are presented. In Section 4.9, we discuss realistic application constraints for fast/fair localization and summarize the chapter in Section 4.10.

\section{Motivation}

As per the conditions for effective location support systems described in Chapter 1, energy efficient operation of location support services is of utmost importance as the wireless nodes are severely energy constrained. For this reason, most WSN deployments incorporate sleep schedules for nodes in which the nodes are awake only for a fraction of a duty cycle. The nodes have to accomplish their assigned tasks within this small period of time. Since the WSN has to perform different tasks at the same, the time allocated for localization will be a fraction of the wake up time of the wireless node. Therefore, the response time of the reference nodes to localization requests from unknown nodes is limited by the duty cycle of the WSN. Each reference node can transmit a single localization packet in each duty cycle.
In order to minimize the response time for localization requests, the time allocated for localization during node wake up should not be wasted through collision of packets or back-offs. This can be achieved only through scheduling of nodes in which nodes are assigned specific duty cycles to transmit their localization packets.

If, for example, the nodes in a WSN wake up on a duty cycle of 100 msecs, the unknown node will receive localization packets from the reference nodes with an interval of 100 msecs. If the localization technique requires packets from 10 different reference nodes, the response time for a localization request is 1 sec.

Having motivated the problem of fast localization, before we formally define the problem of fast/fair localization, we list our assumptions and terminology.

4.3 Assumptions and Terminology

In this section we state our assumptions and introduce terminology.

1. All reference nodes and mobile devices transmit in the same frequency band and at the same power implying that the radio range \( R \) meters is the same for all of them.

2. A set \( A \) of \( K \) reference nodes are deployed in a two-dimensional, square shaped localization area of side \( S(\gg R) \) meters. Their locations depend on the deployment distribution (such as grid, random, etc). The reference node density, \( \beta = \frac{N}{S^2} \).

3. The radio range of reference nodes and mobile devices is the same in all directions \(^1\) and the disc shaped area spanned by the radio range is called a cell. The in-square of a cell is the largest square that is contained within the cell and the out-square of a cell is the smallest square that contains the cell, as illustrated in Figure 4.1.

\(^1\)This is an idealized radio model. In Section 4.9 we discuss the effects of realistic radio models on fast/fair localization.
4. In order to obtain a finite set of mobile device locations for evaluation purposes, the locations, \((x, y)\), of the mobile device are considered to be grid points separated by one meter. In order to avoid edge effects, grid points in a band of \(R \gg 1\) meters away from the boundaries of the localization area are excluded. Therefore, \((x, y) \in [R, S-R] \times [R, S-R]\).

5. Reference nodes transmit their location coordinates to the mobile devices in localization packets. The mobile devices use the localization packets to obtain the reference node coordinates and to measure their signal strength.

6. The location estimate accuracy for mobile devices increases with number of reference nodes (Chapter 3).

7. The number of reference nodes in the cell of a mobile device located at \((x, y)\), denoted by \(\theta(x, y)\), is determined by the reference node density \((\beta)\), radio range \((R)\) and the mobile device’s location \((x, y)\). On average, the number of reference nodes in a cell is equal to \(\pi R^2 \beta\). The number of reference nodes required by a localization technique to guarantee a desired level of location estimate accuracy, is denoted by \(k \leq \theta(x, y))^2\).

\[\text{It should be noted that in a real scenario, owing to the shadowing and multi-path effects of the wireless channel and features of the localization area, the same number of reference nodes may not provide the same level of accuracy for all locations of the mobile device. We assume that the value of } k \text{ is such that it can guarantee the desired level of accuracy even in the worst case.}\]
8. The reference node network is globally synchronized and time is split into slots of equal length. The duration of each time slot \( T \) consists of the transmission time of a localization packet and its propagation delay, transmission time of any acknowledgment packet and its propagation delay and a guard period to account for time synchronization. For evaluation purposes, \( T = 100 \text{ ms} \).

9. Let \( f \) be any time slot allocation function and let \( M \) (a function of \( f \)) be the number of time slots required by \( f \) to allocate time slots to all reference nodes in the localization area. The \( M \) time slots constitute a time frame and each reference node periodically transmits localization packets based on the position of the time slot allocated to it in the time frame. \( f \) is a function that maps the set of reference nodes \( A \) to the time slots in a time frame of length \( M \) and \( F \) denotes the set of all such allocation functions \( f \). Note that \( f \) is a function of the density and distribution of reference nodes' deployment.

10. The time slots assigned to reference nodes in the cell of a mobile device located at \((x, y)\), are denoted by \( t_{i,(x,y)}, i = 1, \ldots, \theta(x,y) \).

11. In order to avoid collisions of localization packets, no two reference nodes within \( 2R \) distance of each other are allocated the same time slot. Henceforth, this condition is referred to as the \( 2R\)-Rule.

### 4.4 Definitions

1. **Localization Request Arrival Time** is defined as the time instance when the mobile device requests the localization service. At this time, the mobile device starts collecting the localization packets transmitted by the reference nodes in its cell\(^3\). Localization requests are assumed to arrive at the starting edges of time slots. Thus, the arrival time, \( t \), of

---

\(^3\)Here we assume that reference nodes transmit their localization packets indefinitely after initialization. In Section 4.9 we discuss other possible cases.
a localization request at a mobile device located at \((x, y)\), is a uniform integer random variable in the interval \([1, M]\).

2. *Localization Delay* is defined as the time taken by a certain number of reference nodes to transmit their localization packets, one each, to the mobile device. For a given time slot allocation function \(f\), it depends on the location of the mobile device, the localization request arrival time and the number of reference nodes required for accurate localization. Therefore, it is denoted by \(D(x, y, t, k)\). It is measured in time slots.

3. *Localizable speed* of the mobile device is defined as the speed at which localization of the desired accuracy is possible. It is determined by the localization delay and time slot duration. The localizable speed \(V(x, y, t, k)\) is calculated in *meters/second* as follows:

\[
V(x, y, t, k) = \frac{1}{D(x, y, t, k)T}
\]  

(4.1)

Since localization delay changes with the location of the mobile device, the localizable speed of the mobile device also changes with its location in the localization area.

4. *Localization Fairness* is defined as the variation in the localizable speed over all possible locations of the mobile device in the localization area. It is measured as the percentage of locations at which the localizable speed is greater than 95% of its average.

The above terminology is illustrated through an example in Figure 4.2. It shows \(N = 36\) reference nodes deployed in a grid in a square localization area of side \(S = 5R\) meters and, their time slots allocated using some time slot allocation function. Notice that the allocation function follows the \(2R - Rule\) and the length of the time frame, \(M\), is equal to 6 for this allocation function. For a mobile device located at \(A\), the number of reference nodes in its cell \(\theta(x, y) = 5\) and their respective time slots, \(t_i,(x,y)\), are \(\{1, 2, 3, 4, 5\}\). Similarly, the number of reference nodes in the cell of a mobile device located at \(B\) is 4 and their respective time slots are
{1, 3, 4, 6}. Let the number of reference nodes required by a particular localization technique for the desired level of accuracy be \( k = 3 \) reference nodes. If a localization request arrives at time \( t = 1 \) for the mobile device at \( A \) then the localization delay is 3 time slots, whereas, if \( t = 5 \), the localization delay is 4 time slots. If the time slot duration is \( T = 100 \) ms, then the localizable speeds are \( \frac{1}{3 \times 0.1} = 3.33 \) m/s and \( \frac{1}{4 \times 0.1} = 2.5 \) m/s respectively, for the above two localization request arrival times.

![Figure 4.2: Example illustrating terminology.](image)

### 4.5 Problem Formulation

The aim of fast/fair localization is minimizing localization delay and providing fairness at the same time. This implies that the time slot scheduling algorithm should minimize localization delay and its standard deviation, over all possible locations of the mobile device, simultaneously. This can be achieved by minimizing the maximum localization delay over all locations of the mobile device [21].

In order to normalize the effect of localization request arrival time on localization delay and fairness, the expected value of the localization delay, with respect to \( t \), denoted by \( E_t(D(x, y, t, k)) \), is considered.
\[ E_t(D(x, y, t, k)) = \frac{1}{M} \sum_{t=1}^{M} D(x, y, t, k) \] (4.2)

Now, the solution to the fast/fair localization problem is an allocation function \( f^* \in \mathbb{F} \) that minimizes the maximum expected localization delay over all locations \((x, y)\) of the mobile device, that is:

\[ f^* = \arg \min_{f \in \mathbb{F}} \{ \max_{(x,y)} \{ E_t(D(x, y, t, k)) \} \} \] (4.3)

Consider the following two propositions.

**Proposition 1.**

\[ \max_{(x,y)} \{ E_t(D(x, y, t, k)) \} \leq \max_{(x,y,t)} \{ D(x, y, t, k) \} \] (4.4)

\( \forall \) allocation functions \( f \in \mathbb{F} \).

**Proof.** Proof by contradiction. Assume that

\[ \max_{(x,y)} \{ E_t(D(x, y, t, k)) \} > \max_{(x,y,t)} \{ D(x, y, t, k) \} \] (4.5)

for some allocation function \( f \in \mathbb{F} \). Let \((x', y') \in [R, S - R] \times [R, S - R]\) be the location at which \( E_t(D(x, y, t, k)) \) is the maximum. Clearly,

\[ \max_{(x,y,t)} \{ D(x, y, t, k) \} \geq \max_{t \in [1, M]} \{ D(x', y', t, k) \} \] (4.6)

From (4.5) and (4.6),

\[ E_t(D(x', y', t, k)) > \max_{t \in [1, M]} \{ D(x', y', t, k) \} \] (4.7)
This is a contradiction because for all locations \((x, y)\)

\[
D(x, y, t, k) \leq \max_{t \in [1,M]} \{D(x, y, t, k)\} \tag{4.8}
\]

and from Equation (4.2), for all locations \((x, y)\)

\[
E_t(D(x, y, t, k)) \leq \max_{t \in [1,M]} \{D(x, y, t, k)\} \tag{4.9}
\]

\[\square\]

**Proposition 2.**

\[
\max_{(x, y, t)} \{D(x, y, t, k)\} \leq M \tag{4.10}
\]

\(\forall\) allocation functions \(f \in \mathcal{F}\). Recall that \(M\) is a function of \(f\).

**Proof.** From the definition of localization delay, its maximum value is at most equal to the length of the time frame \(M\) for all locations of the mobile device and for all localization request arrival times. \(\square\)

The above two propositions lead to the following corollary:

**Corollary 2.**

\[
\min_{f \in \mathcal{F}} \{\max_{(x, y)} \{E_t(D(x, y, t, k))\}\} \leq \min_{f \in \mathcal{F}} \{M\} \tag{4.11}
\]

**Proof.** Since inequalities (4.4) and (4.10) hold true for all allocation functions \(f \in \mathcal{F}\), they are also true for the allocation function \(f^*\) that minimizes the maximum of the expected localization delay over all locations \((x, y)\) of the mobile device. Inequality (4.11) follows from the associative property of inequalities. \(\square\)

In addition, we make the following conjecture, which is open to be proved.
Conjecture 1. If $M^*$ is the frame length due to $f^*$, the allocation function that minimizes the maximum expected localization delay, then $M^*$ is only a small ($\approx 1$) factor away from the optimal $M$. 

Based on corollary 2, the approach we take in this chapter for fast/fair localization is to seek an allocation function that minimizes the upper bound on the maximum expected localization delay and study its performance in terms of localization delay and fairness. Now, the solution to the fast/fair localization problem is the allocation function $f^{**}$ that minimizes the length of the time frame, that is:

$$f^{**} = \arg\min_{f \in F} (M)$$

(4.12)

The above formulation of the fast/fair localization problem is a flavor of the graph coloring problem called the minimum length broadcast frame problem. Ramaswami et al. in [82] have shown that this problem is NP complete; therefore there is no known polynomial time allocation function $f^{**}$ that can schedule all reference nodes in the localization area to transmit in an optimal number of time slots. Nevertheless, many polynomial time heuristic algorithms have been proposed as solutions to this problem. Next, we present and analyze such a heuristic algorithm.
4.6 Scheduling Algorithm

Below, we present pseudo-code for a greedy heuristic algorithm that minimizes the length of the time frame using the location information already programmed into the reference nodes.

Algorithm 3. A Greedy Heuristic Time Slot Scheduling Algorithm

**Input:** Location coordinates \( \{(p_{xi}, p_{yi})\} \) of reference nodes \( \{q_i\}, i = 1, \ldots, N \) in the localization area, reference \( X = (0,0) \) and radio range \( R \).

**Output:** Network time slot schedule.

1. \( \{d_i\} = \text{DIST}(\{q_i\}, X) \)
2. \( \{Q_i\} = \text{MINSORT}(\{q_i\}, \{d_i\}) \)
3. \( T \leftarrow 1 \)
4. for \( i : 1 \rightarrow N \)
5.    if \( Q_i \) is not assigned a slot
6.      Assign slot \( T \) to \( Q_i \)
7.      Add \( Q_i \) to set \( S_T \)
8.    end if
9. end for
10. \( T \leftarrow T + 1 \)
11. end if
• **DIST**\(^{(A, B)}\) determines the Euclidean distance between elements of array \(A\) and point \(B\) and returns the array of distances.

• **MINSORT**\(^{(A, B)}\) minimum sorts the array \(A\) based on the values of array \(B\) and returns the sorted array.

In a centralized implementation, the above algorithm can be executed on a central server that knows the locations of all reference nodes in the network; the time slots can be assigned to the reference nodes later. In a distributed implementation, every reference node executes this algorithm and all of them agree on the same time slot schedule. In this case, every reference node is assumed to know the locations of all other reference nodes in the network.

**Complexity Analysis:** Every reference node determines the order of all reference nodes (line 1 of the pseudo code) in the network based on their distances from a reference point (line 0). This takes \(O(N \log N)\) time, \(O(N)\) space. The ordering of reference nodes with respect to a reference point in line ensures that reference nodes that are assigned the same time slot are as close to each other as possible without violating the 2R-Rule. This ensures scheduling of reference nodes in a minimum number of time slots. A time slot is assigned to each reference node in the network in lines 3 – 15. This takes \(O(N^2)\) time and \(O(N^2)\) space. In total, the algorithm takes \(O(N^2)\) time and \(O(N^2)\) space to assign time slots to all the \(N\) reference nodes in the network.

Next, we define performance metrics for fast/fair localization.
4.7 Metrics

For a given time slot allocation function, localization delay $D(x, y, t, k)$ is determined by the location of the mobile device $(x, y)$, the number of reference nodes in its cell $\theta(x, y)$, their time slots $t_{i,(x,y)}$, the length of the time frame $M$, the number of reference nodes required for localization $k$, and the localization request arrival time $t$. This leads to the following proposition.

**Proposition 3.** Localization delay is given by Equation 4.13 (see Figure 4.3). In this equation, $\tau(p, q)$ is the time slot of the $p^{th}$ reference node starting from time $q$ and $\gamma$ is the number of reference nodes whose transmission time slots are later than the localization request arrival time $t$.

**Proof.** All reference nodes in the cell of the mobile device located at $(x, y)$ are sorted from the earliest ($\min\{t_{i,(x,y)}\}$) to the latest ($\max\{t_{i,(x,y)}\}$) based on their time slots. Figure 4.4 shows the time slots of the $\theta(x,y)$ reference nodes in the cell as a subset of the $M$ time slots that constitute the time frame. Note that ($\min\{t_{i,(x,y)}\} \geq 1$) and ($\max\{t_{i,(x,y)}\} \leq M$).

![Figure 4.4: Three different cases (1), (2) and (3) depending on the relative position of localization request arrival time with respect to the times slots of reference nodes in the cell.](image)

Consider the following three exhaustive cases, based on the position of the localization request arrival time $t$ relative to the set of time slots of reference nodes in the cell of a mobile device located at $(x, y)$. 

