

# **Towards Realizing the Internet-of-Things Vision: In-body, Homes, and Farms**

by

**Deepak Vasisht**

B.Tech., Indian Institute of Technology, Delhi (2013)

S.M., Massachusetts Institute of Technology (2015)

Submitted to the Department of Electrical Engineering and Computer Science  
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

**MASSACHUSETTS INSTITUTE OF TECHNOLOGY**

September 2019

© Massachusetts Institute of Technology 2019. All rights reserved.

Author .....  
Department of Electrical Engineering and Computer Science  
August 30, 2019

Certified by .....  
Dina Katabi  
Professor of Electrical Engineering and Computer Science  
Thesis Supervisor

Accepted by .....  
Leslie A. Kolodziejcki  
Professor of Electrical Engineering and Computer Science  
Chair, Department Committee on Graduate Students

# **Towards Realizing the Internet-of-Things Vision: In-body, Homes, and Farms**

by

Deepak Vasisht

Submitted to the Department of Electrical Engineering and Computer Science  
on August 30, 2019, in partial fulfillment of the  
requirements for the degree of  
Doctor of Philosophy

## **Abstract**

The Internet-of-things (IoT) enables us to connect our physical and digital worlds by embedding computing devices into our environment. Today, there is a huge interest in IoT systems for smart homes, smart cities, digital healthcare, data-driven agriculture, etc. However, for these IoT systems to deliver their intended vision, we need to address two important challenges: (a) operation under limited resources like power and connectivity, (b) operation in spite of extreme heterogeneity in device deployments.

In this thesis, we address both these challenges. We design a new communication primitive that allows inaccessible resource-constrained devices like in-body devices to communicate without requiring them to transmit any power of their own. To address heterogeneity, we present two approaches. First, we build a teacher-student model for IoT systems which allows us to train models that can learn to predict one sensor modality from another. This makes IoT systems more robust to failures, enables more accurate inference, and reduces deployment costs. Second, we build a formal model that embeds contextual information about the environment into the inference process and allows heterogenous devices to perform joint inference that is more accurate and robust than either of the devices alone. We demonstrate the efficacy of our approach through end-to-end systems developed for diverse environments with varying constraints on size, power, communication, and sensing modalities: inside the human body, smart homes, and agricultural farms. We deploy these systems for long-term in real world environments and present our insights from these deployments. Finally, we demonstrate that the techniques developed in this thesis have general applicability beyond the application scenarios themselves, for example, in next generation cellular communications.

Thesis Supervisor: Dina Katabi

Title: Professor of Electrical Engineering and Computer Science

## Acknowledgments

I am deeply indebted to Dina Katabi, my advisor, for shaping my identity as a researcher. Dina is very passionate about research and infinitely committed to the success of her students. She spent countless hours with me discussing research ideas, helping me with my papers, and critiquing my talks. Dina always encouraged me to aim higher, to solve more impactful problems, and to come up with more innovative solutions. She celebrated my personal and professional landmarks, and she stood by me (and fought for me) during setbacks. She has truly set a benchmark for who I'd want to be as a researcher, a mentor, and an advisor. I am very grateful that I had her as an advisor, and I hope that I can be half as good an advisor to my students.

I've been fortunate to have Ranveer Chandra, Romit Roy Choudhury, and Mohammad Alizadeh on my thesis committee. Ranveer mentored me at Microsoft Research during my two summer internships. He's been a wonderful friend, an insightful mentor, and a great collaborator. His belief in my abilities has exceeded my own, and this has made me aim higher. I've seen at close quarters how Ranveer took FarmBeats from a research idea to a system used by farmers across the world, and being a part of this journey has been inspirational and exhilarating. In the many conversations I've had with Romit, he has given me insightful technical advice about my research. More importantly, he has helped me define what it means to be a researcher: technically, ethically, and morally. I hope to have many such conversations and collaborations with Romit in the years ahead. Finally, my initiation into research happened during an undergraduate internship at Stanford, in a group where Mohammad was a Ph.D. student. Then, and now, Mohammad's research has been the benchmark for the research I'd want to do. I am so glad that Mohammad could be on my research committee and provide valuable insights that have gone into shaping this thesis.

At MIT, I worked closely with Jue Wang, Swarun Kumar, Omid Abari, Hariharan Rahul, Anubhav Jain, and Guo Zhang. I am immensely grateful to all of them. Jue held my hand into wireless research, and helped me become comfortable at MIT. She never hesitated to go an extra mile to help me, and in her, I have an ideal mentor to look up to. Swarun collaborated with me on Chronos [222] and R2-F2 [223]. His pursuit of excellence and diligence in research continues to inspire me. Working with Omid on Caraoke [9] showed me how to have fun while doing research. Omid's innovative ideas and unique insights made working with him a great learning experience. Rahul has always been a motivational presence, a source

of ideas, energy, and help. He has been willing to lend a supportive ear, even when he was swamped with work. Anubhav spent many sleepless nights with me working on Duet [220]. His hard work, zeal, and perseverance diffused into my life & made me work harder and feel happier. Finally, Guo introduced me to a new set of tools during our work on ReMix [224]. His warmth, positivity, and creativity led us to very exciting directions of research.

I also owe gratitude to the very exciting research group that I have been fortunate to be a part of: Jue, Haitham, Swarun, Lixin, Fadel, Ezz, Omid, Diego, Zach, Jouya, Chen-Yu, Mingmin, Dinesh, Anubhav, Guo, Shichao, Hao He, Hao Wang, Tianhong, Colin, Anirudh, Rumen, Yuzhe, Abbas, Lijie. Together, we went through numerous submissions, several rounds of internal feedbacks on papers and talks, and many fun group outings. Haitham, Fadel, and Ezz have been incredible mentors whose advice has shaped my Ph.D. journey. I am so grateful to Mary for ensuring that we were well-fed during deadlines, for responding to our late night emails for ordering components urgently, and in general for being the most awesome admin possible.

At Microsoft Research, I had the pleasure of working with Zerina Kapetanovic, Jongho Won, Xinxin Jin, Vasuki Narasimha Swamy, Ashish Kapoor, Sudipta Sinha, Anirudh Badam, and Gireeja Ranade on FarmBeats [221] & TYE [109]. It was an incredibly fun experience, where we spent many days on the farms trying to get “technology” to work. Ashish mentored me during my undergraduate internship at Microsoft Research, inspired me to do a Ph.D., and continued to be a source of support throughout. Sudipta introduced me to computer vision tools, that fundamentally changed our approach to FarmBeats. I also shared my time at Microsoft with Lopa, Amy, Subarna, Matt, Chris, Manikanta, Jinghao, Madhu, and Talal. All of my co-interns were amazingly dedicated, warm, and welcoming. We helped each other do good research, while never missing an opportunity to pull a prank (or three). I will cherish these memories forever.

As I complete my Ph.D. journey, I cannot help but look back and thank people who enabled me to get to MIT in the first place. I am very grateful to Manik Varma who mentored my undergraduate research thesis, helped me write my first paper, and backed me when I was a research novice. I owe deep gratitude to all the teachers who taught me, believed in me, and prepared me for the next steps in my journey.