78
1. The localization request arrival time of the mobile device is equal to or lower than the minimum of the time slots of the reference nodes in its cell \((t \leq \min\{t_{i,(x,y)}\})\): The mobile device has to wait till it receives localization packets from all the \(k\) reference nodes. The time slot of the \(k^{th}\) reference node later than \(t\) is same as the time slot of the \(k^{th}\) reference node later than \(\min\{t_{i,(x,y)}\}\), which is \(\tau(k,\min\{t_{i,(x,y)}\})\). Thus, the localization delay is given by:

\[
D(x, y, t, k) = \tau(k,\min\{t_{i,(x,y)}\}) - t + 1
\] (4.14)

2. The localization request arrival time of the mobile device is in between the minimum and maximum of the time slots of the reference nodes in its cell \((\min\{t_{i,(x,y)}\} < t \leq \max\{t_{i,(x,y)}\})\): Let \(\gamma\) be the number of reference nodes whose time slots are later than the localization request arrival time \(t\). If \(k \leq \gamma\), then the mobile device has to wait till the time slot of the \(k^{th}\) reference node starting from time \(t\), which is given by \(\tau(k,t)\). Therefore, the localization delay is:

\[
D(x, y, t, k) = \tau(k,t) - t + 1
\] (4.15)

If on the other hand \(k > \gamma\), i.e., if the number of reference nodes with time slots later than the localization request arrival time \(t\) is not sufficient, then the mobile device, after receiving localization packets from the \(\gamma\) reference nodes, has to wait till the end of the frame i.e., time slot \(M\), plus, it has to wait till it receives localization packets from the remaining \(k - \gamma\) reference nodes staring from the first time slot. The time slot of the \((k - \gamma)^{th}\) reference node staring from the first time slot is same as the \((k - \gamma)^{th}\) reference node staring from time slot \(\min\{t_{i,(x,y)}\}\), which is \(\tau(k - \gamma,\min\{t_{i,(x,y)}\})\). Therefore, the localization delay is given by:
\[ D(x, y, t, k) = M - t + 1 + \tau(k - \gamma, \min\{t_i, (x, y)\}) \] (4.16)

3. The localization request arrival time of the mobile device is later than the maximum of the time slots of the reference nodes in its cell \((t > \max\{t_i, (x, y)\})\): In this case, the mobile device has to wait till the end of frame \(i.e., \) time slot \(M\) and again starting from the first time slot till the time slot of the \(k^{th}\) reference node, which is \(\tau(k, \min\{t_i, (x, y)\})\). The localization delay in this case is:

\[ D(x, y, t, k) = M - t + 1 + \tau(k, \min\{t_i, (x, y)\}) \] (4.17)

We consider the following five performance metrics:

1. **Average localization delay** \((D_{\text{avg}}(k))\): It is the average of the expected localization delay, for each location \((x, y)\) of the mobile device, over all possible locations of the mobile device. It is a function of the desired level of accuracy manifested as \(k\) and is measured in time slots.

\[
D_{\text{avg}}(k) = \frac{1}{(S - 2R)^2} \sum_{x=R}^{S-R} \sum_{y=R}^{S-R} E_t(D(x, y, t, k))
\] (4.18)

2. **Average localizable speed** \((V_{\text{avg}}(k))\): It is determined by the average localization delay and the length of a time slot and is a function of \(k\).

\[
V_{\text{avg}}(k) = \frac{1}{(S - 2R)^2} \sum_{x=R}^{S-R} \sum_{y=R}^{S-R} E_t(V(x, y, t, k))
\] (4.19)
where, the expected localizable speed at location \((x, y)\) of the mobile device, \(E_t(V(x, y, t, k))\), is given by:

\[
E_t(V(x, y, t, k)) = \frac{1}{M} \sum_{t=1}^{M} V(x, y, t, k) \quad (4.20)
\]

The devices of \(V_{avg}(k)\) are meters/sec. This metric measures the speed of movement of the mobile device at which it can obtain the desired level of location estimate accuracy, on average, in the localization area. If the mobile device moves at a speed equal to \(V_{avg}(k)\) there is no guarantee that it will obtain the desired level of accuracy at all locations in the mobile device.

3. **Localization Fairness** (\(F(k)\)): It measures the percentage of locations of the mobile device in the localization area that can guarantee the desired location estimate accuracy at a speed of \(0.95V_{avg}(k)\). Higher the percentage of these locations, higher is the localization fairness.

4. **Minimum localizable speed** (\(V_{min}(k)\)):

\[
V_{min}(k) = \min_{(x, y)}\{E_t(V(x, y, t, k))\} \text{ m/s} \quad (4.21)
\]

This metric measures the localizable speed of the mobile device at which the reference node network can provide a localization area wide guarantee for the desired level of accuracy. In other words, if the mobile device moves at a speed that is equal to or lower than \(V_{min}(k)\), it is guaranteed to obtain the desired level of location estimate accuracy for all locations in the localization area.

5. **Maximum localizable speed** (\(V_{max}(k)\)):
Parameter & Value
---
Radio range, \( R \) & 40 meters \\
Localization area side, \( S \) & 200 meters \\
Reference node network size, \( N \) & \( \{121, 169, 256, 324, 441\} \) \\
Corresponding reference node densities, \( \beta = \text{one reference node in} \ \frac{\beta}{S^2} \text{sq. meters} \) & \( \{330.6, 236.7, 156.3, 123.5, 90.7\} \text{ sq. meters} \) \\
Number of reference nodes required for localization, \( k \) & \( \{3, 6, 8, 10\} \)

Table 4.1: Simulation parameters and their values.

\[
V_{max}(k) = \max_{(x,y)} \{E_t(V(x,y,t,k))\} \text{ m/s} \\
\tag{4.22}
\]

This metric measures the maximum possible localizable speed of the mobile device for all locations of the mobile device in the localization area.

### 4.8 Evaluation

In this section, first, we analyze the geometries of grid and random deployments of reference nodes and for each of them derive the upper and lower bounds on the time frame length \( M \) required by any scheduling algorithm (i.e., any allocation function \( f \)). Next, we study the performance of the heuristic algorithm described in Section 4.6 in terms of the metrics defined in the previous section using simulations.

#### 4.8.1 Analysis

The definition of a cell ensures that all reference nodes in the cell of a mobile device are at most \( 2R \) distance away from each other. According to the \( 2R - Rule \) described in Section 4.3, the number of time slots required to schedule reference nodes in the network should be at least equal to the number of reference nodes in a cell. But this number of time slots is not sufficient. With this understanding, we first analyze the grid deployment of reference nodes and follow it up with analysis for random deployment.
4.8.1.1 Grid Deployment

**Proposition 4.** For a 2D grid, the number of reference nodes in the in-square of a cell is 
\[ n = 2m^2 + 6m + 5, \quad m = \left(\frac{R}{d} - 1\right), \]
where \(d\) is the inter-node distance and \(R\) is the radio range. The number of reference nodes in the out-square of a cell is \((2n - 1)\), the number of reference nodes on the perimeter of the in-square is \(2\sqrt{2n-1} - 2\) and the number of reference nodes on the perimeter of out-square is \(4\sqrt{2n-1} - 4\).

We do not provide a formal proof for the above proposition as it can be verified using simple geometric arguments. Notice that, at low reference node densities, the number of reference nodes in a cell is equal to the number of reference nodes in its in-square, where as, for high reference node densities, the number of reference nodes in the cell is greater than the number in its in-square.

**Proposition 5.** For a 2D grid, 
\[ \left[\pi R^2 \beta\right] + \sqrt{2n-1} - 3 < M < (2n - 1), \]
where \(\beta\) is the reference node density.

*Proof.* If all reference nodes in the out-square of a cell are assigned different time slots, this schedule satisfies the 2R-Rule. Clearly, these number of slots are sufficient. Therefore, according to proposition 4, at most \((2n - 1)\) time slots are required to schedule all reference nodes in the localization area.

However, these number of slots are not necessary because there exist pairs of reference nodes within this square that are greater than \(2R\) distance from each other and these pairs can be assigned the same time slots. In fact, all the slots assigned to reference nodes on the perimeter of the in-square of the cell can be reused by the reference nodes on the perimeter of the out-square. And the remaining reference nodes on the perimeter of the out-square definitely need extra slots. But, the geometry of the cell ensures that these remaining reference nodes are greater than \(2R\) away from each other in pairs and thus only half of the extra slots are indeed required.
The number of reference nodes on the perimeter of the cell out-square that do not pair up with the reference nodes on the perimeter of the cell in-square is $(4\sqrt{2n-1}) - 4 - (2\sqrt{2n-1} - 2) - 4 = 2\sqrt{2n-1} - 6$, where, the extra 4 is subtracted because these many reference nodes are common between the cell in-square and cell out-square. As stated previously, the number of extra time slots we require is only half the number of reference nodes that do not pair up, which is $(2\sqrt{2n-1} - 6) / 2 = \sqrt{2n-1} - 3$ time slots. In total, since the average number of reference nodes in a cell is $\pi R^2 \beta$, we need at least $(\lceil \pi R^2 \beta \rceil + \sqrt{2n-1} - 3)$ time slots.

**4.8.1.2 Random Deployment**

**Proposition 6.** For uniform random deployment of reference nodes with density $\beta$, $\lceil \pi R^2 \beta \rceil < M < \lceil 16R^2 \beta \rceil$.

**Proof.** As stated previously, the minimum number of time slots required is at least as many as the number of reference nodes in a cell, which is $\lceil \pi R^2 \beta \rceil$.

If all reference nodes in the out-square of a cell of radius $2R$ are assigned different time slots and the schedule is repeated throughout the network, it is sufficient because, with probability one, no two reference nodes in the network within $2R$ distance of each other are assigned the same time slot. The number of time slots required to achieve this is $\lceil 16R^2 \beta \rceil$. 

<table>
<thead>
<tr>
<th>Reference Node Network Size, $N$</th>
<th>Cell In-Square Size, $n$</th>
<th>Analytical M Lower Bound</th>
<th>Simulation M</th>
<th>Analytical M Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>121</td>
<td>$13(m = 1)$</td>
<td>18</td>
<td>19</td>
<td>25</td>
</tr>
<tr>
<td>169</td>
<td>$18.5(m = 1.5)$</td>
<td>25</td>
<td>29</td>
<td>36</td>
</tr>
<tr>
<td>256</td>
<td>$25(m = 2)$</td>
<td>37</td>
<td>41</td>
<td>49</td>
</tr>
<tr>
<td>324</td>
<td>$32.5(m = 2.5)$</td>
<td>47</td>
<td>54</td>
<td>64</td>
</tr>
<tr>
<td>441</td>
<td>$41(m = 3)$</td>
<td>62</td>
<td>70</td>
<td>81</td>
</tr>
</tbody>
</table>

Table 4.2: Comparison of analytical lower and upper bounds of $M$ with simulation results for grid deployment of different value of $N$, the network size. Note that the number of reference nodes in the in-square of a cell is $n = 2m^2 + 6m + 5$, $m = (R/2) - 1$ where, $R$ is the radio range and $d$ is the inter reference node distance (Proposition 4 in Section 4.8.1.1).
<table>
<thead>
<tr>
<th>Reference Node Network Size, N</th>
<th>Analytical M Lower Bound</th>
<th>Simulation Average M</th>
<th>Analytical M Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>121</td>
<td>16</td>
<td>29.3</td>
<td>75</td>
</tr>
<tr>
<td>169</td>
<td>22</td>
<td>38.3</td>
<td>109</td>
</tr>
<tr>
<td>256</td>
<td>33</td>
<td>54.2</td>
<td>164</td>
</tr>
<tr>
<td>324</td>
<td>42</td>
<td>66.0</td>
<td>208</td>
</tr>
<tr>
<td>441</td>
<td>56</td>
<td>89.6</td>
<td>283</td>
</tr>
</tbody>
</table>

Table 4.3: Comparison of analytical lower and upper bounds of $M$ with simulation results for random deployment of different value of $N$, the network size. The simulations results are average over 10 different random reference node network topologies.

### 4.8.2 Simulations

The performance of the heuristic scheduling algorithm is measured using simulations in terms of the metrics described in Section 4.7, for grid and uniform random deployments of reference nodes for five different reference node density values. Table 4.1 lists the various simulation parameters and their values.

The reference node locations are generated according to the deployment distribution and the scheduling algorithm uses these locations to assign time slots to them as described in Section 4.6. Table 4.2 and Table 4.3 compare the length of the time frame $M$ with its analytical lower and upper bounds for five different reference node network sizes for grid and uniform random deployments, respectively. For random deployment of reference nodes the results are averaged over 10 different random reference node network topologies. Clearly, the simulation results are within the analytical bounds. Owing to the regular geometry, the analytical bounds on $M$ are tighter for grid deployment as compared to random deployment.

Figure 4.5 plots the simulation results in terms of the five metrics described in Section 4.7. The main simulation results can be summarized as follows:

1. The average localization delay ($D_{avg}(k)$) increases with the number of reference nodes required for localization ($k$), implying that the mobile device has to wait for a longer time to obtain higher localization accuracy. Also, $D_{avg}(k)$ is lower for grid deployment
as compared to random deployment of reference nodes and its variation with respect to reference node density is minimal. (Figures 4.5(a) and 4.5(g))

2. The average localizable speed \( V_{\text{avg}}(k) \) of the mobile device decreases with the number of reference nodes required for localization \( k \), i.e., on average, the mobile device has to move slower to obtain higher location estimate accuracy, confirming the observation from average localization delay. Also, the average localizable speed of the mobile device is higher for grid deployment than random deployment of reference nodes. The effect of reference node density on the average localizable speed is minimal. (Figures 4.5(c) and 4.5(f))

3. Localization fairness \( F(k) \) is constant with respect to the number of reference nodes required for localization \( k \), i.e., localization fairness is independent of the desired level of location estimate accuracy. Also, localization fairness increases with reference node density implying that the number of locations in the localization area that can guarantee the desired level of location estimate accuracy for 95% of the average localizable speed increases with reference node density. And these locations are higher in number for grid deployment of reference nodes as compared to random deployment. (Figures 4.5(b) and 4.5(h))

4. The minimum localizable speed \( V_{\text{min}}(k) \) decreases with number of reference nodes required for localization \( k \) and increases with reference node density. This implies that the speed of movement of the mobile device should be lower for a localization area wide guarantee of higher location estimate accuracy and this speed increases with increasing reference node density. Also, this speed is higher for grid deployment of reference nodes as compared to random deployment. (Figures 4.5(d) and 4.5(f))

5. The absolute maximum localizable speed of the mobile device over all its locations in the localization area decreases with the desired level of location estimate accuracy. It is higher
for random deployment of reference nodes as compared to grid deployment (even though the difference between the values for the two distributions is very low) and its dependence on reference node density is minimal. (Figures 4.5(e) and 4.5(f))

4.9 Discussion

In this section, we present realistic application constraints faced by wireless sensor networks, and discuss their effect on fast/fair localization.

So far, we have assumed an idealistic radio model in which the radio range for reference nodes and mobile devices is the same in all directions. In this idealized radio model, reference nodes that are farther than $R$ meters from the mobile device cannot communicate with it. But in reality, localization packets sent by such reference nodes have a finite probability of reaching the mobile device [68] and this could potentially lead to collisions of localization packets at the mobile device. In order to avoid such collisions, the heuristic algorithm could be changed to allocate the same time slot to reference nodes that are farther than the present distance ($2R$).

We have assumed that the reference nodes and the mobile device operate in a single frequency band. Instead, if multiple frequencies can be used, the time slot scheduling algorithm can incorporate frequency diversity in addition to time diversity to reduce the response of time of reference nodes to localization requests and thus reduce the localization delay further. For this, the mobile device should be able to switch between different frequency bands quickly.

In this chapter, we have separated fast/fair localization from the techniques used for accurate localization. In contrast, if the time slot scheduling algorithm takes into account the number of reference nodes required for a desired level of accuracy by a specific localization technique there could be potential reduction in the localization delay with some collision tolerance at the mobile device.
Figure 4.5: Simulation Results: (a) Average localization delay $D_{\text{avg}}(k)$, (b) Localization fairness $F(k)$, (c) Average localizable speed $V_{\text{avg}}(k)$, (d) Minimum localizable speed $V_{\text{min}}(k)$, and (e) Maximum localizable speed $V_{\text{max}}(k)$; as a function of number of reference nodes required for localization $k$, for five reference node density values and for grid and random deployment of reference nodes. (f) $V_{\text{avg}}(k)$, $V_{\text{max}}(k)$, and $V_{\text{min}}(k)$, (g) Average localization delay $D_{\text{avg}}(k)$; as a function of reference node density $\beta$ for number of reference nodes required for localization $k = 8$, for grid and random deployments of reference nodes. (h) Localization fairness $F(k)$ as a function of reference node density $\beta$ for four different levels of location estimate accuracy ($k$) for grid and random deployments of reference nodes.
We would like to illuminate the fact that the structure of fast/fair localization problem is not limited to localization. This problem occurs in other applications of wireless sensor networks such as node discovery and data querying. The common underlying structure for fast/fair localization, node discovery and data querying is the minimization of response time as measured by the unknown node for localization, by the node discoverer for node discovery and by the data querier for data querying.

4.10 Chapter Summary

In this chapter, we introduced the problem of fast/fair localization of mobile device in infrastructure wireless sensor networks and showed that it is related to the minimum broadcast frame length problem. We investigated a greedy heuristic time scheduling algorithm for this problem using a defined set of five metrics - average localization delay, average localizable speed, localization fairness, minimum localizable speed and maximum localizable speed. We derived lower and upper bounds for the number of time slots required to schedule all anchors in the localization area by any scheduling algorithm for grid and random anchor deployment distributions using simple geometric arguments. Next, using simulations, we studied the dynamics of the above five metrics with respect to anchor deployment distributions, anchor densities and location estimate accuracies. Results show that the average localizable speed of mobile device decreases with increasing level of location estimate accuracy and its dependence on anchor density is minimal. The percentage of locations in the localization area that can guarantee a desired level of location estimate accuracy at a mobile device speed of 95% of the average localizable speed, the localization fairness, increases with anchor density and is independent of the accuracy level desired. The average localizable speed of the mobile device and localization fairness are better for grid deployment of anchors than for random deployment. Also, the localizable speed of the mobile
device at which a localization area wide guarantee of a desired level of accuracy can be provided
increases with anchor density and it is higher for grid deployment of anchors.
Part II

Medium Access for One-Hop Data Collection
Chapter 5

Background on Medium Access Techniques for One-Hop Data Collection

5.1 Introduction

Medium access techniques for wireless packet radio networks is one the fundamental areas that has been an active area of research for well over quarter of a century. However, new technologies introduce new wireless medium access control (MAC) conditions and challenges. The emergence of wireless sensor networks (WSN) over the last half a decade has introduced many new challenges and opportunities in MAC techniques owing to the diverse range of applications WSN are anticipated to be deployed for. In this thesis we consider the problem of medium access control for one-hop data collection in WSN.

In this problem a data sink seeks information from data sources (sensor nodes) in its radio range by issuing requests or queries. The sources give appropriate response to the sink based on the type of queries. This problem occurs frequently in many applications of WSN such as location support, neighbor discovery, data query and response, continuous data download, etc. Thus there exists a broad spectrum of application space based on the data query type for the one-hop data collection problem. In the next section we describe this application space.
5.2 Application Space

We identify a broad spectrum of medium access problems for the data collection application in wireless sensor networks based on the characteristic of the data to be collected. At one extreme of this spectrum is continuous data (CD) collection and at the other extreme is one-shot data (OSD) collection.

In continuous data collection, the sources always have a packet in their transmission queues. In one-shot data collection the sink is interested in one-shot data queries such as “Which nodes have observed the event?” or “Which nodes have recorded temperatures above 50F?” etc. The response to such one-shot queries is a single packet from each sensor node that contains the location of the node or a similar identification. Once the packet has been successfully transmitted from a node it is not in contention for the channel anymore. Neighbor discovery is a WSN application that falls at the one-shot data collection end of the application spectrum. It is an essential part of many routing protocols. In this application the sink is the node that discovers its neighboring nodes and a single packet is sufficient for each neighboring node to transmit its ID to the sink. Another WSN application in which the one-shot data collection problem occurs in localization. The unknown node (node with unknown location) sends out a localization request and the reference nodes (nodes with known location) in the radio range of the unknown node respond with a single packet each which contains their respective location coordinates.

The application space between these two extremes is made up of applications which have non-continuous data as shown in Figure 5.1. The transition from black to white indicates the spectrum transition from infinite packets in the queues to single-packet in the queues of contending nodes. The region between these two extremes is characterized by medium access conditions which are identified by queues with probabilistic occupancy and queues with finite (≤ 1) packet sizes.
We mainly focus on the two ends of the above spectrum of application space, namely, continuous data collection and one-shot data collection in the context of the one-hop data collection problem. As mentioned previously, the key distinguishing aspect for these two applications is the number of data packets in the transmission queues of data sources. The continuous data (CD) collection application is defined by the presence of “infinite” packets in the transmission queues of contending source nodes or alternatively called as back-logged queues. The one-shot data (OSD) collection application is defined by the presence of a single packet in the transmission queues of contending source nodes. This difference fundamentally alters the way these two applications are modeled, analyzed, and understood.

Traditionally, researchers have modeled, analyzed, and studied the performance of various medium access techniques for the one-hop data collection problem for continuous data and queues with probabilistic occupancy. However, owing to the application space specific to WSN, the one-shot data collection application occurs much more frequently in WSN. To the best of our knowledge ours is the first effort at modeling, analyzing, and understanding the performance of various medium access techniques for the one-shot data collection application.
In the next section we review the various medium access techniques proposed in the literature that are applicable to the one-hop data collection problem in WSN.

5.3 Medium Access Techniques

Medium access techniques for the one-hop data collection problem can be broadly classified into two classes, namely, randomized and scheduled. In this work, we mainly focus on randomized medium access techniques and we review the literature for the same. Review of scheduled medium access techniques (e.g. TDMA) is beyond the scope of this thesis.

Many solutions have been proposed over the years for the problem of medium access in wireless packet radio networks (e.g. [21], [34], [17], [16], [96], [85], [42], [43], [22], [49], [7], [56], [61], [13], [92], [37], [86], [89], [90], [97], [99], [9], [70], [35], [51], [87], [53], [52], [8], [88], [58], [95], [64], [67], [93], [23], [39], [62], [63], [25], [38], [27], [31], [14], [12], [98], [50], [46], [24], [28], [73], [29], [72], [66], [71], [81], [76]). These medium access control (MAC) techniques can be broadly classified into three main categories:

1. Slotted Aloha Medium Access
2. Carrier Sensing Medium Access
3. Tree/Stack Resolution Algorithms

Before we review each of the above category of MAC techniques we describe the MAC problem in the context of one-hop data collection, discuss the required assumptions, and define performance metrics for evaluation purposes.

5.3.1 Problem Description

The modeling and analysis of wireless MAC techniques or protocols is usually characterized by the assumption that each contending node always has packets to transmit or that the number of
packets in the queue is probabilistic based on a stationary arrival rate. In this case the network system reaches a *steady state* in which the system parameters approach a constant average value over time (for example, see [13], [23], [25], [27]). Significantly, these assumptions do not hold for one-shot data collection because the number of nodes contending for the wireless channel continuously and deterministically decreases with each successful packet transmission. This automatically implies that the system cannot reach steady state; and such a system is therefore said to be *transient*.

An important aspect of MAC protocols is the time synchronization of the data sources and data sinks. Time synchronization of nodes in wireless sensor networks is a very active field of research and much progress has been done in this area. Owing to this success even the IEEE standard 802.15.4 for low-power low-rate wireless networks stipulates time synchronization between all nodes in the network ([24]). In this thesis we focus on randomized medium access techniques in which all the data sources and the data sink are time synchronized. Thus, we assume that time is split into slots of equal length and that each node in the network can identify the starting and ending times of these time slots.

Recently many *sleep scheduling* based medium access mechanisms have been proposed (e.g. [97], [99]) for saving energy in wireless sensor networks. However, we do not consider sleep scheduling in this thesis.

### 5.3.1.1 Metrics

Now we define metrics that accurately capture the performance dynamics of MAC protocols for both CD and OSD scenarios. Let the number of contending nodes in the wireless sensor networks be $N$.

- **Throughput:** This metric is measured in terms of the amount of data successfully transmitted to the sink per unit time. We use this metric for the CD scenario in which
each node always has a packet to transmit. We denote throughput by $\Phi_{CD}(N)$ for the CD scenario.

- **Delay**: This metric is measured in terms of the amount of time taken by the sensor nodes to successfully transmit one packet each to the sink. We use this performance metric for the OSD scenario in which each node has a single packet to transmit. We denote delay by $\Delta_{OSD}(N)$ for the OSD scenario.

- **Energy Consumption**: This metric is measured in terms of the amount of energy consumed by each sensor nodes in the WSN for data transfer to the sink. For the CD scenario the energy consumption is measured per unit time and for the OSD scenario the energy consumption is measured for the entire time to transfer packets from all contending nodes to the sink. We calculate the energy consumption per node as an amortized quantity by calculating the energy consumption for all the nodes and dividing it by the number of nodes. We denote the amortized energy consumption metric by $\Sigma_{CD}(N)$ for the CD scenario and by $\Sigma_{OSD}(N)$ for the OSD scenario.

Next, we describe the energy model that we use to model the energy consumption in a sensor node.

### 5.3.1.2 Energy Model

For the two data collection scenarios CD and OSD under consideration the sensor nodes are either transmitting packet to the sink or receiving packets transmitted to the sink by other sensor nodes. For the OSD scenario once a sensor node transmits its packet successfully it shutdowns to save energy. This is not the case in the CD scenario as nodes always have packets to transmit. We denote the energy consumed by a sensor node per unit time during packet transmission by $\xi_T$ and the energy consumption per unit time during reception by $\xi_R$. 


Next, we briefly review each of the three classes of medium access techniques described in the previous section.

### 5.3.2 Slotted Aloha Medium Access

Slotted Aloha medium access protocol ([21], [9]) was one of the first random access mechanisms suggested for wireless medium access. In this protocol each contending node transmits a packet to the sink at the beginning of a time slot as soon as the packet is available. In order to avoid collisions a back-off collision avoidance policy is used at each node. In this thesis we consider the commonly used binary exponential back-off mechanism. We give a detailed description of this mechanism in the next chapter.

Much work has been done on the modeling, analysis and performance evaluation of the slotted Aloha protocol for different applications (a few examples include [70], [35], [51], [87], [53], [52], [8], [88], [10], [11]). The above previous work is completely based on either continuous data or non-continuous data collection. To the best of our knowledge no prior work has been done to analyze the performance of the slotted Aloha protocol for the one-shot data collection application.

In Chapter 6 we present a model for the slotted Aloha protocol with binary exponential back-off for the one-shot data collection scenario, analyze it using flow equations, and then evaluate the performance of the protocol.

### 5.3.3 Carrier Sense Medium Access

Carrier sensing medium access (CSMA) techniques have been proposed (example [58], [95], [64]) over a quarter of century for wireless packet radio techniques. In these techniques each contending node senses the channel, and transmits its packet if it finds the channel to be free of any transmissions and if the channel is busy with another transmission the node defers its transmission.
Since in this thesis we are interested in scenarios in which all the contending nodes are time synchronized to a common global clock, the CSMA techniques are applicable to cases in which the transmission of a data packet spans over multiple time slots. The slotted Aloha medium access techniques discussed in the previous subsection are applicable to cases in which the packet transmission time is equal to a single time slot.

If more than one nodes sense the channel to be free in a given time slot and both of them transmit their packet simultaneously, it results in a collision. One mechanism to avoid collisions is for the nodes to randomly back-off when they sense the channel to be busy. Thus different nodes choose different random times to sense the channel and transmit their packets. A commonly used and studied collision avoidance (CA) mechanism with CSMA techniques (CSMA/CA) is binary exponential back-off. We elaborate more on this mechanism in the context of CSMA techniques in Chapter 8.

Many IEEE standard MAC protocols have been proposed based on CSMA techniques. The most popular wireless MAC standard, the IEEE 802.11, or Wi-Fi, is a CSMA/CA technique. Tremendous amount of work has been done in modeling, analyzing and evaluating the performance of this standard in a diverse range of applications. A sample of these works is captured in [23], [39], [62], [25], [38], [27], [31], [14], [12], [98], [50], [46]. The IEEE 802.11 MAC protocol uses binary exponential back-off as the collision avoidance mechanism. It is beyond the scope of this thesis to present a review of the work on IEEE 802.11 MAC protocol, even though a thorough literature review has been done in this area for this thesis.

The IEEE 802.11 MAC protocol has been designed mainly for high speed wireless communication between compliant devices or for high speed internet access. Thus the main feature of the IEEE 802.11 MAC protocol is high data rate ranging from 10’s to 100’s of Mbps. This aspect is significantly different for wireless sensor networks which operate at several 100 Kbps at best. Starting with this difference, the operating conditions of the IEEE 802.11 protocol significantly differ from that of wireless sensor networks in such important aspects as energy conservation.
goals, device form factors, application requirements, etc. In view of these significant divergences
the IEEE has proposed and standardized a new protocol for low-power, low-rate wireless net-
works called the IEEE 802.15.4 standard. We provide a thorough overview of this standard
MAC protocol next.

5.3.3.1 IEEE 802.15.4

The IEEE 802.15.4 standard ([28]) allows different network topologies such as one-hop star and
multi-hop. In this thesis we consider the one-hop star topology with multiple data sources and
a single sink. In the star topology, a global synchronization of nodes is assumed and the time
is separated by beacons transmitted by a network coordinator. The beacon-interval consists of
a superframe and an optional energy saving time in which the nodes switch off their radio and
go to sleep. The superframe is divided into 16 time slots of $\delta = 320 \mu$secs duration each. The
superframe consists of a contention access period (CAP) and a period of guaranteed time slots
(GTS). The GTS is dedicated for low latency applications. In this chapter we consider only
the CAP mode (without the energy saving mode, GTS, and beacons) where medium access is
through slotted CSMA/CA.

In slotted CSMA/CA, a node can transmit its packet only after it senses the channel free for
a contention window (CW) of 2 time slots. The main purpose of the CW is to avoid collisions
between acknowledgement packets (ACKs) from the sink and data packets from the sources as
the protocol does not specifically provision time slots for ACKs [81]. A node chooses a time slot
uniformly at random from an initial window of $[0, 2^{BE} - 1]$, where $BE$ is the back-off exponent
with an initial value of 3. The node transmits its packet if the channel is sensed to be free in
that and the next time slots; if the channel is sensed to be busy the node backs off to a bigger
window with $BE = 4$. On a second busy channel sensing or a collision the node backs off to a
window with $aMaxBE = 5$ and remains constant. If a node is unable to transmit its packet
within 5 back-offs the transmission is assumed to be a failure and the packet is dropped. We
relax this condition in this thesis and allow a node to retransmit its packet until it is successful. Figure 5.2 shows the flow chart for a node using the IEEE 802.15.4 MAC. The IEEE 802.15.4 standard specifies a data rate of 250 kbps and a maximum MAC protocol data unit (MPDU) of 127 Bytes. Given this data rate, the transmission time for a typical packet of 50 Bytes is 5 time slots and for the MPDU it is 13 time slots.

Figure 5.2: Flow chart for IEEE 802.15.4 operation at a node.

In [66], the performance of the IEEE 802.15.4 MAC is evaluated in terms of throughput and energy efficiency using ns – 2 simulations for a maximum of 49 nodes. In [73], the performance of the standard MAC is evaluated for medical applications where the IEEE 802.15.4 devices interface with the traditional MAC technologies such as Ethernet. In [76], the performance of IEEE 802.15.4 MAC protocol is analyzed in the context of medical body area networks (BAN) where the energy efficiency of body implanted sensors is the focus given that their required life time is in the order of 10-15 years in these applications. In [71], a queuing analysis is presented for the sleep mode with possible finite buffers. In [72], the performance of the standard MAC is evaluated in the presence of both uplink and down-link traffic in the one-hop star topology network.
In this work, we evaluate the performance of the IEEE 802.15.4 MAC protocol for both ends of the application spectrum of the one-hop data collection problem in both high and low density scenarios. We also propose enhancements to the protocol based on channel-feedback.

Other recent work on CSMA based MAC protocols specific to wireless sensor networks include [67] and [93]. However, this work differs significantly from ours as the authors in this work focus on developing techniques to minimize the delay until the first packet transmission from the sensor nodes to the sink in the one-hop network. In our work we are interested in the delay and the corresponding energy consumption for all relevant data to be transmitted to the data sink.

Next, we discuss the next category of medium access techniques called the tree/stack algorithms.

### 5.3.4 Tree/Stack Algorithms

The main idea in this category of medium access techniques is to probabilistically and hierarchically (in a tree form) isolate the contending nodes and separate their collision domains to reduce collisions and thus increase the throughput of the wireless network. A stack is usually used to efficiently implement these tree based isolation algorithms. We will explain these algorithms through the example of the HT-splitting MAC protocol. In this protocol [21], the collision domains of the contending nodes are isolated probabilistically. The protocol starts out by all contending nodes in the radio range of the sink tossing a coin, and the subset of nodes with a heads (H) transmitting their packet. If there is a collision, all nodes with a H in the first level, again toss a coin and the subset of nodes with a H in both the present and the previous levels transmit their packets. This is continued until a single node has H from all the previous tosses and the present toss. Once this node finishes transmitting its packet, the node with a tails (T) in the present toss and a H in all the previous tosses transmits its packet. This process of descending and ascending the “tree” of coin tosses continues until all nodes transmit their packets.
In Chapter 7 we propose a tree-based location-aware medium access technique for one-shot data collection applications in which the collision domains of contending sensor nodes are separated based on their locations rather than probabilistically as in the HT-splitting MAC protocol.

5.3.5 Enhancements

Next, we discuss different possible enhancement mechanisms proposed over the years to the above discussed categories of medium access techniques. The two main mechanisms that have been used in the literature to enhance the performance of medium access protocols are through using relevant feedback from the wireless channel or through using the location information of nodes.