Outside of work, I am blessed to have a fantastic set of friends. Ananth, Pranay, and Srivatsan were amazing flatmates. We grew into our graduate journeys’ together and shared each other’s sorrows and joys. I enjoyed the many dinners I had with them, and Yamini, Rajesh, and Reetik. Anuj provided sage

counsel on all things Ph.D. and has been a wise motivational presence. Many friends provided warmth and companionship even though they were located far away. I will forever cherish their friendship, warmth, support, and conversations.

Finally, and more importantly, I can't find words to describe my gratitude to my family. My parents, Anil and Saroj, taught me the values of hard work, dedication, and sacrifice. My brother, Arun, tries to bring a smile to my face, at every chance he gets. My loving wife, Shikha, has been the source of my happiness throughout my Ph.D. Every time I felt low, she would pick me up with her unconditional love, supportive words and calm presence. This thesis would not have been possible without them.

## Previously Published Material

Chapter 2 revises a previous publication [224]: Deepak Vasisht, Guo Zhang, Omid Abari, Jay Flanz, Hsiao Ming-Lu, Dina Katabi. In-body Backscatter Communication and Localization. ACM SIGCOMM, 2018.

Chapter 3 revises a previous publication [221]: Deepak Vasisht, Zerina Kapetanovic, Jongho Won, Xinxin Jin, Ranveer Chandra, Ashish Kapoor, Sudipta Sinha, Madhusudhan Sudarshan, Sean Stratman. Farmbeats: An IoT Platform for Data-Driven Agriculture. USENIX NSDI, 2017

Chapter 4 revises a previous publication [222]: Deepak Vasisht, Swarun Kumar, Dina Katabi. Decimeter-Level Localization with a Single WiFi Access Point. USENIX NSDI, 2016.

Chapter 5 revises a previous publication [220]: Deepak Vasisht, Anubhav Jain, Chen-Yu Hsu, Zachary Kabelac, Dina Katabi. Duet: Estimating User Position and Identity in Smart Homes using Intermittent and Incomplete RF-Data. ACM IMWUT/UbiComp, 2018.

Chapter 6 revises a previous publication [223]: Deepak Vasisht, Swarun Kumar, Hariharan Rahul, Dina Katabi. Eliminating Channel Feedback in Next-Generation Cellular Networks. ACM SIGCOMM, 2016.

# Contents

<b>1</b>	<b>Introduction</b>	<b>15</b>
1.1	Systems Developed	17
1.1.1	In-body Devices	17
1.1.2	Data-driven Agriculture	18
1.1.3	Context-aware Smart Homes	19
1.1.4	Beyond IoT Applications: Communication for Next Generation Cellular Networks	20
1.2	Contributions	21
1.3	Beyond this Thesis	21
1.4	Organization	22
<b>2</b>	<b>In-body Backscatter Communication and Localization</b>	<b>23</b>
2.1	Background and Related Work	26
2.2	RF Signals in Body and Implications for Backscatter	28
2.3	ReMix System Setup	31
2.4	In-Body Backscatter Communication	32
2.4.1	Understanding Surface Interference	32
2.4.2	Exploiting Non-Linear Behavior	33
2.4.3	System Design	35
2.5	In-Body Backscatter Localization	36
2.5.1	Understanding the Positioning Challenge	37
2.5.2	Principles of Positioning	38
2.6	ReMix’s Localization Algorithm	39
2.6.1	Measuring Effective In-Air Distances	39

2.6.2	Mapping Effective Distance to Actual Location . . . . .	41
2.7	Implementation . . . . .	43
2.8	Human Tissue Emulation . . . . .	44
2.9	Empirical Results . . . . .	45
2.9.1	Microbenchmarks . . . . .	45
2.9.2	Backscatter Communication . . . . .	47
2.9.3	In-Body Localization . . . . .	48
2.10	Limitations and Future Work . . . . .	50
2.11	Conclusion . . . . .	50
<b>3</b>	<b>FarmBeats: An IoT Platform for Data-driven Agriculture</b>	<b>52</b>
3.1	IoT Platform: Objectives . . . . .	55
3.2	The FarmBeats IoT Platform . . . . .	56
3.2.1	Design Decisions . . . . .	57
3.2.2	Architecture . . . . .	58
3.3	Duty Cycling the Base Station . . . . .	60
3.3.1	Duty Cycling Goals . . . . .	60
3.3.2	Power Budget . . . . .	61
3.3.3	Duty Cycling Approach . . . . .	61
3.3.4	Discussion . . . . .	63
3.4	The FarmBeats Gateway . . . . .	63
3.4.1	UAV Path Planning . . . . .	64
3.4.2	Generating Orthomosaics from UAV Videos . . . . .	65
3.4.3	Generating Precision Maps . . . . .	67
3.5	Deployment . . . . .	69
3.6	Results . . . . .	70
3.6.1	Weather Aware Base Station . . . . .	71
3.6.2	UAV Flight Planning . . . . .	72
3.6.3	Orthomosaic Generation . . . . .	73
3.6.4	Generating Precision Maps . . . . .	74



3.6.5	Other Applications . . . . .	76
3.6.6	End-to-end Deployment Statistics . . . . .	76
3.7	Related Work . . . . .	77
3.8	Conclusion & Contributions . . . . .	78
<b>4</b>	<b>Decimeter-Level Localization with a Single WiFi Access Point</b>	<b>79</b>
4.1	Overview . . . . .	83
4.2	Measuring Time of Flight . . . . .	83
4.3	Eliminating Packet Detection Delay . . . . .	86
4.4	Combating Multipath . . . . .	89
4.4.1	Computing Multipath Profiles . . . . .	89
4.4.2	Inverting the NDFT . . . . .	90
4.5	Correcting for Phase Offsets . . . . .	93
4.6	Computing Distances and Location . . . . .	95
4.7	Implementation . . . . .	97
4.8	Results . . . . .	98
4.8.1	Time-of-Flight Accuracy . . . . .	98
4.8.2	Localization Accuracy . . . . .	100
4.8.3	Impact on Network traffic . . . . .	100
4.9	Applications . . . . .	102
4.9.1	Room Occupancy Detection . . . . .	102
4.9.2	Wi-Fi Geo-Fencing . . . . .	103
4.9.3	Personal Drones . . . . .	104
4.10	Related Work . . . . .	105
4.11	Conclusion . . . . .	106
<b>5</b>	<b>Duet: Estimating User Position and Identity in Smart Homes Using Intermittent and Incomplete RF-Data</b>	<b>107</b>
5.1	Illustrative Example . . . . .	111
5.2	Duet Overview . . . . .	114
5.3	Collecting Location Observations . . . . .	115

5.3.1	Device-Free Observations	115
5.3.2	Wi-Fi Based Observations	116
5.4	Tagging Observations with Identity	117
5.5	Extending Coverage & Correcting Observations	118
5.5.1	HMM Background	119
5.5.2	Design of the HMM	119
5.6	Reasoning with Probabilistic Logic	120
5.6.1	Context Model	121
5.6.2	Updating Context	123
5.6.3	Checking State Validity	124
5.6.4	Discussion	126
5.7	Implementation	126
5.7.1	Evaluation Setup	127
5.8	Results	128
5.8.1	Overall Accuracy	128
5.8.2	Error Breakdown in Duet	129
5.8.3	Accuracy in Sub-spaces	130
5.8.4	Identity Matching	131
5.8.5	Expanding Coverage	131
5.8.6	Event Detection Accuracy	132
5.9	Related Work	133
5.10	Conclusion	134
<b>6</b>	<b>Eliminating Channel Feedback in Next-Generation Cellular Networks</b>	<b>135</b>
6.1	Related Work	138
6.2	Background	140
6.3	Intuition Underlying R2-F2	141
6.4	Algorithm	144
6.4.1	Transforming Physical Paths to Wireless Channels	144
6.4.2	From Wireless Channels to Paths	146

6.4.3	Solving the Optimization . . . . .	149
6.5	Integrating R2-F2 with the LTE Architecture . . . . .	150
6.5.1	Measuring the Uplink Channels . . . . .	151
6.5.2	Inter-cell Interference . . . . .	152
6.6	Implementation . . . . .	153
6.7	Results . . . . .	155
6.7.1	Micro Benchmarks . . . . .	155
6.7.2	Effectiveness of Beamforming . . . . .	156
6.7.3	Performance as a Function of Channel Separation . . . . .	157
6.7.4	Interference Nulling at Edge Clients . . . . .	158
6.7.5	Comparison with Angle-of-Arrival Power Profile . . . . .	159
6.8	Concluding Remarks . . . . .	161
<b>7</b>	<b>Conclusion</b>	<b>162</b>
7.1	Lessons Learnt . . . . .	163
7.2	Future Directions . . . . .	164
<b>A</b>	<b>ReMix: Lemma for Localization Algorithm</b>	<b>167</b>
<b>B</b>	<b>Duet’s Algorithm to Check State Validity</b>	<b>169</b>

# List of Figures

2-1	RF Signal Propagation through Human Body . . . . .	24
2-2	RF Signal Change in Human Body . . . . .	28
2-3	An Overview of ReMix’s System Design . . . . .	31
2-4	Radio Signal Propagation through Human Tissue Layers for In-body Localization . . . . .	36
2-5	ReMix’s Localization Model . . . . .	41
2-6	ReMix Implementation . . . . .	43
2-7	Microbenchmarks for Evaluation of ReMix . . . . .	45
2-8	Evaluation of ReMix’s Backscatter Communication Capabilities . . . . .	47
2-9	Evaluation of ReMix’s In-body Localization Accuracy . . . . .	48
2-10	Variance in In-body Localization due to Variation in Tissue Properties . . . . .	49
3-1	FarmBeats System Overview . . . . .	56
3-2	Duty Cycling Approach for FarmBeats base station . . . . .	61
3-3	FarmBeats Path Planning Algorithm for UAVs . . . . .	64
3-4	Orthomosaic Generation from UAV Videos . . . . .	66
3-5	Precision Maps Generated using FarmBeats’ Algorithm . . . . .	69
3-6	FarmBeats Usage Examples . . . . .	71
3-7	Evaluation of FarmBeats’s Power-aware Base Station Design . . . . .	72
3-8	Qualitative Evaluation of Orthomosaics . . . . .	73
3-9	Example Applications of FarmBeats . . . . .	74
4-1	Available Wi-Fi Bands . . . . .	83
4-2	Measuring Time-of-flight . . . . .	86
4-3	Chronos’s Approach to Combat Multipath . . . . .	90

4-4	Testbed for Evaluation of Chronos . . . . .	96
4-5	Chronos: Accuracy in measuring Time of Flight . . . . .	97
4-6	Variation of Chronos’s Ranging Accuracy with Distance . . . . .	100
4-7	Chronos: Localization Accuracy in Line-of-Sight and Non-Line-of-Sight . . . . .	101
4-8	Impact of Chronos on Network Traffic . . . . .	101
4-9	Chronos’s Applications . . . . .	102
4-10	Application to Personal Drones . . . . .	104
5-1	An Example of Duet’s State Tracking System . . . . .	111
5-2	Duet’s System Overview . . . . .	114
5-3	Example Deployment . . . . .	114
5-4	Duet’s Deployment in Home and Office Environments . . . . .	125
5-5	Evaluation of Duet’s Accuracy . . . . .	129
5-6	Error Analysis . . . . .	130
5-7	Distribution of Duet’s Accuracy Results Across Spatial Locations . . . . .	130
5-8	Duet’s Tagging Accuracy . . . . .	132
5-9	Expansion of Coverage . . . . .	132
5-10	Duet’s Event Detection Accuracy . . . . .	133
6-1	R2-F2’s Approach at a High Level . . . . .	136
6-2	Sample Power Profile . . . . .	137
6-3	Transforming Signal Paths to Channels on Two Frequency Bands . . . . .	142
6-4	Basestation Antenna Setup . . . . .	144
6-5	R2-F2’s System Design Overview . . . . .	151
6-6	R2-F2’s Experiment Testbeds . . . . .	153
6-7	Microbenchmark Evaluation for R2-F2 . . . . .	154
6-8	Evaluation of Beamforming using R2-F2 . . . . .	156
6-9	Nulling Interference at Edge Clients using R2-F2 . . . . .	158
6-10	Comparison of R2-F2 with AoA Power Profiles . . . . .	159

# List of Tables

2.1	Configurations for Tissue-Layer Interchange Experiment . . . . .	46
3.1	Cost Comparison of Farm Sensor Networking Solutions . . . . .	57
3.2	Classification of Agricultural Applications based on Requirements . . . . .	58
5.1	An Example of Duet’s State Tracking System . . . . .	112

# Chapter 1

## Introduction

Internet-of-things (IoT) systems extend computing into our physical world and promise to transform multiple spheres of human lives: medicine, healthcare, homes, industries, and cities. Over the next decade, the IoT vision will encompass networks of micro-robots that can move inside human bodies to deliver targeted medicine, help regulate cancer treatments, and even perform in-body biopsies. IoT devices will enable smart homes that organically adapt to their occupants' needs, while also monitoring their health and wellbeing. Arrays of cheap sensors spread around the city will allow us to monitor and regulate traffic in real-time, reduce emissions, and aid navigation of autonomous vehicles. This exciting promise of IoT has led to large commercial investments in both home and industrial setups, and these investments are expected to exceed 1 trillion US dollars by 2022 [105].