5.3.5.1 Channel Feedback

The idea of using feedback from the channel to control the transmission probabilities of contending nodes has been used for a long time. Rivest in [86] has proposed a ternary feedback model in which each node has to monitor three channel conditions - absence of transmissions, successful transmissions and, collisions. Rivest has shown that estimating the true value for the number of nodes \( n \) and setting the transmission probability to \( \frac{1}{n} \) maximizes the throughput in slotted-Aloha type protocols (in which the packet length is equal to a single time slot). If the packet length is of multiple time slots, this result does not hold true as we show in Proposition 9 in Section 8.3. In [25] a control mechanism has been presented that uses the energy consumed by a tagged node in the network in the above three channel conditions between two successful packet transmissions. This mechanism is not applicable in the case of OSD because each node has a single successful packet transmission. Similar strategies based on the estimation of the three channel conditions have been proposed ( [22], [56], [70]) all of which are more suitable for steady state conditions (like in CD) in which the number of contending nodes remain constant.
5.3.5.2 Location Information

The idea of using location information of contending nodes to enhance the performance of medium access techniques has been prevalent for some time. Corbett et al. in [33] propose a hybrid TDMA – Contention based protocol for multi-hop sensor networks that uses the locations of nodes for spatial reuse and time slot allocation to avoid collisions and interference. The space is divided into hexagonal cells, similar to cellular networks, and nodes within each cell use contention based medium access. In contrast to this, in our work on location-aware medium access technique, mentioned previously and presented in detail in Chapter 7, we use the locations of nodes to solve the problem of medium access within a cell. Liu et. al. in [65] use the location information of nodes within one-hop to provide energy efficiency and fault tolerance, even though the medium access is through contention-based random-access schemes. In our location-aware medium access work, the medium access itself is based on the locations of nodes. Nadeem et. al. in [74] use the location information in tandem with the capture effect to increase throughput in IEEE 802.11 DCF networks. In this, the location information is used to increase the spatial reuse efficiency and better manage interference leading to additional concurrent transmissions, thereby increasing the overall throughput of the protocol. Again, this work differs from ours in that, we solve the one-hop medium access problem using the location information of nodes in contrast to the multi-hop one.

5.4 Chapter Summary

In this chapter, we have reviewed the background work on medium access techniques for the one-hop data collection problem for wireless sensor networks. We have described the application space of operation and defined the metrics of interest that capture the performance of protocols in this application space. We have then presented an energy model we will use to evaluate the
energy performance of protocols. Finally, we have discussed three different categories of medium access techniques and the common performance enhancement techniques used in the literature.
Chapter 6

Analysis of Slotted Aloha Medium Access for One-Shot Data Collection

6.1 Introduction

In this chapter, we analyze the performance of slotted Aloha medium access for the one-hop one-shot data collection problem. As discussed in Chapter 5, the slotted Aloha protocol has been analyzed for many different application scenarios; but to the best of our knowledge ours is the first attempt at analyzing the protocol for the one-hop one-shot data collection problem.

More specifically, we model and analyze the performance of slotted Aloha with binary exponential back-off collision avoidance scheme. We present a Markov chain model that captures the average temporal dynamics of the network and derive flow equations that depict the behavior of the network as a function of time. Through simulations, we show that the flow equations match the network dynamics very accurately. Finally, we analyze the performance of the protocol using these flow equations.

The rest of the chapter is organized as follows. In the next section, we review the assumptions made and define the performance metrics of interest for the one-hop one-shot data collection problem. In Section 6.3, we model and analyze slotted Aloha with binary exponential back-off.
We evaluate its performance for the one-hop one-shot data collection problem in Section 6.4 and summarize the chapter in Section 6.5.

6.2 Problem Description

In this section, we review the assumptions made and define the performance metrics of interest.

The number of contending nodes in the radio range of the sink is $N$. We assume that time is split into slots of equal length and that all contending sensor nodes (the sink is not a contending node) in the network are time synchronized to transmit at the beginning of each time slot. The time required to transmit a packet is equal to that of a single time slot duration. When more than one node transmits its packet in the same time slot, it results in a collision, otherwise, if a single node transmits, it results in a successful transmission. The sink informs the sensor nodes of the either case through acknowledgements. We assume that the acknowledgement packet have a negligible effect on the protocol performance.

The following two performance metrics are of interest for the one-hop one-shot data collection problem:

- $\Delta_{O_S D}(N)$: The number of time slots required for the sink to successfully receive packets from all the $N$ contending nodes.

- $\Sigma_{O_S D}(N)$: The energy consumption by the contending nodes in the sensor network for the sink to successfully receive packets from all the $N$ contending nodes. The energy model we consider is as described in Chapter 5.

In the next section, we model and analyze slotted Aloha with binary exponential back-off for the one-hop one-shot data collection problem.
6.3 Slotted Aloha with Binary Exponential Back-Off

In this protocol, each node starts out with a minimum congestion window of size \( W_0 \) and each time it backs-off, it doubles the size of its congestion window up to a maximum value of \( W_{MAX} \), leading to a binary exponential increase in its window size. At each stage of the increase, the node chooses a time slot to transmit uniformly at random within the current window size. The congestion window size of the node remains constant once the maximum window size is reached.

If the number of stages of increase before the maximum window size is reached is \( M \), then the window size at stage \( i \), \( (0 \leq i \leq M - 1) \) is \( W_i = 2^i W_0 \), and \( W_{MAX} = W_{M-1} \).

![Markov chain of states for a contending node using slotted Aloha with binary exponential back-off protocol.](image)

Figure 6.1: Markov chain of states for a contending node using slotted Aloha with binary exponential back-off protocol.

Figure 6.1 shows the Markov chain of states a system using this protocol goes through before all packets are transmitted to the sink through state \( S \). The state \((i, j)\) implies that the node has entered the stage \( i \) after backing-off from stage \((i - 1)\), \( j \) back-off time counter slots ago.

We assume that the back-off time counter length for each node in the network is equal to the time slot length.

Let \( n_{i,j}(t) \), \( (i \in [0, M - 1], j \in [0, W_i - 1]) \) be the number of nodes in stage \( i \) at time \( t \) that have entered this stage \( j \) time slots ago. The probability that a node in state \((i, j)\) attempts to transmit its packet, given that it is in this state is given by,
\[ p_{i,j} = \frac{1}{W_i - j} \]  

(6.1)

All nodes in the network start at time slot 0 at state \((0, 0)\), such that \(n_{0,0}(0) = N\). However, \(\forall \ t > 0\), \(n_{0,0}(t) = 0\), as all nodes move to a different state in the next time slot and do not return to state \((0, 0)\).

Whenever a node attempts to transmit in a time slot, either it backs-off due to collisions or packet errors, or successfully delivers its packet to the sink. Therefore, the number of nodes that enter state \((i, j), \ j \neq 0\), at \(t + 1\) is equal to the average number of nodes in state \((i, j - 1)\) that do not attempt to transmit at \(t\). And all nodes in state \((i, j), \ j \neq 0\), at time \(t\) leave that state either through successful transmission or back-off or by just moving to state \((i, j + 1)\) at time \(t + 1\). Thus,

\[ n_{i,j}(t + 1) = n_{i,j-1}(t)(1 - p_{i,j-1}) \]  

(6.2)

The average number of nodes that enter state \((i, 0), \ 0 < i < M - 1\), at time \(t + 1\) is equal to the sum of the average number of nodes that back-off from all states \((i - 1, j)\) in the previous back-off stage \(i - 1\), at time \(t\). And all nodes in state \((i, 0)\) at time \(t\) would leave to other states by time \(t + 1\). Therefore, for \(i \neq 0, M - 1\),

\[ n_{i,0}(t + 1) = \sum_{q=0}^{W_i-1} n_{i-1,q}(t)p_{i-1,q}[1 - \pi_{i-1,q}(t)] \]  

(6.3)

where,

\[ \pi_{i,j}(t) = (1 - p_{i,j})^{n_{i,j}(t)-1} \prod_{k=0}^{M-1} \prod_{l=0}^{W_k-1} (1 - p_{k,l})^{n_{k,l}(t)} \]  

(6.4)
The number of nodes that enter state \((M-1,0)\) at time \(t+1\) is equal to the average number of nodes that back from all states of stages \(M-2\) and \(M-1\) at time \(t\). Therefore,

\[
n_{M-1,0}(t+1) = \sum_{q=0}^{W_{M-1}-1} n_{M-1,q}(t)p_{M-1,q}[1 - \pi_{M-1,q}(t)] + \sum_{q=0}^{W_{M-2}-1} n_{M-2,q}(t)p_{M-2,q}[1 - \pi_{M-2,q}(t)] \quad (6.5)
\]

Consequently, the average number of nodes that enter state \(S\) at time slot \(t+1\) is equal to the sum of the average number of successful deliveries from all states \((i,j)\) in the Markov chain of Figure 6.1 to the sink.

\[
n_s(t+1) = n_s(t) + \sum_{i=0}^{M-1} \sum_{j=0}^{W_i-1} n_{i,j}(t)p_{i,j}\pi_{i,j}(t) \quad (6.6)
\]

The expected delay \(\Delta_{OSD}(N)\) is such that \(n_s(\Delta_{OSD}(N)) = N\).

The expected energy consumption per node for the successful reception of \(N\) packets by the sink is:

\[
\Sigma_{OSD}(N) = \frac{1}{N} \sum_{t=0}^{\Delta_{OSD}(N)} \sum_{i=0}^{M-1} \sum_{j=0}^{W_i-1} (n_{i,j}(t)p_{i,j}\xi_T + (n_{i,j}(t)(1-p_{i,j})\xi_R) \quad (6.7)
\]

\[
\Sigma_{OSD}(N) = \frac{1}{N} \sum_{t=0}^{\Delta_{OSD}(N)} \sum_{i=0}^{M-1} \sum_{j=0}^{W_i-1} n_{i,j}(t)(p_{i,j}\xi_T + (1-p_{i,j})\xi_R) \quad (6.8)
\]

where \(n_{i,j}(t)p_{i,j}\) and \(n_{i,j}(t)(1-p_{i,j})\) are the expected number of transmissions and receptions respectively in time slot \(t\) from state \((i,j)\). Since for each sensor node the energy consumption in almost equal for both transmission and reception, \(\xi_R \approx \xi_T\). This reduces the expected energy consumption to
\[ \Sigma_{\text{OSD}}(N) = \frac{1}{N} \Delta_{\text{OSD}}(N) \sum_{t=0}^{M-1} W_{i-1} \sum_{i=0}^{M-1} \sum_{j=0}^{n_{i,j}(t)} \] (6.9)

One of the most important aspects of counting the number of nodes at each state by the above, average expressions is the handling of fractional values. If \( n_{i,j}(t) \) is a real number less than one, then that number can be approximately assumed to be the probability of existence of a node at state \((i, j)\). All real numbers greater than one are used without change. It should be noted the number of states with fractional values of \( n_{i,j}(t) \)'s, increases with increase in their number.

### 6.4 Performance Evaluation

In this section, first we verify the accuracy of the flow equations of the previous section using simulations and then evaluate the performance of slotted Aloha protocol with binary exponential back-off for the one-hop one-shot data collection problem.

For simulations, the value of \( N \) is varied from 2 to 100 to capture the network dynamics from low to high densities. The results are averaged over 100 random trials, with ten different random seeds. Figure 6.2 plots the analysis and simulation results for the number of nodes in each of the \( M \) back-off stages as a function of time, for \( M = 5 \) and \( W_0 = 32 \). The number of nodes \( n_i(t) \) in stage \( i \) at time \( t \) is calculated as:

\[ n_i(t) = \sum_{j=0}^{W_i-1} n_{i,j}(t), \quad 0 \leq i \leq M - 1 \] (6.10)

The above figure shows that the analysis matches with simulations very well, with one virtually superimposed over the other. It is important to note that the analysis presented in the previous section is first-order in nature and thus is approximate. Exact analysis of the system dynamics would involve tracking the probability distribution of nodes in each state of Figure 6.1,
Figure 6.2: Comparison of analysis and simulations for slotted Aloha with binary exponential back-off.

as a function of time. However, the above figure shows that the approximate analysis of the previous section works well and is valuable in this setting.

It can be seen from this figure that, the final back-off stage contributes the maximum to the delay. For example, the first four stages contribute close to half of the delay and the final stage (stage 4) contributes the other half of the delay. The reason for this is that, the low probability of transmission in the final stage accompanied by the reducing number of contending nodes in the network, increases the number of idle time slots. This implies that the delay can be reduced either by reducing the number of back-off stages or by reducing the initial back-off window or both. This intuition is confirmed by Figure 6.3 which plots the expected delay and energy consumption as a function of the number of contending nodes for different values of $M$ and $W_0$.

The main observations from the figure are as follows.

- For a given initial window size, (Figures 6.3(a) & (c)), there exist lower and upper threshold number of nodes between which the performance due to a particular value of $M$ is better than other values. The delay is reduced up to that value of $M$ because of the reduction in the number of idle time slots in the final back-off stage by implicitly increasing the probability of transmission. Lower delay implies that the rate at which nodes transition
into the absorbing state $S$ is higher and this implies lesser number of nodes in other states per time slot. Thus, lower number of nodes in the transition states $(i, j)$ implies lower energy consumption according to Equation 6.9. For $M = 1$, where the back-off window is constant, the performance of slotted Aloha in terms of both delay and energy is lower than that for $M > 1$ for low densities and higher for high densities. Also from the figure it is evident for node numbers greater than 100 the delay and energy consumption for $M = 2$ increases beyond $M = 3$. This is because at high densities lower number of back-off stages implies higher number of collision thus increasing both delay and energy. This suggests
that for high densities multiple back-off stages is preferred and for low densities a single back-off stage performs better.

- Figure 6.3(b) & (d) confirm the intuition that at high node densities larger initial windows \( (W_0) \) perform better and at low node densities smaller initial windows are better. These figure also suggest the existence lower and upper thresholds of initial window sizes between which performance due to a particular \( W_0 \) value is better than other values.

6.5 Chapter Summary

In this chapter, we have investigated the performance of the slotted Aloha protocol with binary exponential back-off applied for the one-hop one-shot data collection problem that occurs frequently in many data-gathering applications of wireless sensor networks. We derived flow equations based on transient state transitions and verified their accuracy through simulations. Using these equations, we then evaluated the performance of the protocol. Results suggest the existence of a delay-energy trade-off based on the number of back-off stages and the initial window size.
Chapter 7

Location-Aware Medium Access

7.1 Introduction

Location awareness of sensor nodes is increasingly common in many wireless sensor network applications. For example, protocols such as GPSR [54] have used it to provide efficient routing. In this thesis, we propose a novel medium access protocol that makes use of the location awareness of sensor nodes to provide efficient wireless medium access.

The main idea in our protocol is the separation of collision domains of nodes using spatial partitioning. A tree-based space partitioning procedure is used to adaptively partition the space until each node can transmit its packet successfully, without collisions. The key point here is that spatial partitioning allows us to leverage the location distribution of sensor nodes to provide efficient medium access.

In this chapter, our focus is on the one-hop one-shot data collection problem discussed in chapter 5. The two main performance metrics of interest for this problem are the delay in obtaining packets from all contending sensor nodes in the radio range of the sink and the energy consumption incurred by the sensor network in this operation. We evaluate the performance of our location-aware medium access protocol in terms of these two performance metrics and compare it with three location-unaware medium access protocols – HT-split, optimal p-persistent
slotted CSMA, and the IEEE 802.15.4 standard MAC. We show through simulations that our protocol can take advantage of the location distribution of sensor nodes to provide significantly lower delay and energy consumption compared to location-unaware medium access protocols.

The rest of the chapter is organized as follows. In the next section we discuss related work and in Section 7.2, we describe the assumptions and metrics associated with the one-hop one-shot data collection problem. In Section 7.3, we present our location-aware medium access protocol in detail and discuss its implementation aspects. In Section 7.4, we present the protocol performance evaluation results and discuss its scope in Section 7.5. Finally we conclude and discuss the future directions of our work in Section 7.6.

### 7.2 Problem Description

In this section, we describe the assumptions made and performance metrics associated with the one-hop one-shot data collection problem.

The one-hop sensor network has \( n \) contending sensor nodes, not including the sink (which does not contend for the channel), each with a single data packet to be transmitted. The locations of all the nodes including that of the sink are known. Time is divided into slots and each node transmits its packet only at the beginning of a time slot. The packet length is such that its transmission time is equal to one time slot. If more than one node transmits in the same time slot, it results in a collision. Otherwise, if a single node transmits in a time slot, it results in successful transmission of the packet. On successful transmission, the node is no longer in contention of the medium. The sink uses explicit acknowledgement (ACK) and negative acknowledgement (NACK) packets to indicate successful packet reception and collision, respectively, to the sensor nodes. The sink broadcasts the ACK/NACK packets as soon as the data packet(s) reception is completed. We assume that the ACK/NACK packet
is accommodated by the time slot length such that it has negligible influence on the protocol performance.

In order to study the intrinsic performance advantages of the location-aware MAC protocol, we initially isolate the random errors due to noise and wireless channel non-idealities such as multi-path fading and shadowing. Later on in the chapter we discuss the implications of channel errors on the protocol performance.

We assume an energy model in which the sensor node is either in the receive state or the transmit state until its packet is successfully transmitted after which it moves in to the shutdown state. We assume that the energy consumed by a node in the receive state is equal to that consumed in the transmit state, and that the energy consumed in the shutdown state is negligible. Also, for simplicity we assume that the energy consumed by a node in the receive/transmit state per time slot is equal to one energy unit.

We consider the following two performance metrics for the one-hop one-shot data collection problem:

1. **Expected delay** ($\Delta_{OSD}(N)$): The average number of time slots required for the sink to successfully receive packets from all contending sensor nodes in the one-hop network.

2. **Expected energy consumption** ($\Sigma_{OSD}(N)$): The amortized energy consumption per sensor node for the sink to successfully receive packets from all contending nodes. This metric is calculated by first evaluating the energy consumed by all nodes until all packets are successfully transmitted and then by dividing that value by the number of nodes that participated in this operation.

### 7.3 Location-Aware MAC Protocol

Now, we describe the our location-aware MAC protocol and illustrate its working through examples.
Figure 7.1: Example of the location-aware MAC protocol for $m = 4$. (a) The square space splitting (b) The corresponding tree.

The main idea in our protocol is a tree-based splitting of space that adaptively reduces the collision domain of sensor nodes until each node is able to transmit its packet successfully. The protocol starts out by splitting the space in the radio range of the sink into $m$ equal partitions. Each partition is a separate collision domain. At each step, only nodes belonging to the current partition are allowed to transmit their packets. When a partition has more than one node their transmission leads to collision. In the event of a collision the current partition is further split into $m$ equal partitions. This continues until the current partition has at-most a single node in it. The protocol moves onto the next partition after all nodes in the current partition have successfully transmitted their packets. This process of space splitting builds a tree with $m$ branches at each split, where, each branch is a separate collision domain. The leaves of the tree are collision domains with at-most a single sensor node in them and therefore successful transmissions can take place only from the leaves of the tree.

We illustrate our location-aware protocol through an example shown in Figure 7.1. Here the space is a square whose half-diagonal is equal to the radio range of the sink (the sink is located at the center of the square). In this example the space is split into $m = 4$ equal square partitions at each level. Figure 7.1(a) shows the square space splitting and Figure 7.1(b) shows the

---

1In this chapter we focus on symmetrical square space partitions that are multiples of 4. The value of $m$ in this case has to be power of 4. Results for values of $m$ that split the space into other shapes (such as into rectangles when $m$ is 2, 8, 32, etc.) are not presented due to lack of space.
corresponding tree. The space contains 14 sensor nodes numbered 1 through 14. The numbers in the tree show the nodes involved in collision at each branch. At time slot 1, the space is split into 4 equal squares and nodes 1, 2, and 3 transmit their packets as all of them belong to partition 1 at level 1. Since this results in collision, partition 1 of level 1 is further split into 4 equal partitions.

Now, each partition has a single node. Therefore, node 1 successfully transmits its packet at time slot 2, node 2 at time slot 3 and node 3 at time slot 4. Time slot 5, allotted to partition 4 at level 2 of partition 1 at level 1, goes idle because it does not have any nodes in it. Similarly, nodes 4, 5, and 6 collide at time slot 6 and transmit successfully in time slots 7, 8, and 9 respectively. Time slot 10 goes idle as there are no nodes in partition 4 at level 2 of partition 2 at level 1. Nodes 7, 8, and 9 collide at time slot 11 and after time slot 12 goes idle, they successfully transmit their packets at time slots 13, 14, and 15, respectively. At time slot 16, nodes 10, 11, 12, 13 and 14 belonging to partition 4 at level 1 transmit their packets and collide, leading to further splitting of that partition into 4 partitions at level 2. Since partitions 1, 2, and 3 at level 2 have nodes 10, 11, and 12, respectively, a single node each, all of them transmit their packets successfully at time slots 17, 18, and 19 respectively. At time slot 20, nodes 13 and 14 transmit their packets and collide. This results in further splitting of partition 4 at level 2 of partition 4 at level 1. Due to absence of nodes in partitions 1, 2, and 3 at level 3, time slots 21, 22, and 23 go idle. In time slot 24, nodes 13 and 14 transmit again, collide, and the partition is further split into 4 partitions at level 4. Due to absence of nodes in partition 1 at level 4, time slot 25 goes idle. Finally, nodes 13 and 14 successfully transmit their packets in time slots 26 and 27 respectively.

Thus, in the above example, the delay for the sink to receive packets from all the 14 sensor nodes is \( D(n) = 27 \) time slots. Also, since the space is split into 4 equal square partitions at

\[2\text{We follow the convention of counting partitions from left to right and bottom to top.}\]
each level we call it a 4-split strategy. Similarly, Figure 7.2 illustrates the space splitting for 16-split and 64-split strategies.

7.3.1 Implementation Aspects

A key aspect in the implementation of our location-aware MAC protocol is the determination of nodes that belong to the current partition. This can be achieved by issuing location tokens that contain the boundaries of the current partition to the nodes in each time slot. The location tokens are generated using the Location Token Generator (LTG), shown in Figure 7.3 for the $m$-split strategy, where $m$ is a power of 4. The LTG uses the current splitting level, the partition numbers of all the levels, the sink location and its radio range to determine the boundaries of the current partition. The equations show that the boundaries are calculated relative to the lower left corner of the square space.

The implementation of the protocol depends on where the location tokens are generated – at the sink or at the sensor nodes. In the former, the sink has to run the LTG and transmit the location token to the sensor nodes. This can be achieved by piggy-backing the location tokens on the ACK/NACK packets. In the latter, the sensor nodes themselves run the LTG and generate the location token at the beginning of each time slot. The advantage of the latter
LTG($L, \{P(l) : 1 \leq l \leq L\}, (s_x, s_y), S)$:

\[
x_1 = \left(s_x - \frac{S}{2}\right) + \sum_{l=1}^{L} \left\lfloor \frac{P(l) - 1}{\sqrt{m}} \right\rfloor \cdot \frac{S}{(\sqrt{m})^l}; \quad x_2 = x_1 + \frac{S}{(\sqrt{m})^L};
\]

\[
y_1 = \left(s_y - \frac{S}{2}\right) + \sum_{l=1}^{L} \left\lfloor \frac{P(l) - 1}{\sqrt{m}} \right\rfloor \cdot \frac{S}{(\sqrt{m})^l}; \quad y_2 = y_1 + \frac{S}{(\sqrt{m})^L};
\]

Return $(x_1, x_2, y_1, y_2)$;

- $L$: current level in the space splitting tree.
- $P(l)$: partition number at level $l$ for the current partition.
- $(s_x, s_y)$: location coordinates of the sink.
- $S$: side length of the square whose half-diagonal is equal to the radio range of the sink.
- $x_1$ is the left vertical boundary, $x_2$ is the right vertical boundary, $y_1$ is the lower horizontal boundary, and $y_2$ is the upper horizontal boundary.

Figure 7.3: Location Token Generator for symmetrical square $m$-split strategy ($m$ is a power of 4).

over the former is the small size of the ACK/NACK packets. This advantage is obtained at the cost of shifting the computational load of LTG from the sink to the sensor nodes.

Nevertheless, in either case, each sensor node decides if the location token belongs to it by verifying if its location falls within the boundaries specified by the location token\(^3\). If the location token belongs to a node it transmits its packet, otherwise, it is ignored. Figure 7.4 shows the sink and sensor node state diagrams for the location-aware MAC protocol for the implementation in which the sensor nodes determine the location tokens by themselves.

\(^3\)For the case of one-shot data querying the sensor node has to decide if it satisfies the query simultaneously. If the node does not satisfy the query the node does not transmit any packet.
Receive message from Radio

Successful Reception

Collision

Transmit ACK.

pRx++

Yes

STOP

Is pRx = k?

NO

Transmit NACK.

Receive message from Radio

Successful Reception

Collision

Transmit ACK.

pRx++

Yes

STOP

Is pRx = k?

NO

Transmit NACK.

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Location Token = LTG(L,…)

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?

YES

L=1, P(L) = 1

YES

YES

Leave Network

Is Location Token for me?

Receive Request from Sink?
results are averaged over 1000 random trials with 100 different random seeds. In each random
trial the locations of the sensor nodes are different.

7.4.2 Performance of Location-Aware MAC Protocol

Now we discuss the performance of our location-aware MAC protocol in terms of the delay and
energy consumption metrics discussed in Section 7.2.

Figure 7.5(a) shows the results for grid-random deployment of nodes for 4-split, 16-split,
and 64-split strategies. The delay is in number of time slots and the energy consumption is in
terms of the energy units described in Section 7.2. According to the figure, 4-split and 16-split
strategies perform much better than 64-split strategy for both low and high node densities. The
figure also suggests that while for low densities 4-split performs better for high densities 16-split
provides lower delay and energy consumption. This is expected because, for high node density,
the 4-split strategy is too conservative leading to many partition levels and thus higher delay
and energy consumption. Whereas for low node density the 16-split strategy is too aggressive
leading to many idle time slots, again leading to higher delay and energy consumption.

Figure 7.5(b) plots the delay and energy consumption as a function of $n$ for the 16-split
strategy for the three random location distributions. As the figure shows, our location-aware
protocol performance improves with increasing order in the location distribution of nodes from
uniform random to even random to grid random suggesting that our protocol is inherently de-
dsigned to take advantage of the nodes' location distribution. It should be noted in the figure that
the delay and energy consumption remain constant with increasing number of sensor nodes for
higher node densities. The reason for this is that, for grid-random deployment, with increasing
number of sensor nodes, the node density becomes more uniform across all split partitions. The

---

4For the rest of the evaluation we consider the performance of only 4-split and 16-split as the delay and energy
consumption due to 64-split is an order of magnitude higher.
corresponding partition levels remain constant at high node densities irrespective of the actual number of nodes once a certain node density is crossed.

Next, we present results of a comparative study between our location-aware protocol and three other location-unaware protocols.

### 7.4.3 Comparative Study

We consider three location-unaware protocols:

1. **HT-Split**: We have chosen to compare the performance of location-aware MAC protocol to that of the HT-split protocol to show that our location-aware MAC protocol, in addition to taking advantage of collision domain separation like the HT-split protocol, also takes advantage of the nodes’ location distribution, to provide lower delay and energy efficiency.

2. **Optimal p-persistent Slotted CSMA**: In the p-persistent slotted CSMA protocol ([21], [25]), each contending node senses the channel at the beginning of each time slot and if the channel is free it transmits its packet with probability $p$. If the channel is not free, the node attempts to transmit its packet in the next available free time slot with probability
Figure 7.6: Expected delay and expected energy consumption for HT-split, optimal p-persistent
slotted CSMA and IEEE 802.15.4 standard MAC.

\[ p \] When the packet length is equal to that of a single time slot, this protocol is identical
to p-persistent slotted Aloha. Even if the packet length is equal to multiple time slots,
the packet-length-normalized delay and energy consumption will be identical to that of
p-persistent slotted Aloha [25].

The p-persistent CSMA protocol is optimized for delay and energy consumption if the
transmission probability is equal to the inverse of the current number of contending nodes
i.e., \( p = \frac{1}{k} \) where \( k \) is the current number of contending nodes [25]. We use this optimal
p-persistent slotted CSMA protocol as a benchmark.

3. **IEEE 802.15.4**: In order to compare our location-aware MAC protocols’ performance
with a state-of-the-art MAC protocol for sensor networks, we chose the recently standardized
IEEE 802.15.4 protocol for low-rate, low-power personal area networks described in
Chapter 5.

Figure 7.6 shows the expected delay and energy consumption for the above three location-
unaware protocols as a function of \( n \). Clearly, for all node densities the IEEE 802.15.4 standard
MAC protocol performs the worst. The reason for this is that, for high number of nodes,
due to multiple back-offs, most nodes quickly reach the highest back-off stage which has the
lowest probability of transmission. The advantage of low probability of transmission is off-set
Figure 7.7 compares the performance of the four protocols in terms of expected delay and energy consumption as a function of the number of nodes (n) for the one-hop one-shot data collection problem. The delay is in number of time slots and the energy consumption is in terms of the energy units described in Section 7.2. The main observations can be summarized as follows:
• The 4-split location-aware MAC protocol performs better (for grid and even random deployments) than or equal (uniform random deployment) to that of the optimal p-persistent CSMA protocol both in terms of delay and energy consumption. This is significant because in optimal p-persistent CSMA knowledge of the current number of contending nodes is required at each step, but even in the worst case (uniform random deployment) with 4-split location-aware MAC the same optimal performance is obtained without any knowledge of the number of nodes. This performance gain can be observed for both low and high node densities.

• The 16-split location-aware MAC performs even better than 4-split MAC for grid random deployment for high node densities. For grid-random deployment the performance gains for high node densities are in the order of 60% for both delay and energy consumption for location-aware MAC compared to that of the optimal location-unaware MAC.

• The performance gains for the location-aware MAC protocols are higher for higher node densities. This is because the advantage due to location distribution becomes more significant for higher node density.

7.5 Discussion

In this section, we discuss various issues inherent to our location-aware MAC protocol and elaborate on its scope.

1. **Partition Shape:** We have illustrated and evaluated the performance of the location-aware MAC protocol for the case in which the space is symmetrically partitioned into \( m \) equal squares at each level. However intuition suggests that the shape of space splitting does not affect the performance of the location-aware MAC protocol as long as the its tree structure remains the same. For the same number of nodes and the same node location
distribution, if the space is considered to be circular and if it is split into \( m \) equal sectors at each level (as shown in Figure 7.8(a) for \( m = 4 \)), then, on an average, the delay and energy consumption of nodes would remain the same as that for square splitting. This intuition is verified by the simulation results shown in Figures 7.8(b).

2. **Sink Location**: The location of the sink is a crucial part in the implementation of our location-aware MAC protocol. However, for certain one-hop one-shot data collection applications such as localization [101], the location of the sink is not available. In fact, the application has to determine the location of the sink. This problem can be solved by first assuming an approximate location for the sink and then using the location-aware MAC protocol to obtain data-packets from the sensor nodes. We propose to use transmission power control for this purpose.

The main idea here is that the sink assumes the location of the nearest sensor node and uses this location to obtain packets from other contending nodes in its radio range. The sink can obtain the location of the nearest sensor node by using power control, in which, its transmission power is incremented by small steps starting from the lowest power until it is able to reach a sensor node and receive a packet from it. In the possibility of the
existence of more than one nodes in the lowest connected radio range of the sink, the nodes can contend for the channel using a random medium access scheme such as p-persistent slotted CSMA. The key observation here being that, for typical node densities, the number of nodes in the lowest connected radio range of the sink is very low compared to that of in typical operational radio ranges.

For example, Tmote-sky [5] devices have a radio range of above 100 m for the highest transmission power of 0 dBm. For the lowest transmission power of −25 dBm the radio range is less than 6 m. Thus, for a uniform node density, the number of nodes in the lowest-power connected radio range is at-least two orders of magnitude lower than the number in the highest-power radio range.

For the case in which more than one node exists in the lowest connected radio range of the sink, the delay in determining its nearest sensor node is the delay for the first packet to reach the sink. This delay depends on the density of node distribution, the topology of the network and the reliability of the wireless channel. If the number of nodes in the lowest-power connected radio range is $a$ and if the nodes use p-persistent slotted CSMA [25] with $T$ back-off slots (since each node chooses uniformly at random to transmit, the probability of transmission at each time slot is $p = \frac{1}{T}$), then the expected number of time slots for the first successful transmission is given by:

$$ap(1 - p)^{a-1} + 2(1 - ap(1 - p)^{a-1})ap(1 - p)^{a-1} + 3(1 - ap(1 - p)^{a-1})^2ap(1 - p)^{a-1} + \cdots$$

$$= \frac{1}{ap(1 - p)^{a-1}} (7.1)$$

Figure 7.9 shows the behavior of the above equation for different values of $a$ and $T$. As the figure shows, the value of $T$ can be chosen such that the delay due to determination
Figure 7.9: The expected number of time slots for the first successful transmission.

of the sink’s location is very low, usually much lower than 10 time slots. A thorough investigation into the influence of error in sink location on the protocol performance is left open for future work.