However, realizing this vision requires us to solve two key challenges that limit IoT systems today:

- (i) **Resource-Starved Environments:** Today, IoT devices are designed for environments that have easy access to resources like power, and connectivity. For instance, in-homes devices like smart thermostats, smart light bulbs, etc. can be plugged into the power line, and can be connected to Wi-Fi for internet. This is a far cry from the IoT vision, where IoT systems can be deployed in the wild: in remote rural areas, in farms and forests, under water, in space, and even inside human bodies. For all of these environments, resources like power and connectivity are scarce. For example, in-body devices are bound to be small in size, cannot carry large batteries, and as a result, are very power starved. Similarly, agricultural farms are located in sparsely populated areas that do not have traditional sources of connectivity like cellular or Wi-Fi. How does one design IoT systems in such resource-starved environments?

(ii) **Extreme Heterogeneity:** IoT deployments are fundamentally heterogenous. A single deployment can consist of devices ranging from tiny temperature sensors that generate few bytes of data to cameras that stream Gigabits of videos. These devices sense diverse physical parameters, have very different networking and power requirements, and vary vastly in their interface with the end users. As a result of this heterogeneity, these devices are frequently treated as independent from each other or are (superficially) connected through a hub. This approach to IoT system design is very inefficient and fails to leverage the benefits that arise from optimizing the system as a whole. For example, in smart homes, some IoT devices (like a smart watch) learn information about a user, while others (like smart thermostat, smart fridge) learn about actions performed in the environment. By combining information across devices, we can learn a richer set of actions while tagging those actions with the identity of the people performing the actions. More generally, collaboration between different devices in an IoT deployment can exploit complementarity in data, coverage, and failure modes to increase robustness and deliver better inferences about the physical environment. How can we design IoT systems that allow the different devices to efficiently collaborate and co-operate for higher level sensing tasks?

In this thesis, we design new primitives that address these challenges. To address resource scarcity, we design a new communication primitive for inaccessible resource-constrained devices like in-body devices. Communication is a fundamental component of IoT devices, but it accounts for majority of the power budget. In fact, for resource constrained devices like in-body devices, communication contributes to about half of the power budget[32, 42]. We design a new communication primitive that allows these devices to communicate without requiring them to transmit any power of their own. Our system relies purely on radio signals reflected by these devices and hence minimizes the power cost of communicating data.

We develop two approaches to address heterogeneity in IoT system design. First, we develop a teacher-student model for IoT systems, where we train inference models that can use physical data sensed by one sensing modality to predict physical data in other modalities. This ability fundamentally changes the design of IoT systems. Specifically, it allows IoT systems to be more robust to failures by allowing other sensors to fill in for failed sensors. It enables systems to increase the spatiotemporal range of sensing modalities when one of the modalities is limited in space or time. Finally, it significantly reduces deployment costs and overheads by removing redundant sensors.

Second, we build a new approach based on probabilistic first order logic that allows a system to reason about and relate information from multiple sensors. We also embed contextual information about the



environment (e.g., the floor map in smart homes) into our formulation to further strengthen the reasoning. Combined with the constraints imposed by the contextual information, our approach allows heterogeneous sensors to make joint inferences about the environment that are more accurate and more robust to errors than either of the sensors alone. As a result, our formulation can ingest multiple error-prone sensor data streams and output accurate inferences about the environment.

In this dissertation, we demonstrate the efficacy of these approaches by incorporating them into end-to-end systems designed for a diverse range of environments that vary in their size, scale, communication, and power requirements: in-body, agricultural farms, and smart homes. In building these systems, we develop novel algorithms and techniques based on unique insights from all layers of the computing stack: hardware, communication, network, inference, and application. We deploy these systems for long-term in real world environments and present our deployment experiences and insights. Finally, we demonstrate that the techniques developed in this thesis have general applicability beyond the application scenarios themselves.

## **1.1 Systems Developed**

### **1.1.1 In-body Devices**

In-body devices promise to transform modern medicine. Small swallowable capsules can replace 40-inch long tubes that had to be thrust through a person’s mouth, throat, and esophagus for endoscopy. Small tumor markers can enable doctors to track tumors and avoid radiation damage to healthy tissues. Micro-robots can perform targeted drug delivery, in-body biopsies, and targeted nerve stimulation.

All of these devices need to communicate the data that they are sensing and we need to identify their location in-body. For instance, endoscopy capsules need to send the images from the G.I. tract and tag the images with the location so that the doctors can isolate the part of the G.I. tract that has abnormalities. However, these tiny devices are constrained by the lack of power sources in-body and cannot carry large batteries to support traditional communication technologies. To solve this problem, we built ReMix, the first system that can communicate with and localize in-body devices without requiring them to transmit any power of their own. ReMix uses a device outside the body to transmit radio signals. The in-body devices communicate by just reflecting the signal and hence, do not have to generate a signal of their

own. Crucially, ReMix can also use these reflected signals to find the location of the device inside the body (a.k.a. in-body GPS). This allows tracking of in-body devices such as tumor markers using safe, inexpensive radio waves; a capability that has so far relied on expensive and harmful technologies like X-rays & CT.

We evaluated the performance of ReMix in human tissue substitutes – chemically synthesized human tissues, and dead animal tissues like whole chicken and pork belly. ReMix can locate devices in-body to centimeter-level accuracy while communicating with them at data rates up to 1 Megabits per second.

### **1.1.2 Data-driven Agriculture**

The global food production needs to double by 2050 to meet the growing demands of the increasing population. Unfortunately, the production is expected to go down, given the challenges of reduced cultivable land, lower water tables, and climate change. Data-driven agriculture is the most promising approach to alleviate this challenge. Data-driven techniques rely on dense measurements of farm’s physical characteristics and can enable farmers to reduce input costs, enhance production, and ensure environmental sustainability. Yet, such techniques are limited to few farmers in the developed countries (and almost none in the developing world). This is because of three reasons: (a) dense deployment of sensors is expensive and inhibits the natural workflow of a farmer who needs to drive tractors and bullock carts through this complex maze of sensors; (b) farm sensors are susceptible to harsh weather conditions, making data collection intermittent; (c) farms are located in rural areas with sparse connectivity (cellular or otherwise), forcing labor-extensive manual data collection.

We built FarmBeats, an end-to-end IoT platform for data-driven agriculture. FarmBeats includes system designs and algorithms that automate data collection, ensure availability in harsh weather conditions, and estimate dense sensor maps of the farm from sparse sensor deployments. The key component of FarmBeats is a new multi-modal sensor fusion algorithm that combines sensor data from sparse on-ground sensors with visual images collected using a drone (an increasingly common farm tool) to create a dense sensor map of the farm. Our algorithm operates on the teacher-student model. Specifically, we use spatially sparse sensor data collected from on-ground sensors to train a graphical model that converts visual data from drones to sensor data (like temperature, moisture, pH, etc). This model, once trained, can be used to infer spatially dense sensor values, even at locations that do not have any ground sensors. This

allows FarmBeats to operate with an order of magnitude fewer sensors.

We deployed FarmBeats in one small and one large farm in the United States over a period of six months. FarmBeats collected over 10 million sensor measurements, 1 million farm images, and hundreds of aerial surveys. In our deployments, the multi-modal sensor-fusion has reduced the need of ground sensors by one to two orders of magnitude. FarmBeats has been resilient to long stretches of cloudy days that cripple availability of solar power on the farms and thunderstorms that break down the backbone Internet connection for days.