3. **Channel Errors**: In our location-aware MAC protocol packet losses due to channel errors should be treated differently than that due to collisions, in order to maintain the performance gains over location-unaware MAC protocols. Packet loss due to channel error should not lead to further partitioning of the space. Space partitioning should happen only when more than one node transmit its packet that results in a collision. In order to achieve this we propose to use different NACKs - NACK\_COL and NACK\_ERR - which identify collision or erroneous packet respectively. The sink sends the appropriate NACK packet based on the event that occurs. On receipt of NACK\_COL the location token generator (LTG) uses the previous token without generating a new token. Whereas receipt of NACK\_ERR prompts the LTG partitions the space and generates a new token. The loss of ACK/NACK packets due to packet errors is a relatively rare occurrence in hop networks given their small form factor. Their loss due to collisions is not a possibility as the sink is the only transmitter of those packets in the one-hop scenario.

4. **Capture Effect**: In the presence of capture effect, when two or more nodes transmit in the same time slot and the signal to noise ratio (SNR) of a packet is above a threshold
than that of other packets then that packet is received successfully by the sink. Therefore in the presence of capture effect our location-aware MAC protocol could cause starvation at some nodes. Further research to efficiently solve this problem is left open for future work.

5. **Hybrid**: In the location-aware MAC protocol we have illustrated and evaluated, the space is split into $m$ equal parts at each level. However, the value of $m$ can be changed, adaptively, at each level, depending on the sensor node deployment density. For example, for a given sensor node density, the number of nodes in partitions of higher split levels is lower than that in lower split levels. This fact can be taken advantage to adaptively reduce the value $m$ for higher split levels, thus, reducing the number of idle time slots and consequently reducing the delay. Or alternatively, since lower number of nodes contend for the channel at higher split levels, random medium access techniques such as CSMA/CA could be used in conjunction with the location-aware MAC protocol and potentially reduce the delay and energy consumption.

### 7.6 Chapter Summary

In this chapter, we have presented a novel location-aware medium access protocol for the one-hop one-shot data collection problem in wireless sensor networks. The defining feature of this problem is that each sensor node in the one-hop radio range of the sink has a single packet to transmit. We illustrated the working of our protocol using examples and discussed its implementation aspects. The main idea in our location-aware protocol is a tree-based hierarchical partitioning of space to progressively reduce the collision domains of nodes until there are no collisions.

Further, we presented results from a thorough performance evaluation of the location-aware protocol in comparison to three location-unaware MAC protocols – HT-split, optimal $p$-persistent slotted CSMA, and the IEEE 802.15.4 standard protocol – using simulations. We
evaluated the protocol for three different location distributions of nodes – uniform-random, even-random, grid-random. Results showed that our location-aware MAC protocol in the worst case (uniform random distribution) provides as good a performance as an optimal location-unaware MAC protocol and in the best case (grid random distribution) provides 60% lower delay and energy consumption.
Chapter 8

Enhancement of IEEE 802.15.4 MAC Protocol

8.1 Introduction

IEEE 802.15.4 is an important standard for low-rate low-power wireless personal area networks that is in increasing commercial use for a diverse range of embedded wireless sensing and control applications. The standard provides specifications for both the physical layer and the medium access control (MAC) protocol.

We characterize the performance of the IEEE 802.15.4 MAC for one-hop data collection in a star topology where there are multiple transmitters and a single receiver. Our primary focus is on settings where the number of transmitters is large. Because 802.15.4-enabled devices are meant to be low-cost and operate at relatively low rates, such dense deployments are of interest in many sensing applications involving these devices.

We model the IEEE 802.15.4 as a p-persistent CSMA with changing transmission probability $p$. We derive the optimal transmission probabilities to maximize the throughput and minimize energy consumption in p-persistent CSMA. We show that, particularly for large number of transmitters, the ratio of the expected idle time between successful receptions to the expected time between successful receptions is a constant for a given packet size when the transmission probabilities are optimal. Further, we find that when the transmission probability is lower...
(higher) than the optimal, the ratio is higher (lower) than this constant. This yields a distributed channel feedback-based control mechanism that changes the transmission probabilities of nodes dynamically towards the optimal. We develop an enhanced version of the IEEE 802.15.4 MAC protocol using this feedback scheme.

In our modeling and evaluation, we consider two extremes of the one-hop data collection spectrum in dense sensor networks: one-shot and continuous data collection. In one-shot data collection, each node sends only a single packet (this could be the response to a one-shot query) and once that packet is transmitted the node is no longer in contention for the channel. In continuous data collection, we assume that each node is backlogged, i.e. always has a packet to transmit.

In both cases, we find that the IEEE 802.15.4 protocol performs poorly in dense settings, showing a steep reduction in throughput and increase in energy with network size. In contrast, the enhanced protocol that we propose is significantly more scalable, showing a relatively flat, slow-changing total system throughput and energy as the number of transmitters is increased.

In this chapter we mainly focus on dense sensor networks in which at-least 50 nodes contend for the channel in either scenario. We assume that the packet lengths are deterministic and constant.

The rest of this chapter is organized as follows. In Section 8.2, we present an overview of IEEE 802.15.4 and model it as a p-persistent CSMA with changing p. In Section 8.3, we present the modeling and optimization of p-persistent CSMA and characterize the performance of the IEEE 802.15.4 MAC in Section 8.4. In Section 8.5, we present a channel feedback-based medium access control technique and adapt it to present the enhanced IEEE 802.15.4 MAC. In the same section we discuss directions of our future work. We conclude in Section 8.6.
8.2 IEEE 802.15.4

In this section we present a p-persistent CSMA MAC model for the IEEE 802.15.4 MAC protocol in which the probability of transmission $p$ changes with each collision.

8.2.1 Model

Now, we model the IEEE 802.15.4 MAC as a p-persistent CSMA MAC with changing $p$. Before we present the MAC model, we describe the assumptions made and the energy model used.

- **Assumptions**: Let the number of sensor nodes in the radio range of the sink be $N$. All sensor nodes are synchronized to a global time which is divided into slots of equal length and each node transmits at the beginning of a time slot. Let the packet length be $L$ time slots. A sensor node is informed of its packets’ successful transmission through acknowledgement packets (ACKs) from the sink. Failure to receive an ACK from the sink implies a collision. The ACK is sent by the sink as soon as the packet reception is completed. Table 8.1 summarizes the notations used.

- **Energy Model**: According to the IEEE 802.15.4 standard a node can exist in any one of the following four states - Shutdown, Idle, Transmit, Receive. For CD, we assume that the nodes are either in the Transmit or the Receive state and are not concerned with the Shutdown or Idle states. For OSD, again each node is either in the Transmit or the Receive state until its packet is transmitted, after which the node moves to the Shutdown state permanently. Let the power consumed in the Transmit state be $\xi_T$ and the power consumed in the Receive state be $\xi_R$. According to [4], for the CC2420 IEEE 802.15.4 complaint radio, $\xi_R = 35$ mW and $\xi_T = 31$ mW for the highest transmission power. The power consumed in the Shutdown state is negligible.

- **MAC Model**: In [81] the authors model the IEEE 802.15.4 MAC in the contention access period (CAP) as a non-persistent CSMA with back-off. They approximate the three
original uniform-random back-off windows to geometrically distributed back-off windows with parameters $p_1$, $p_2$ and $p_3$ such that $p_i = \frac{2}{BO_i+1}$, $(1 \leq i \leq 3)$ where $BO_i$ is the original uniform-random back-off window size. With $BO_1 = 8$, $BO_2 = 16$ and $BO_3 = 32$, the respective values of $p_1$, $p_2$ and $p_3$ are $\frac{1}{4.5}$, $\frac{1}{8.5}$ and $\frac{1}{16.5}$.

In this chapter we further simplify this model to a p-persistent CSMA in which the probability of transmission changes from $p_1$ to $p_2$ to $p_3$ with each collision and remains constant after two collisions at $p_3$. The key difference in our model from the non-persistent CSMA model is that in our case the transmission probability changes with a packet collision instead of a busy carrier sense. Thus in our model, a node starts out with an initial transmission probability of $p_1$. The node senses the channel at the beginning of each time slot and if the channel is found to be free for two consecutive time slots, it transmits its packet with probability $p_1$. If the channel is busy, the node tries to transmit the packet with the same probability the next time it finds two consecutive free time slots. If more than one node transmits in the same time slot it results in a collision and if a node is involved in a collision for the first time it changes its transmission probability to $p_2$. On a second collision its transmission probability is changed to $p_3$ and it remains constant beyond the second collision.

We evaluate the accuracy of our model using simulations. The results are averaged over 1000 random trials with 100 different random seeds. For the CD scenario, we simulated the protocol for 10000 time slots for a packet length of 50 Bytes (or 5 time slots).

Figure 8.1 plots the simulation results comparing the IEEE 802.15.4 and our p-persistent CSMA model and shows that our model is reasonably accurate. Next, we determine the optimal performance of a generic p-persistent CSMA MAC with a similar time slot structure and characterize the performance of IEEE 802.15.4 MAC in comparison to that.
8.3 p-Persistent CSMA MAC

In this section we model and analyze a generic p-persistent CSMA MAC and determine the transmission probabilities that optimize its performance.

8.3.1 Overview

In a slotted p-persistent CSMA ([21]), each node senses the channel at the beginning of each time slot and if the channel is found to be free of any transmissions, it transmits its packet with a probability $p$. If the channel is not free, the node attempts to transmits its packet in the next free time slot. If more than one node transmits in the same time slot it results in a collision.

Traditionally, system dynamics due to the p-persistent CSMA protocol have been modeled using renewal theory (example [58], [64], [25], [27]). The key assumption that makes the use of renewal or regenerative models feasible is that the system attains stationarity and that the models capture the system behavior at the state. While this assumption is still true for the CD scenario, it is not true for the one-shot data scenario. Nevertheless, we observe the system at
Figure 8.1: IEEE 802.15.4 standard is modeled as a p-persistent CSMA with probability of transmission reducing in three steps – \( p_1 = \frac{1}{4.5}, \ p_2 = \frac{1}{8.5}, \ p_3 = \frac{1}{16.5} \) – with each new collision.

every successful packet transmission like in [64] and [27], for both scenarios and derive expressions for throughput, delay and energy consumption.

8.3.2 Model

We observe the system at every successful packet transmission. The time interval between two consecutive successful transmissions is defined as an *epoch*. An epoch is made up of idle time, in which the channel is free of any transmissions, collision time, in which more than one node is transmitting and a single successful transmission time which marks the end of the epoch, as illustrated in Figure 8.2.

It is important to note that, for CD, the number of nodes remain constant in all epochs. However, for OSD the number of nodes decreases by one with each passing epoch. Let \( T_n \) be the epoch delay – the time interval between two consecutive successful packet transmissions – in seconds and \( E_n \) be the energy consumption – the total energy consumed by all contending nodes – in Joules, for the epoch with \( n \) contending nodes. Then

\[
\Phi_{CD}(N) = \frac{1}{E[T_N]} \cdot (80L) \ \text{bps} \tag{8.1}
\]
Figure 8.2: An epoch illustrating the time interval between consecutive successful transmissions.

\[ \Sigma_{CD}(N) = \frac{E[E_N]}{N} \text{ Joules} \] (8.2)

\[ \Delta_{OSD}(N) = \sum_{n=1}^{N} E[T_n] \text{ seconds} \] (8.3)

\[ \Sigma_{OSD}(N) = \frac{1}{N} \sum_{n=1}^{N} E[E_n] \text{ Joules} \] (8.4)

where \( 80L \) in Equation 8.1 is the packet length in bits. Clearly, the above metrics are optimized when \( E[T_n] \) and \( E[E_n] \) are minimized. First we determine expressions for \( E[T_n] \) and \( E[E_n] \).

**Proposition 7.** For a constant packet length \( L \), the expected epoch delay for \( n \) contending nodes is given by

\[ E[T_n] = \frac{L - (L - 1)(1 - p)^n}{np(1 - p)^{n-1}} \cdot \delta \] (8.5)

**Proof.** As illustrated in Figure 8.2 the delay in an epoch is due to idle time, collision time and successful transmission time. Therefore, the expected delay in epoch \( n \), is given by

\[ E[T_n] = E[T_{Idle,n}] + E[T_{Collision,n}] + E[T_{Success}] \] (8.6)
where $E[T_{Idle,n}]$ is the expected number of idle time slots, $E[T_{Collision,n}]$ is the expected number of collision time slots and $E[T_{Success}]$ is the expected number of time slots of successful transmission. Since the packet length $L$ is a constant $E[T_{Success}]$ is equal to $L\delta$ and independent of $n$.

If $E[N_{coll,n}]$ is the expected number of collisions in an epoch with $n$ nodes, then

\begin{align}
E[T_{Idle,n}] &= (E[N_{coll,n}] + 1) \cdot E[T_{IdlePeriod,n}] \quad (8.7) \\
E[T_{Collision,n}] &= E[N_{coll,n}] \cdot E[T_{CollisionPeriod,n}] \quad (8.8)
\end{align}

where $E[T_{IdlePeriod,n}]$ is the expected number of idle time slots between two consecutive packet transmissions (collision or successful) and $E[T_{CollisionPeriod,n}]$ is the expected number of collision time slots at each collision. Owing to the constant probability of transmission $p$ within an epoch, the $IdlePeriods$ between any two consecutive packet transmissions are $i.i.d$ random variables with the same mean value. Also, since the decision to transmit in a time slot after a free channel sense is independent of the number of previous free channel senses, the number of collisions is independent of the length of $IdlePeriods$. This holds true for $CollisionPeriods$ also, thus justifying the above two equations.

$E[N_{coll,n}]$ and $E[T_{IdlePeriod,n}]$ are given by [27]:

\begin{align}
E[N_{coll,n}] &= \frac{1 - (1 - p)^n}{np(1 - p)^{n-1}} - 1 \quad (8.9) \\
E[T_{IdlePeriod,n}] &= \frac{(1 - p)^n}{1 - (1 - p)^n} \cdot \delta \quad (8.10)
\end{align}

We use the above two equations to derive the expected delay in the epoch $n$. Since the packet length is constant $E[T_{CollisionPeriod,n}] = L\delta$. Therefore,
\[ E[T_{idle,n}] = \frac{1-p}{np} \cdot \delta \]  \hspace{1cm} (\text{8.11})

\[ E[T_{Collision,n}] = \frac{L\delta(1-(1-p)^n-np(1-p)^{n-1})}{np(1-p)^{n-1}} \]  \hspace{1cm} (\text{8.12})

Substituting the above equations in Equation 8.6, we get Equation 8.5.

\[ \square \]

\textbf{Proposition 8.} For a constant packet length of \( L \), the expected epoch energy consumption for \( n \) contending nodes is given by

\[ E[E_n] = \xi R \delta \cdot \frac{L - (L-1)(1-p)^{n-1}}{p(1-p)^{n-2}} + \xi T \delta \cdot \frac{L}{(1-p)^{n-1}} \]  \hspace{1cm} (\text{8.13})

\textbf{Proof.} Similar to Equation 8.6, the energy consumption in the epoch \( n \) is equal to the sum of the energy consumption in idle time, the energy consumption in collision time and the energy consumption in a successful transmission.

\[ E[E_n] = E[E_{Idle,n}] + E[E_{Collision,n}] + E[E_{Success}] \]  \hspace{1cm} (\text{8.14})

Using equations from Proposition 7, \( E[E_{Idle,n}] \) can be calculated as

\[ E[E_{Idle,n}] = (E[N_{colt,n}] + 1) \cdot n \xi R \cdot E[T_{IdlePeriod,n}] \]  \hspace{1cm} (\text{8.15})

\[ = n \xi R \delta \cdot \frac{1-p}{np} = \xi R \delta \cdot \frac{1-p}{p} \]  \hspace{1cm} (\text{8.16})

Surprisingly, for a constant \( p \), the idle time energy consumption is independent of the number of contending nodes in an epoch, and depends only on \( p \). Similarly, the collision time energy consumption is given by
\[ E[E_{\text{Collision}, n}] = E[N_{\text{coll}, n}] \cdot E[E_{\text{CollisionPeriod}, n}] \]  

(8.17)

The expected energy consumption in a \textit{CollisionPeriod}, \( E[E_{\text{CollisionPeriod}, n}] \), is equal to the sum of the expected energy consumption by nodes involved in packet transmissions and the expected energy consumption by nodes in idle state during the \textit{CollisionPeriod}. Therefore,

\[ E[E_{\text{CollisionPeriod}, n}] = L\xi_T \delta \sum_{i=2}^{n} iP\{\text{Trans.} = i \mid \text{Collision}\} + L\xi_R \delta \sum_{i=2}^{n} (n-i)P\{\text{Trans.} = i \mid \text{Collision}\} \]  

(8.18)

where \( P\{\text{Trans.} = i \mid \text{Collision}\} \) is the probability that \( i (\geq 2) \) nodes transmit their packets given that a collision has occurred, and it is given by

\[ P\{\text{Trans.} = i \mid \text{Collision}\} = P\{\text{Trans.} = i \mid \text{Trans.} \geq 2\} = \frac{\binom{n}{i} p^i (1-p)^{n-i}}{1 - (1-p)^n - np(1-p)^{n-1}} \]  

(8.19)

Substituting the above equation in Equation 8.18, we get

\[ E[E_{\text{CollisionPeriod}, n}] = \frac{L(\xi_T - \xi_R) \delta \cdot np(1 - (1-p)^{n-1})}{1 - (1-p)^n - np(1-p)^{n-1}} + nL\xi_R \delta \]  

(8.20)

Substituting the above equation in Equation 8.17, we get

\[ E[E_{\text{Collision}, n}] = \frac{L(\xi_T - \xi_R) \delta \cdot (1 - (1-p)^{n-1})}{(1-p)^{n-1}} + \frac{L\xi_R \delta \cdot (1 - (1-p)^n - np(1-p)^{n-1})}{p(1-p)^{n-1}} \]  

(8.21)

And finally, the expected energy consumption during a successful transmission is given by

\[ E[E_{\text{Success}}] = \xi_T \cdot L\delta + \xi_R \cdot L\delta \cdot (n - 1) \]  

(8.22)
Substituting the above equations in Equation 8.14 we get Equation 8.13.