### **1.1.3 Context-aware Smart Homes**

For a long time, we have been captivated by the vision of smart homes that organically adapt themselves to their occupants' needs. Smart homes can control heating and lighting, alert against intruders, and monitor occupants' daily wellbeing. However, current devices hardly exhibit such smart behavior and need to be controlled by explicit user input either through an app or a voice-based assistant like Google Home.

RF-based indoor localization can play a central role in realizing the vision of smart homes, by enabling the home to be aware of its occupants' locations and react to their presence. It can allow a smart home to track the user as she sits on the TV couch and tune to her favorite channel, alert the parents if, during their absence, the babysitter enters their bedroom instead of the baby's room, and monitor mobility of occupants to check for sign of depression or diseases. However, traditional RF-based localization systems are targeted to enterprise environments, and cannot directly adapt to smart homes. Specifically, they require measurements from multiple vantage points, while homes typically have a single access point. Moreover, such systems rely on users carrying their smartphones all the time, which is a false assumption in homes.

We developed a series of systems (Chronos, and Duet) that enable location-aware smart homes. Chronos is the first system that can enable accurate decimeter-level localization with a single Wi-Fi access point. It allows the single Wi-Fi access point in homes to track user devices. Then, we built Duet, a multi-modal positioning framework for smart homes. Duet ingests two modes of positioning data: location of user devices in the home and identity-free locations of humans obtained by relying on radio reflections. Both these data streams are partially correlated because people intermittently interact with their devices. Furthermore, these data streams are prone to errors because of blockage caused by TV screens, bathroom

tiles, concrete pillars, and other metallic objects. Duet builds a first-order logic based reasoning framework that incorporates constraints of the physical world like the inability of people to walk through walls, or the inability to be present at multiple places simultaneously. By incorporating these constraints, Duet’s formulation can ingest intermittent, error-prone location data streams and predict the actual location and identity of users in a smart home. We deployed Duet in homes and small offices for several weeks with users using their own devices and requiring no change to their behavior. Across these deployments, Duet estimated the correct semantic location (e.g., bed, TV couch, kitchen) of the user over 95% of the times.

### **1.1.4 Beyond IoT Applications: Communication for Next Generation Cellular Networks**

The techniques developed in this thesis have applications beyond IoT. Specifically, Chronos is built on a new primitive that allows accurate distance measurements between off-the-shelf devices. We showed that we can build on this primitive to reduce overhead for communications in next generation cellular networks. In particular, use of MIMO techniques in cellular networks requires client devices to measure the wireless channel from the basestation to the client and send it back to the basestation as feedback. This feedback overhead scales linearly with the number of antennas on the base station. For resource constrained user devices, this overhead can consume significant power resources. Furthermore, as the number of devices scales up, this uplink feedback wastes increasingly large fraction of the uplink bandwidth. Since 5G networks plan to rely on massive antenna arrays on the base stations and gear for large scale IoT deployments, this overhead has become a core problem; so much so that the LTE standardization body has recognized this as a challenge for future LTE networks.

We built R2-F2, a system that can eliminate this feedback requirement, by inferring these wireless channels from natural client transmissions. The key challenge in building R2-F2 is that a majority of cellular networks (all networks in United States) use different frequencies for uplink transmissions from the client and downlink transmissions from the basestation. Thus, R2-F2 must use transmissions on one frequency (uplink) to determine wireless channels on a different frequency (downlink). R2-F2 makes the observation that while the transmission happens on different frequencies, the underlying paths that the signal travels are the same. So, we identified a frequency-invariant representation of the physical paths and designed an optimization problem that could convert wireless channel measurements to physical paths

and vice versa. Solving this optimization problem allows the basestation to convert channel measurements on uplink frequency to underlying physical paths, and then use these physical paths to infer the wireless channels on downlink frequency; without relying on user feedback.

## 1.2 Contributions

In building these systems, this thesis makes the following core contributions:

- ReMix is the first system that can enable communication (far-field) and localization of deep-tissue in-body devices without requiring the in-body devices to transmit any power of their own.
- FarmBeats built a novel primitive that can combine data from multiple sensing modalities (specifically visual data and ground sensor data) to enable a novel IoT platform design for data-driven agriculture.
- Chronos presents the first technique to enable decimeter level positioning using a single Wi-Fi access point, thereby opening up an opportunity to extend Wi-Fi positioning to homes and small businesses that have just one access point.
- Duet built a novel logical reasoning framework that exploits the synergy between verification techniques and indoor positioning to deliver accurate semantic localization in smart homes; even though the underlying data might be intermittent and error-prone.
- R2-F2 is the first system that demonstrates the practicality of inferring LTE downlink channels from uplink channels using reciprocity and without channel feedback. This result contributes a better understanding of reciprocity in FDD systems, and a solution to one of the important challenges facing future 5G MIMO networks.

## 1.3 Beyond this Thesis

The techniques present in this thesis have wider applicability beyond the scope of systems developed here.

We list a few examples below:

- ReMix’s design to communicate with in-body devices without requiring them to transmit any power can be extended to inaccessible sensors in other domains, like sensors deployed inside the ground

for soil monitoring, or sensors deployed inside bridges and buildings for monitoring the construction strength.

- Chronos’s ability to measure distance between two devices has potential applications in robotics, where swarm formations need to maintain fixed distance within themselves.
- Duet’s first-order logic formulation shows how constraints of the environment can be embedded in the inference process, and hence, can be extended to inferences not just in smart homes, but also in smart cities, farms, and other environments.
- FarmBeats’s approach of training models to predict ground sensor data from visual imagery collected by drones has wider significance given the prevalence of satellite imagery (visible bands along with infra-red images) available for farms and forests. In the future, the algorithm can be updated to incorporate such satellite imagery, when available.
- R2-F2’s algorithm to convert wireless channels from one frequency band to another band is a fundamental capability for wireless communications and has future applications in the domains of cellular networks, mmWave communications, physical layer security, etc.

## 1.4 Organization

The rest of the thesis delves deeper into the algorithms, techniques, and implementation details underlying the systems described above. Chapter 2 describes ReMix and how it enables devices inside the body to communicate and be localized just using signal reflections. Chapter 3 discusses the techniques behind FarmBeats. Chapter 4 presents a new primitive for indoor positioning, i.e., the ability to measure distance between two devices using off-the-shelf Wi-Fi cards. It includes the system design for Chronos, an indoor positioning system built on top of this primitive. In Chapter 5, we describe Duet, a logical reasoning framework for estimating the semantic location of occupants in a smart home, even though the occupants may not carry a device on them all the time. Finally, we demonstrate the efficacy of techniques built in the thesis for cellular communication in Chapter 6. We conclude with a discussion of the lessons learnt and a list of possible future directions in chapter 7.