8.3.3 Optimality

Let $p_{opt}^T(n, L)$ and $p_{opt}^E(n, L)$ respectively be the transmission probabilities at which $E[T_n]$ and $E[E_n]$ are minimized.

**Proposition 9.** For $n > 1$, the transmission probability that minimizes the expected epoch delay $E[T_n]$ is given by

\[
p_{opt}^T(n, L) = \frac{1}{n}, \quad L = 1 \tag{8.23}
\]

\[
p_{opt}^T(n, L) \approx \sqrt{n^2 + 2n(n - 1)(L - 1) - n}, \quad L > 1 \tag{8.24}
\]

**Proof.** The value of $p$ that minimizes $E[T_n]$ is obtained by equating its first derivative with respect to $p$ to zero.

\[
\frac{dE[T_n]}{dp} = 0 \tag{8.25}
\]

For $L = 1$,

\[
E[T_n] = \frac{\delta}{np(1 - p)^{n-1}} \tag{8.26}
\]

Taking the derivative and equating it to zero results in $p = \frac{1}{n}$. Similarly, for $L > 1$, equating the derivative of $E[T_n]$ from Equation 8.5 to zero yields the following equation.

\[
(1 - p)^n = \frac{L}{L - 1} \cdot (1 - np) \tag{8.27}
\]
For \( np < 1 \), \((1-p)^n\) can be approximated to \(1 - np - \frac{n(n-1)}{2} p^2\). Using this approximation and further simplification, Equation 8.27 reduces to Equation 8.24 as an unique root to a quadratic equation. It can be verified that \( \frac{dE[T_n]}{dp} > 0 \) for \( p = p_{opt}^T(n, L) \), thus minimizing \( E[T_n] \).

**Proposition 10.** For \( n > 1 \) and \( \gamma = \frac{E}{R} \) the transmission probability that minimizes the expected epoch energy consumption \( E[E_n] \) is given by

\[
p^E_{opt}(n, L) \approx \sqrt{\frac{n^2 + 2n(n-1)(L-1) + 4L(n-1)(\gamma - 1) - n}{n(n-1)(L-1) + 2L(n-1)(\gamma - 1)}}
\]  

(8.28)

**Proof.** Similar to the previous Proposition equating \( \frac{dE[E_n]}{dp} \) to zero yields

\[
(1-p)^n = \frac{L}{L-1} \cdot (1 - np - p^2(n-1)(\gamma - 1))
\]

(8.29)

The same approximation as in the previous proposition and further simplification of the above equation results in Equation 8.28. It can be verified, as in the previous Proposition, that the second derivative of \( E[E_n] \) with respect to \( p \) is positive for \( p = p^E_{opt}(n, L) \), thus minimizing \( E[E_n] \).

Numerical calculations show that the approximations are very close to the actual values. For \( n = 1 \), the optimum transmission probability is equal to 1; *i.e.*, when there is a single sensor node left, delay and energy are minimized when it transmits its packet with probability 1. Figure 8.3 plots \( p^T_{opt}(n, L) \) and \( p^E_{opt}(n, L) \) as a function of the number of contending nodes from \( n = 100 \) to \( n = 2 \) for different values of \( \gamma \). As the figure shows, for optimal performance the probability of transmission should increase with decreasing number of nodes in an epoch in order to avoid excessive idle time slots. We can also see that the transmission probabilities are higher for lower values of \( \gamma \). This is because if the node spends more energy in the receive state than in the transmit state, energy is saved if it transmits more than it receives.
Figure 8.3: The optimal probability of transmission.

**Corollary 3.** If $\xi_T = \xi_R$, then $p_{opt}^T(n, L) = p_{opt}^E(n, L)$, i.e., the delay and energy consumption are jointly optimized with a single probability of transmission for $\xi_T = \xi_R$.

**Proof.** For $\gamma = 1$ Equations 8.24 and 8.28 are equal, which proves the corollary.

### 8.3.4 Optimality Criteria

Now, we discuss some interesting optimality criteria for the epoch delay and energy consumption.

- **Proposition 11.** Let $\Gamma(L) = \frac{(L-1)^2}{L - \sqrt{2}L - 1}$. If $L > 1$ and $n$ is large such that $\frac{n - 1}{n} \approx 1$ then for optimal transmission probability the average epoch delay is a constant equal to $\Gamma(L)$.

  \[ p = p_{opt}^T(n, L) \Rightarrow E[T_n] \approx \Gamma(L) \]  

  **Proof.** For optimal transmission probability, substituting Equation 8.27 into Equation 8.5 we get

  \[ E[T_n] = \frac{(L - 1)(1 - p_{opt}^T(n, L))}{1 - np_{opt}^T(n, L)} \]
For $n$ large such that $\frac{n-1}{n} \approx 1$, from Equation 8.24

\[ p_{opt}^T(n, L) \approx 0 \quad (8.32) \]

\[ np_{opt}^T(n, L) \approx \frac{\sqrt{2L-1} - 1}{L-1} \quad (8.33) \]

Substituting the above equations into Equation 8.31 proves the proposition.

\[ \text{Corollary 4. For optimal transmission probability and for large number of nodes such that} \]

\[ \frac{n-1}{n} \approx 1 \text{ the throughput of p-persistent CSMA MAC protocol is a constant independent of} \]

\[ n \text{ and depends only the length of the packet.} \]

\[ \text{OptimalThroughput} \approx \frac{L - \sqrt{2L-1}}{(L-1)^2} \text{ packets/timeslot} \quad (8.34) \]

\[ \text{Proof. Throughput is calculated as the inverse of the epoch delay. Equation 8.34 is a} \]

\[ \text{direct result from Proposition 11.} \]

\[ \text{•} \]

\[ \text{Proposition 12. Let} \quad \Gamma_R(L) = \frac{L-\sqrt{2L-1}}{(L-1)(\sqrt{2L-1}-1)} \quad \text{. If } L > 1 \text{ and } n \text{ is large such that} \quad \frac{n-1}{n} \approx 1, \]

\[ \text{then for optimal transmission probability the ratio of average idle time in an epoch to the} \]

\[ \text{average epoch delay is a constant equal to } \Gamma_R(L). \text{ Also, if the transmission probability is} \]

\[ \text{greater than optimal then the ratio is lower than } \Gamma_R(L) \text{ and vice versa.} \]

\[ p = p_{opt}^T(n, L) \Rightarrow \frac{E[T_{I_{dle,n}}]}{E[T_n]} \approx \Gamma_R(L) \quad (8.35) \]

\[ p \leq p_{opt}^T(n, L) \Rightarrow \frac{E[T_{I_{dle,n}}]}{E[T_n]} \geq \Gamma_R(L) \quad (8.36) \]

\[ \text{Proof. Using Equations 8.9, 8.5 and 8.27 for optimal } p, \]
\[
\frac{E[T_{idle,n}]}{E[T_n]} = \frac{1}{L-1} \left( \frac{1}{n p_{opt}(n, L)} - 1 \right) \tag{8.37}
\]

For \( \frac{n-1}{n} \approx 1 \), using Equation 8.24

\[
np_{opt}(n, L) \approx \sqrt{\frac{2L-1}{L-1}} - 1 \tag{8.38}
\]

Substituting the above equation into the previous equation the first part of the proposition is proved.

Similarly, for \( p \leq p_{opt}(n, L) \)

\[
np \approx \sqrt{\frac{2L-1}{L-1}} - 1 \tag{8.39}
\]

\Rightarrow \frac{E[T_{idle,n}]}{E[T_n]} \geq \Gamma_R(L) \tag{8.40}

Hence the proposition is proved. \qed

Figure 8.4 illustrates Proposition 12 for \( L = 5 \). The approximation of the ratio to \( \Gamma_R(L) \) is primarily due the approximation in Equation 8.24. As the figure shows, for low values of \( n \) the ratio deviates away from \( \Gamma_R(L) \).

- Figure 8.5 plots the expected delay and energy consumption for an epoch with \( n = 50 \) nodes as a function of the transmission probability \( p \) for different values of the packet length \( L \). The figure can be explained through the following question:

  In p-persistent CSMA, if the length of the packet is increased from \( L \) to \( L + l \) \((l > 0)\), should the value of transmission probability \( p \) be increased or decreased to maintain the delay and energy consumption constant?
Figure 8.4: Ratio of expected idle time to expected epoch delay.

Figure 8.5: Expected delay and energy consumption in an epoch with \(n\) nodes as a function of transmission probability, \(p\), for different values of packet length \(L\).

Figure 8.5 shows us that the answer to the above question is that it depends on the value of \(p\). If \(p < p_{opt}^T(n, L)\), then for the same delay, \(p\) should be increased and if \(p > p_{opt}^T(n, L)\) then \(p\) should be decreased. The same answer holds true for energy if \(p_{opt}^E(n, L)\) is replaced by \(p_{opt}^E(n, L)\). The figure also shows that the optimal transmission probability values \(p_{opt}^T(n, L)\) and \(p_{opt}^E(n, L)\) decrease with increasing \(L\).

- Figure 8.6 plots ratios of consecutive epoch delays and energy consumptions as functions of \(n\). In this figure, if the ratio is greater than 1, it implies that the delay or energy value
increases with decreasing $n$ and *vice versa*. Greater the difference from 1, higher the rate of increase or decrease. The following observations can be made from the figure:

- For $p = p_{opt}^T(n, L)$, $E[T_n]$ is almost constant over all $n$. For $p > p_{opt}^T(n, L)$, $E[T_n]$ shoots up for higher values of $n$ due to higher number of collisions. For $p < p_{opt}^T(n, L)$, $E[T_n]$ shoots up for lower values of $n$ due to higher number of idle time slots.

- For $p = p_{opt}^E(n, L)$, $E[E_n]$ increases monotonically with increasing $n$. For $p > p_{opt}^E(n, L)$, $E[E_n]$ shoots up for higher values of $n$ due to higher number of collisions. For $p < p_{opt}^E(n, L)$, $E[E_n]$ is higher than the optimal energy consumption values for lower values of $n$ due to higher number of idle time slots.

---

Figure 8.6: Ratio of expected delays and energy consumptions for consecutive epochs.

For CD the implication of this criterion is that the delay between two successful packet transmissions is independent of the number of nodes in the network as long as the nodes are transmitting at optimal transmission probabilities. For OSD, the implication given by is the following proposition.
Proposition 13. For OSD, if $n$ is large such that $\frac{n-1}{n} \approx 1$, then the transmission probability is optimal if and only if the epoch of delay of two consecutive epochs are equal.

$$p = p_{opt}^T(n, L) \Leftrightarrow E[T_{n-1}] = E[T_n] \quad (8.41)$$

Proof. (i) To prove that

$$p = p_{opt}^T(n, L) \Rightarrow E[T_n] = E[T_{n-1}] \quad (8.42)$$

From Equation 8.27 $p$ is optimal when

$$(1 - p)^n = \frac{L}{L-1} \cdot (1 - np) \quad (8.43)$$

For optimal value of $p$,

$$\frac{E[T_n]}{E[T_{n-1}]} = \frac{(1 - p_{opt}^T(n, L))(1 - (n-1)p_{opt}^T(n-1, L))}{(1 - p_{opt}^T(n-1, L))(1 - np_{opt}^T(n, L))} \approx 1 \quad \text{for } \frac{n-1}{n} \approx 1 \quad (8.44)$$

We have used Equation 8.24 in the above simplification.

(ii) To prove that

$$E[T_n] = E[T_{n-1}] \Rightarrow p = p_{opt}^T(n, L) \quad (8.45)$$

Using equation 8.5, $E[T_n] = E[T_{n-1}]$ implies

$$\frac{L - (L-1)(1 - p)^n}{np(1 - p)^{n-1}} = \frac{L - (L-1)(1 - p)^{n-1}}{(n-1)p(1 - p)^{n-2}} \quad (8.46)$$

Algebraic manipulations reduce the above equation to
\[(1 - p)^n = \frac{L}{L-1} \cdot (1 - np) \quad (8.47)\]

which is the same as equation 8.27, which implies \( p = p_{opt}^T(n, L) \). Hence proved. □

### 8.4 Characterization of IEEE 802.15.4

Having determined the performance of optimal p-persistent CSMA, we characterize the performance of IEEE 802.15.4 MAC in this section.

Figure 8.7 plots the average transmission probabilities for the IEEE 802.15.4 MAC (obtained using the p-persistent CSMA model) in comparison to the transmission probabilities for optimal p-persistent CSMA for both CD and OSD. The transmission probabilities shown for IEEE 802.15.4 are obtained using the default values specified in the standard including the two required sensing slots, which is not required for the generic p-persistent CSMA MAC. For OSD, the transmission probability for IEEE 802.15.4 quickly stabilizes at \( \frac{1}{16.5} = 0.0606 \) and for CD, close to that value. This behavior is in contrast to the trend shown by optimal probabilities. This implies that the back-off mechanism of IEEE 802.15.4 protocol can be modified for enhanced performance as follows:

- The change of back-off window sizes should happen at successful transmissions instead of at collisions or busy channel senses. Further, for OSD, successful packet transmissions are a better indicator for future congestion than collisions or busy channel senses.

- For CD, the average transmission probability for IEEE 802.15.4 MAC remains almost constant irrespective of the number of contending nodes, while for optimal p-persistent CSMA it reduces with \( N \). For optimal performance the window sizes should be reflective of the number of contending nodes.
• For OSD, the back-off window size should actually decrease with every successful transmission as the optimal transmission probability increases.

Figure 8.7: Comparison of transmission probabilities for IEEE 802.15.4 and optimal p-persistent CSMA for CD and OSD.

In the next section, we present a channel feedback enhanced IEEE 802.15.4 MAC that incorporates the above features.

8.5 Enhanced IEEE 802.15.4

The key idea in enhancing the performance of the IEEE 802.15.4 is to use the optimality criteria for p-persistent CSMA derived in Section 8.3. In particular, we consider the criterion described in Proposition 12 which requires measurement of the idle time as well as the delay between two consecutive successful transmissions. These measurements can be construed as feedback from the channel. In chapter 5 we reviewed the related work in channel feedback-based medium access control techniques.

Prior work on channel feedback based medium access enhancements has been reviewed in Chapter 5. A good control mechanism should depend on the network and traffic conditions as well as the application requirements. Our objective is to present a feedback control mechanism
that is suitable for both CD and OSD scenarios. One major challenge presented in OSD is to estimate the true system state using channel conditions in the face of constantly changing state of the system (decreasing number of contending nodes). Nevertheless, the analysis presented in Section 8.3 presents us with unique opportunities to efficiently control the transmission probabilities in real time.

8.5.1 Our Approach

Our approach for channel feedback-based control of transmission probabilities is mainly based on Proposition 12. According to the proposition, if the transmission probability is optimal, then the ratio of idle time to the delay between two consecutive successful packet transmissions is $\Gamma_R(L)$. If the transmission probability if higher than the optimal value then the ratio is lower than $\Gamma_R(L)$ and vice versa.

First we describe how this optimality criterion can be used for an enhanced p-persistent CSMA and then adapt it to design an enhanced IEEE 802.15.4 MAC protocol.