## Chapter 2

# In-body Backscatter Communication and Localization

Backscatter is one of the most power-efficient communication technologies [143, 96]. It allows a device to communicate at zero power budget, simply by reflecting an ongoing RF signal. The reflected signal may also be used to localize the backscatter device. Because of these properties, there is much interest in using backscatter to localize and communicate with micro-implants in the human body [92, 257, 256, 154]. In particular, we have recently witnessed the introduction of implantable RFID microchips, the size of a grain of rice, which can be injected in the body using a needle. Such RFIDs have been deployed in animals and humans for identification, tracking, sensing, and even replacing one's own credit card [154, 152, 35].

Today, RFID implants are inserted directly under the skin, within a few mm from the body surface. But, if backscatter could be made to work much deeper in the body (e.g., 3-10cm), it will enable multiple critical applications. In particular, swallowable capsule endoscopes (e.g., PillCam) have made examining the gastrointestinal (GI) tract much easier by eliminating the pain associated with inserting conventional scopes. Backscatter can address two key challenges with existing capsules. Today half the capsule's energy is spent on RF transmission [253]. Backscatter can significantly reduce the power budget of these capsules, allowing them to function for a longer time [11]. Further, if the backscatter signal can be localized accurately (to within a couple of cm), it would enable the capsule to adapt its functions depending on its location in the GI tract. For example, it can deposit drugs in certain areas, or adapt video frame rate to obtain higher resolution at critical areas [178, 218]. Other potential applications of deep-tissue backscatter involve tracking micro-robots in blood stream [257, 256], and localizing fiducial markers to

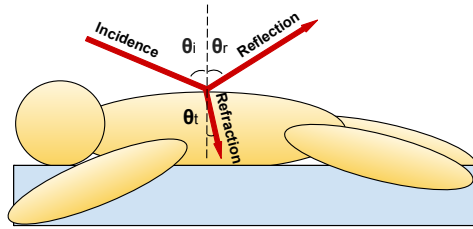


Figure 2-1: A large portion of RF signal is reflected off the human body. The signal that crosses the interface changes direction.

detect movements of breast, liver or lung tumors during radiation therapy [125, 100].

However, deep-tissue backscatter is challenging because:

- *Surface Interference:* Due to the large difference in electrical properties between the human body and air, a large portion of the incident RF signal is reflected off the human surface, as shown in Fig. 2-1. Furthermore, RF signals experience exponentially more attenuation in human tissues (e.g., muscles) than in air. Thus, if the implant is few centimeters below the surface, the RF signal loses about 40 dB each way compared to reflections off the skin. As a result, the backscatter signal is 80 dB (100 million times) weaker than the signal reflected off the body surface. This causes a dynamic range problem at the receiver –i.e., the receiver ADC will be saturated by the larger reflection from the skin level and unable to detect the backscatter signal.
- *Signal Deflection:* RF signals propagate 8 times slower in muscles than in air. This difference breaks past algorithms for RF-based localization. A key principle in localization is that signals propagate along straight lines, and hence (in the absence of multipath), the direction of the signal is the same as the direction of the source. Yet, the difference in speed between air and muscle causes the signal to significantly deflect ( Fig. 2-1) as it enters the body.

This chapter introduces ReMix, a novel wireless system for deep-tissue backscatter. A key property of ReMix is its ability to address the above challenges –i.e., it can capture the backscatter signal and localize a deep-tissue device despite surface interference and signal deflection.

So, how does one isolate the signal from a backscatter implant, when skin reflections are 8 orders of magnitude stronger than the backscattered signal. Our insight is to use non-ideal behavior in circuits to separate the backscatter reflections from all other reflections in the environment. Specifically, we rely on the concept of non-linearity. Standard circuits are designed to have linear behavior in their operating range,



i.e., if the input receives signal at two different frequencies  $f_1$  and  $f_2$ , the output transmits frequencies  $f_1$  and  $f_2$ . In contrast, non-linear behavior (which is non-ideal) mixes the input frequencies, i.e, given input signal at frequencies  $f_1$  and  $f_2$ , the output signal contains the original frequencies  $f_1$  and  $f_2$ , along with linear combinations such as  $f_1 + f_2$ ,  $f_1 - f_2$ ,  $2f_1 - f_2$ , etc. In traditional RF design, these combinations are unwanted and designers suppress them as much as possible. In contrast to the traditional approach, our insight is that we can use this unwanted behavior to our advantage and isolate the backscatter reflections.

Specifically, instead of using a standard backscatter circuit on the implant, we use a circuit that promotes non-linearities. In this case, the backscatter circuit mixes the input frequencies and reflects both the original frequencies as well as the various mixes. We can then tune our receiver to listen to the mixed frequencies, e.g.,  $f_1 + f_2$ ,  $2f_1 - f_2$ , etc. Since skin reflections are at  $f_1$  and  $f_2$ , they can be filtered out. Thus, the backscatter signals at mixed frequencies like  $f_1 + f_2$  and  $2f_1 - f_2$  are interference free. In Sec. 2.4, we explain how we implement such non-linearity while keeping the RFID circuit passive (i.e., no power source).

Second, we need to localize the backscatter implant even though the signal travels along crooked paths. To address this problem, we introduce a new model for RF-based localization where the path between two points is modeled with linear splines (piecewise segments) as opposed to straight lines. The length of each segment refers to the stretch of the path in a particular material (air, fat, muscle). We analyze the model given the electrical properties of human tissues, which reveals interesting properties of in-body propagation such as: (a) one can ignore in-body multi-path without affecting accuracy; and (b) while the implant antenna radiates signal in the body in all directions, due to the difference in muscle electrical properties, all signals that leave the body exit through a small area on the surface. By leveraging such in-body propagation properties, we design an accurate localization system that accounts for the fact that the signal suffers deflection and has a different propagation speed (i.e. different wavelength) in different body tissues. Our model also accounts for the signal changing frequency inside the body –i.e. it enters the body as  $f_1$  and  $f_2$ , gets mixed at the implant, and continues as  $f_1 + f_2$ ,  $2f_1 - f_2$ , etc.

We have implemented ReMix and evaluated its performance using both animal tissues (chicken and pork) and human tissue-phantoms, which are designed to emulate human tissue electrical properties. This is in-line with evaluations used for in-body applications like ultrasound imaging [49], in-body wireless power transfer [11], tomographic imaging [160, 192], etc. Our results can be summarized as follows:

- ReMix delivers effective backscatter communication. It achieves an SNR between 11.5 dB and 17

dB (for 1 MHz bandwidth) for a backscatter device that is 1 to 8 cm deep in animal tissues. This performance is due to ReMix's non-linear mixing of the RF signals which allows it to deliver good SNRs despite strong surface interference.

- ReMix delivers accurate backscatter localization. Its algorithm can localize a deep-tissue implant to an average accuracy of 1.4 cm. In contrast, directly applying standard localization algorithms results in an average error of 7.5 cm due to inability to deal with signal deflection.

**Contribution:** ReMix makes the following contributions:

- It is the first to demonstrate deep-tissue backscatter communication and localization in phantom and animal tissues.
- It is the first system that proposes the use of non-linearities to deal with interference from unwanted skin reflections that can mask the desired signal from the in-body implant.
- It is the first to present a time-of-flight localization algorithm that accounts for signal deflection in body tissues.

## 2.1 Background and Related Work

The medical industry is looking at a wide array of in-body devices that include pacemakers that communicate their data over the wireless channel, smart pills that image the gastrointestinal tract, and microscale robots that access organs through the bloodstream. Today, such deep tissue systems communicate by generating their own radio signal, a process that consumes a lot of energy. For instance, in wireless capsule endoscopes, RF consumes 4 to 10 times more power than the sensors [253]. As a result, these capsules use large batteries that occupy about 40-50% of the space of the capsule [32, 42]. Reducing the power requirement for RF transmissions can reduce the size of the capsules making them more easy to swallow. It can also improve completion likelihood. Past work has found that 16.5% of the times, capsule endoscopes fail to completely visualize the small bowel primarily due to limited battery life [173].

Similarly, interest in deep tissue localization is on the rise. Localization of deep tissue sensors like capsule endoscope can enable physicians to isolate the parts of the GI tract with abnormalities, adapt video

frame rates based on location, and deposit biomarkers at specific locations [178, 218]. The localization requirements for such capsules are on the order of a few centimeters [178].

Past work has tried to tackle these problems along multiple axes. Researchers have considered wireless power transfer –i.e., charging an implant using RF signals [148, 11]. These systems typically operate in the midfield where the RF transmitter is either directly in touch with the body or within a few centimeters from it. Our work is complementary to this literature. First, we address localization which is not addressed by power transfer. Second, even if the implant has the ability to harvest power, it can still leverage backscatter to communicate at zero power and save its harvested energy for its sensing tasks.

The literature also has few proposals for in-body localization. One line of research uses magnetic field analysis [178, 22, 95]. The advantage of using the magnetic field is that its properties do not change much between air and human tissues. The disadvantage however is that the magnetic field power decays with a factor  $d^6$  as it travels through air [48]. Hence, the magnetic receiver (the receiving coil) has to be in touch with the body surface or within a few centimeters. Further, magnetic implants can be problematic. They can be painful if the person is exposed to a strong magnetic field as in the case of MRI [117]. They can also affect MRI images making it difficult to detect a tumor in the area near the implant [117]. As a result, this form of localization is not widely used. Doctors also use X-ray or sonar for localization. These methods are expensive. Further, continuous tracking of an implant requires excessive x-ray exposure which increases cancer risk [52]. Finally, the use of ultrasound for in-body localization requires direct contact with the human skin, making it infeasible for several medical applications. For instance, presence of metallic equipment close to the human body can be a hindrance for administering X-ray/proton beams used for radiation therapy in cancer treatments [167, 56].

Many implants need RF signals to communicate their data. It would be highly beneficial if the same signal can also be used for localization. Past proposals for in-body RF-based localization use the received signal strength (RSS) [250, 233]. Those systems use an array of receive antennas and either assume the implant to be closest to the receive antenna with the highest power or use path loss models to estimate location [218]. Analysis of the error bounds on RSS in-body localization has reported lower bounds of 4 to 6 cm [250] even when using up to 50 receive antennas. Past work has also tried to adapt indoor localization based on time-of-flight (ToF) or angle of arrival (AoA) for the domain [217, 149, 41]. Unfortunately, these systems are based purely on simulation, lack any empirical results, and most of them ignore signal deflection.

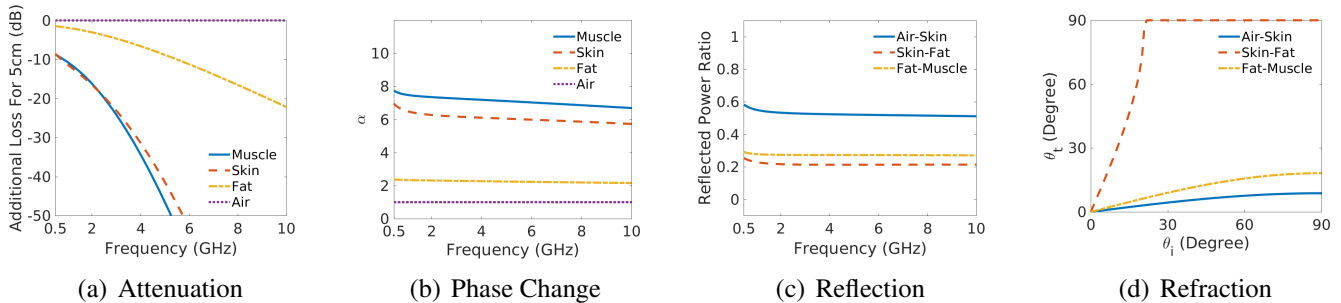


Figure 2-2: **Signal Change in Human Body:** As RF signals traverse human body, they suffer (a) additional attenuation, (b) increased phase change, and (c) reflection and (d) refraction at the interfaces.

There is also a rich literature about backscatter communication and localization in-air [34, 258, 96, 143]. We build on this foundation but focus on intrinsically different problems that stem from RF propagation in deep tissues, such as signal deflection and body surface interference. We note that the design in [258, 96, 120] proposes shifting the frequency of the backscatter signal to avoid Wi-Fi interference. While ReMix also shifts the backscatter frequency, it differs in both goal and technique. ReMix’s aim is to avoid interference from the skin surface, and its solution is completely passive and does not require any active components on the implant.

Finally, past work on harmonic RADARs and RFID-based localization ([185, 46, 82, 83]) has used non-linearity to mix two frequencies and weed out unwanted reflections from the environment. However, ReMix is the first system to use this technique to separate skin reflections from the signal from in-body implants. Furthermore, as described before, ReMix solves unique localization challenges (like signal deflections, change of wavelength in-body) that do not exist in *in-air* localization.

## 2.2 RF Signals in Body and Implications for Backscatter

We start with an analysis of how RF signals behave as they propagate in biomaterial (e.g., fat, muscles), and the implications for in-body backscatter. From the perspective of electromagnetic (EM) waves, each material is characterized by two parameters: relative electrical permittivity,  $\epsilon_r$  and relative magnetic permeability,  $\mu_r$ . These are *complex numbers* that capture how the electrical and magnetic fields in an EM wave interact with the material. Both  $\epsilon_r$  and  $\mu_r$  are 1 for air and vacuum. For biological tissues, the relative magnetic permeability  $\mu_r$  can be approximated as 1 [123], so we set  $\mu_r = 1$  for the rest of the chapter. However,  $\epsilon_r$  has high variability depending on the tissue type and frequency of transmission. For example,

for frequencies around 1GHz (commonly used by in-body implants), the value of  $\epsilon_r$  in muscle is  $55 - 18j$  [103].