8.5.1.1 Enhanced p-Persistent CSMA MAC

Each contending node can start by choosing the same transmission probability uniformly at random in a small interval of say $(0, 0.05)$. Each node in the network measures the current epoch’s idle time and delay and uses these measurements to determine the transmission probability for the next epoch. If the ratio of idle time to the delay is lower than $\Gamma_R(L)$ then it means that the transmission probability would have been greater than the optimal value. Therefore the transmission probability of the next epoch should be lower than the current epoch’s to bring the delay closer to optimal. Similarly, if the ratio if higher than $\Gamma_R(L)$ the next epoch’s transmission probability should be increased for optimal delay. Thus, the transmission probability update rule is given by
\[ p_{\text{next}} = p_{\text{current}} \cdot \frac{\alpha}{\Gamma_R(L)} \]  \hspace{1cm} (8.48)  

where \( \alpha = \frac{T_{\text{Idle, current}}}{T_{\text{current}}} \). In this update rule the increase or decrease in the transmission probability is directly proportional to the value of the ratio \( \alpha \).

### 8.5.1.2 Enhanced IEEE 802.15.4

The IEEE 802.15.4 MAC protocol uses different window sizes to control the transmission of packets. In order to use the above optimality criterion, the transmission probability update rule should be converted into a window size update rule. For this we make use of the approximation we used in Section 8.2 to model the IEEE 802.15.4 MAC as a p-persistent CSMA MAC with changing \( p \). In this, if a uniform-random back-off window has a size of \( W \) time slots then it can be closely modeled as a geometric-random choice of time slot with parameter \( p \) as long as \( p = \frac{2}{W+1} \). Thus a transmission probability can be converted into window size by using the inverse relationship, \( i.e., W = \frac{2\cdot p}{p} \). Based on this and the transmission probability update rule given above, the window update rule for the Enhanced IEEE 802.15.4 MAC is:

\[ W_{\text{next}} = \frac{(W_{\text{current}} + 1)\Gamma_R(L) - \alpha}{\alpha} \]  \hspace{1cm} (8.49)

A key aspect of this update rule is that, all nodes in the network should updated their windows at every successful packet transmission. Figure 8.8 shows the flow chart for the Enhanced IEEE 802.15.4 MAC operation at a node.

It should be noted that all aspects of the original IEEE 802.15.4 MAC have been preserved except for when the window is changed and how it is changed.
8.5.1.3 Evaluation

Figure 8.9 shows the performance gains for the Enhanced IEEE 802.15.4 MAC in comparison to the original. The figure also shows the performance of the optimal p-persistent CSMA and enhanced p-persistent CSMA. It should be noted that the performance of the enhanced IEEE 802.15.4 MAC matches that of the enhanced p-persistent CSMA MAC for CW = 0, i.e., if the nodes do not sense the channel for two consecutive free slots but transmit their packet once their
chosen time slot occurs. Thus, for the enhancement we use, the performance of the enhanced p-persistent CSMA is an upper-bound on the performance of the enhanced IEEE 802.15.4 MAC.

An important observation from the figure is that the system throughput reduces drastically with increasing number of contending nodes for the original IEEE 802.15.4 MAC. But for the enhanced version, the system throughput is almost constant with the number of nodes; implying that it is more scalable than the original. This holds for energy also. These significant gains in performance are observed for both CD and OSD scenarios.

8.5.1.4 Discussion

In actual implementation the measurement of idle time and the delay between two consecutive successful packet transmissions can be achieved easily at each node by observing ACKs from the sink. If all nodes in the network are in the radio range of each other then all nodes see the same idle time between two consecutive successful packet transmissions. If, on the other hand, all nodes are in the radio range of the sink but not in the radio range of each other, then each node sees an idle time that is based on the number of nodes in its neighborhood. Thus, the above update rule tries to optimize the transmission probability for the number of nodes in the neighborhood of each node and not for the entire network. However, the sink can measure the idle time for the entire network and piggy back this value in the ACKs to the sensor nodes. The sensor nodes measure the epoch delay as the interval between the ACKs. Thus, in this case, the channel feedback is via the sink.

The performance difference in terms of degradation or improvement, if any, between the local feedback and global feedback based mechanisms needs to be investigated. This is a direction for future work.

An important aspect of the Enhanced IEEE 802.15.4 MAC protocol is that all nodes should change their window sizes and choose a new time slot (or start a new counter) at every successful
packet transmission. Otherwise, only a few nodes optimize their window sizes and this could lead to unfairness in the CD scenario.

Another important aspect to consider is the effect of channel errors. The current standard MAC assumes channel errors based packet losses to be collisions and backs-off accordingly, thus misconstruing channel errors as congestion. But the enhanced MAC protocol does not change any protocol parameters due to channel errors based packet losses, as successful packet transmissions are taken as the only indicators of channel congestion. Nevertheless, a thorough investigation of the effect of channel errors should be addressed in future work.

In this chapter, we have focused on dense sensor networks. The following table shows the throughput performance comparison of the original and enhanced IEEE 802.15.4 MAC protocols for lower number of nodes. Clearly, according to the results, the current MAC performs better than the enhanced MAC for low number of nodes. But with increasing number of nodes, the enhanced MAC increasingly performs better.

<table>
<thead>
<tr>
<th>N</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original</td>
<td>110</td>
<td>85</td>
<td>58.75</td>
<td>38.75</td>
</tr>
<tr>
<td>Enhanced</td>
<td>13.75</td>
<td>63.75</td>
<td>63.75</td>
<td>61</td>
</tr>
</tbody>
</table>

Table 8.2: Performance comparison of Original and Enhanced IEEE 802.15.4 MAC for CD in term of throughput $(\Phi_{CD}(N))$ in Kbps for Low density networks.

In the enhanced MAC protocol we have used a single optimality criterion from Section 8.3. We would like to investigate the use of the other criteria also. Recent research has focused on the effect of capture effect on wireless MAC protocols. The influence of capture effect on the enhanced IEEE 802.15.4 MAC for the two data collection scenarios is another direction for future work.
8.6 Chapter Summary

We have shown that the current IEEE 802.15.4 MAC performs poorly for data collection in dense sensor networks. We presented a channel feedback enhanced MAC protocol that performs significantly better than the current version. For this we modeled the IEEE 802.15.4 MAC as a p-persistent CSMA with changing $p$, optimized a generic p-persistent CSMA MAC and used the resultant optimality criteria to propose a channel feedback-based enhancement for the original IEEE 802.15.4 MAC. Results showed that our Enhanced IEEE 802.15.4 MAC scales significantly better for both continuous data and one-shot data collection scenarios in dense networks (number of nodes is greater than 50). For low density networks the performance of the current MAC is better for upto 25 nodes after which the performance of the enhanced MAC is better.
Chapter 9

Thesis Summary

We have considered two key problems in wireless sensor networks - location support and efficient medium access for one-hop data collection.

In the first part of the thesis, we first determined operating specifications for effective location support services in consultations with engineers from Bosch Research. Then, based on these specifications we provided effective solutions to the two main components of location support services - accurate localization and fast/fair localization.

We addressed the problem of accurate localization by proposing two novel localization techniques - one based on location constraints called Ecolocation and another based on location sequences called Sequence Based Localization (SBL). In Ecolocation the unknown node’s location is identified by a set of location constraints. This location constraint set is derived from the ranks of reference nodes based on their RSS measurements of RF signals from the unknown node. The location of the unknown node is estimated by searching through grid points in the localization space and choosing the grid point that satisfies the maximum number of matched constraints as its location. If there are more than one such grid points, their centroid gives the location of the unknown node.
In Sequence-Based Localization location sequences are used to uniquely identify distinct regions in the localization space. The location of the unknown node is estimated by first determining its location sequence using RSS measurements of RF signals between the unknown node and the reference nodes. And then searching through a pre-determined list of all feasible location sequences in the localization space, called the location sequence table, to find the region represented by the “nearest” one. In this chapter, we derived expressions for the maximum number of location sequence and presented an algorithm to construct the location sequence table. We described distance metrics that measure the distance between location sequences and used them to determine the corruption in location sequences due to RF channel non-idealities. We identified an approximate indicator of the extent of location estimation error using the same distance metrics.

Through examples, we demonstrated the robustness of Ecolocation and SBL to RF channel non-idealities. We compared Ecolocation and SBL and argued that the former is equivalent to the latter for high scanning resolutions. The comparison suggested that Ecolocation is more suitable for small localization spaces and low location resolutions and that SBL is more suitable for large localization spaces and high location resolutions. Through exhaustive simulations and systematic real mote experiments, we evaluated the performance of our localization systems and presented a comparison with other state-of-the-art localization techniques for different RF channel and node deployment parameters. Results showed that sequence-based localization performs well, better than other localization techniques in both indoor and outdoor environments.

Next, we introduced the problem of fast/fair localization of mobile device in wireless sensor networks and showed that it is related to the minimum broadcast frame length problem. We investigated a greedy heuristic time scheduling algorithm for this problem using a defined set of five metrics - average localization delay, average localizable speed, localization fairness, minimum localizable speed and maximum localizable speed. We derived lower and upper bounds for the number of time slots required to schedule all reference nodes in the localization area by any
scheduling algorithm for grid and random reference node deployment distributions using simple geometric arguments.

Using simulations, we studied the dynamics of the above five metrics with respect to reference node deployment distributions, reference node densities and location estimate accuracies. Results show that the average localizable speed of mobile device decreases with increasing level of location estimate accuracy and its dependence on reference node density is minimal. The percentage of locations in the localization area that can guarantee a desired level of location estimate accuracy at a mobile device speed of 95% of the average localizable speed, the localization fairness, increases with reference node density and is independent of the accuracy level desired. The average localizable speed of the mobile device and localization fairness are better for grid deployment of reference nodes than for random deployment. Also, the localizable speed of the mobile device at which a localization area wide guarantee of a desired level of accuracy can be provided increases with reference node density and it is higher for grid deployment of reference nodes.

In the second part of the thesis, we presented our research on medium access techniques for one-hop data collection application in wireless sensor networks. We identified a spectrum of application space for the one-hop data collection problem with continuous data collection at one end and one-shot data collection at the other end. While in the continuous data collection problem the contending nodes always have a packet to transmit, in the one-shot data collection problem each contending node has a single packet to transmit.

We started out the second part of the thesis by presenting a review of the existing medium access techniques and their applicability to the spectrum of one-hop data collection applications. We argued that the traditional medium access techniques have been studied extensively for the continuous data end of the application and that ours is the first attempt at studying them for the one-shot data collection end. Thus, through this thesis, we made research contributions to the understanding of medium access techniques for one-hop data collection in the
following three directions - (a) modeling and analysis of slotted Aloha multi-access technique with binary exponential back-off for the one-shot data collection problem, (b) development of a novel location-aware medium access technique for the one-shot data collection problem, and (c) modeling, analysis, and performance evaluation of the IEEE 802.15.4 MAC protocol for both the continuous and one-shot data collection problems.

The one-shot data collection problem is characterized by the presence of a single packet in the transmission queue of each contending node. On transmission of the packet the node does not contend for the channel anymore. This leads to a non-steady state, transient behavior of the wireless networks. In this thesis we modeled the slotted Aloha medium access protocol with binary exponential back-off collision avoidance scheme using a non-ergodic Markov chain to analyze the transient nature of the one-shot data collection problem. Using this approximate model we derived flow equations to capture the network dynamics of the wireless sensor network and verified their accuracy using simulations. Using these equations, we evaluated the performance of the protocol in terms of the delay in obtaining packets from all contending nodes and the corresponding energy consumption in the wireless sensor network. Results suggest that for a given initial window size for the binary exponential back-off scheme, reducing the number of back-off stages reduces both the delay and energy consumption. According to the results, while for high node densities multiple back-off stages are preferred, for low node densities a single back-off stage performs better.

Next, we presented a novel location-aware medium access protocol for the one-shot data collection problem in wireless sensor networks. We illustrated the working of our protocol using examples and discussed its implementation aspects. The main idea in our location-aware protocol is a tree-based hierarchical partitioning of space to progressively reduce the collision domains of nodes until there are no collisions. Further, we presented results from a thorough performance evaluation of the location-aware protocol in comparison to three location-unaware
MAC protocols – HT-split, optimal p-persistent slotted CSMA, and the IEEE 802.15.4 standard MAC protocol – using simulations. We evaluated the protocol for three different location distributions of nodes – uniform-random, even-random, grid-random. Results showed that our location-aware MAC protocol in the worst case (uniform random distribution) provides as good a performance as an optimal location-unaware MAC protocol and in the best case (grid random distribution) provides 60% lower delay and energy consumption.

Finally, we modeled, analyzed, and evaluated the performance of the IEEE 802.15.4 MAC protocol for both ends of the one-hop data collection application spectrum. We have shown that the current IEEE 802.15.4 MAC performs poorly for data collection in dense sensor networks, i.e., with sensor networks with more than 50 contending nodes. We presented a channel feedback enhanced MAC protocol that performs significantly better than the current version. For this we modeled the IEEE 802.15.4 MAC as a p-persistent CSMA with changing $p$, optimized a generic p-persistent CSMA MAC, and used the resultant optimality criteria to propose a channel feedback-based enhancement for the original IEEE 802.15.4 MAC. Results showed that our Enhanced IEEE 802.15.4 MAC scales significantly better for both continuous data and one-shot data collection scenarios in dense networks. We have also shown that for low density networks the performance of the current MAC is better for upto 25 nodes beyond which the performance of the enhanced MAC is better.
Chapter 10

Future Directions

In this chapter we discuss possible directions for future work in the area of location support and efficient medium access for one-hop data collection in wireless sensor networks.

1. Location Support:

   (a) **Accurate Localization:** In the first part of this thesis we have presented localization techniques for location support in wireless sensor networks mainly for the two dimensional scenario. However, there exist many real world problems that are characterized by three dimensional operational conditions. However, many researchers have used localization techniques developed for two-dimensions in three-dimensional scenarios. It is not clear if this introduces unseen errors in the estimated location of the unknown node. We believe that a comprehensive study should be undertaken to verify if two-dimensional localization techniques are good enough in three dimensional scenarios.

   (b) **Fast & Fair Localization:** In this thesis we have formulated the problem of fast & fair localization as a graph-coloring problem and presented a heuristic based solution for the same. Based on the level of information the sensor nodes have, this solution can be either centralized or distributed. However, we believe that this problem, which occurs frequently in many wireless sensor network applications, can be formulated as
a linear program and a close to optimal solution can be derived by solving this linear program. Further, a Lagrange-Duality based formulation for the problem could reveal a sub-gradient based distributed solution for the fast & fair localization problem.

2. Medium Access for One-Hop Data Collection:

(a) **Location-Aware Medium Access for One-shot Data Collection:** In this thesis we have presented a simulation based evaluation for the tree-based hierarchical, space splitting location-aware MAC protocol for one-shot data collection problems. However, this protocol opens itself for a thorough mathematical analysis and derivation of closed form expressions for the delay and energy consumption metrics of interest. Also, this protocol is open to many possible improvements and enhancements as discussed in Section 7.5 of Chapter 7 and further research should be done to explore such possible performance enhancers for the protocol. Also, a thorough real systems-based evaluation of the protocol should be done to address the implementation concerns such as the effect of errors in the locations of nodes on the performance of the protocol. Another interesting direction for future work is the application of this protocol for 3-dimensional sensor node deployments.

(b) **Enhancement of the IEEE 802.15.4 MAC Protocol:** In Chapter 8 we have presented a channel-feedback based enhancement for the IEEE 802.15.4 MAC protocol. For this enhancement we had used a global feedback from the sink to effect the enhancement for the protocol. However, further investigations should be taken up to study the performance difference, if any, between local and global channel feedback, for the enhanced protocol. In addition a real-world practical implementation of the proposed enhancement is required to study the effect of real world phenomena such as channel errors and packet losses on the performance of the enhanced protocol.
Also, it is important to study the influence of capture effect on the enhancement of the protocol.
Reference List


Appendix A

Even-Random Distribution

Procedure to obtain even-random distribution of nodes: Below, we illustrate the steps to divide a square region into $n$ equal area partitions:

1. Divide the square into $x = \lfloor \sqrt{n} + 0.5 \rfloor$ vertical partitions.

2. Each vertical partition will be divided into a minimum of $y_{min} = \lfloor \frac{n}{x} \rfloor$ horizontal partitions.

3. Let $r = n - x \cdot y_{min}$. Determine the number of horizontal partitions $y_i$ in each vertical partition $i$ ($1 \leq i \leq x$) using,

$$y_i = \begin{cases} y_{min} + 1, & 1 \leq i \leq r \\ y_{min}, & r < i \leq x \end{cases}$$

4. The width of vertical partition $i$ is $W_i = S \cdot \frac{y_i}{n}$, where $S$ is the side length of the square.

5. Finally, divide vertical partition $i$ into $y_i$ equal parts.