The value of  $\epsilon_r$  is very important because it changes the speed of light and other electromagnetic waves (EM) in a material. Specifically, the speed of light in a biomaterial (e.g., muscle, fat, skin) is given by:  $v = \frac{c}{\sqrt{\epsilon_r}}$ , where  $c$  is the speed of light in vacuum and (to a good approximation) air. The change in the speed of the EM wave has important implications.

**(a) Attenuation Increases Exponentially:** Recall that, for a signal at frequency  $f$ , traveling in free space from a transmitter to a receiver separated by distance  $d$ , the wireless channel  $h(f, d)$  is given by

$$h(f, d) = \frac{A}{d} e^{-j2\pi f \frac{d}{c}} \quad (2.1)$$

where  $A$  is the attenuation constant that depends on the antenna beam patterns and  $c$  is the speed of light in vacuum.

For biomaterial, incorporating EM wave speed change in equation (2.1) gives us the wireless channel,  $h_M(f, d)$ :

$$h_M(f, d) = \frac{A}{d} e^{-j2\pi f \frac{d\sqrt{\epsilon_r}}{c}} \quad (2.2)$$

To understand the impact of  $\epsilon_r$  on wave propagation, let us write  $\sqrt{\epsilon_r} = \alpha - \beta j$ , where  $\alpha$  and  $\beta$  are positive real numbers. The channel equation can then be updated as:

$$h_M(f, d) = \frac{A}{d} e^{-j2\pi f \frac{d(\alpha - \beta j)}{c}} = \frac{A}{d} e^{-j2\pi f \frac{d\alpha}{c}} e^{-2\pi f \frac{d\beta}{c}} \quad (2.3)$$

Note that the term  $e^{-2\pi f \frac{d\beta}{c}}$  causes exponential loss in magnitude of the signal during propagation. The higher the value of  $\beta$ , the higher the loss. This is in addition to the propagation attenuation experienced by the signal in free space, given by  $\frac{A}{d}$ . Fig. 2-2(a) plots the additional loss observed by an EM wave traveling for 5cm in different biomaterials, i.e. muscle, fat and skin. As can be seen from the figure, muscle tissues and skin tissues are similar to each other but are very different from fat, which is closer to air. Also, muscle tissues experience significant additional loss in comparison with in-air signals.

There are two take home messages from Fig. 2-2(a). First, in-body RF signals should use relatively low frequencies to avoid the drastic power loss occurring at higher frequencies. In fact, it is a common practice to use frequencies about 1GHz, which are small enough to have a relatively low loss, but also large enough to enable relatively small electronics and antennas [148, 11]. Second, for backscatter signals

which have to traverse the body twice, they lose more than 20dB just to get 5cm deep. This amount is important as we start accumulating the various losses that eventually lead to about 80dB of surface to backscatter interference (discussed in detail in Sec. 2.4.1).

**(b) Antennas become less efficient:** The electrical permittivity of a material further affects the efficiency of in-body antennas. As an antenna is placed in-body, its radiation efficiency decreases and its inherent losses increase as a function of  $\epsilon_r$ . For muscle tissues, these effects incur another 10 – 20 dB of loss depending on the antenna design [122].

**(c) Wavelength Shrinks:** Consider again Eq. 2.3. Note that the signal phase changes much faster in biomaterial than in air. Specifically, the phase changes  $\alpha$  times faster in biomaterial than in air. This is because the wavelength is  $\alpha$  times smaller. Fig. 2-2(b) plots  $\alpha$  for different materials inside the human tissues, i.e. muscle, fat and skin. It shows that the phase changes 8 times faster in muscle than air. This property is useful for RF-based localization algorithms that leverage phase changes to measure distance because it increases sensitivity and allows for measuring smaller distances (for the same signal SNR).

**(d) Signal Reflection:** The value of  $\epsilon_r$  affects not only how the signal travels through a material, but also affects what happens at the interface between two materials. Consider a signal traveling from a material with relative permittivity  $\epsilon_{r1}$  to a material with relative permittivity  $\epsilon_{r2}$ . Further, for ease of exposition, assume that the signal is traveling perpendicular to the interface, which is the direction of minimum reflection (the more general case is discussed in [172].) In this case, the ratio of the reflected power  $P_r$  and incident power  $P_t$  is given by:

$$\frac{P_r}{P_t} = \left| \frac{\sqrt{\epsilon_{r1}} - \sqrt{\epsilon_{r2}}}{\sqrt{\epsilon_{r1}} + \sqrt{\epsilon_{r2}}} \right|^2 \quad (2.4)$$

As can be seen in the equation, larger the difference between the properties of two materials, the more signal power is reflected. Fig. 2-2(c) plots the ratio of power reflected for different interfaces as a function of frequency. Since going in-body requires the signal to traverse air-skin, skin-fat and fat-muscle interfaces (twice for backscatter), a large portion of the power is reflected back before it reaches the implant.

**(e) Signal Refraction:** When an RF signal traverses the interface between two materials, it experiences a change in direction, as shown in Fig. 2-1. This bending in the signal is called refraction. The relationship of the angle of incidence ( $\theta_i$ ) and the angle of refraction ( $\theta_t$ ) can be approximated by the following equation

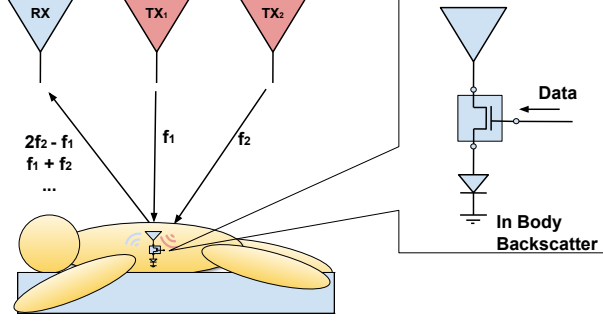


Figure 2-3: An overview of ReMix’s system setup and a schematic of the backscatter device (inlet).

(for exact equation, see [172]):

$$Re(\sqrt{\epsilon_{r1}}) \sin \theta_i = Re(\sqrt{\epsilon_{r2}}) \sin \theta_t \quad (2.5)$$

where  $Re(\cdot)$  denotes the real part of a complex number. We plot the angle of refraction for different angles of incidence for various interfaces in Fig. 2-2(d). The figure shows that RF signals experience significant bending at skin-fat, fat-muscle and air-skin interfaces. This is the key challenge for localization of a device implanted inside the body.

We make an important observation related to the air-skin interface, i.e., the blue line in Fig. 2-2(d). The figure clearly shows that regardless of the incident angle, the refraction angle is always near zero. This means that it does not matter how the signal arrives in air, it enters the body almost along the direction of the normal on the surface (Fig. 2-1). Since EM wave paths are reversible, this result means that all RF waves that exit the body must arrive at the skin-air interface almost perpendicular to the body surface. RF waves that try to exit the body with an angle far from the normal are not allowed to do so, and are reflected back inside the body. We will revisit this observation later as we talk about RF-based in body localization.

Finally, we end by reminding the reader that properties (a), (b) and (d) above all combine to reduce the power of the backscatter signal and contribute to the surface interference problem. Properties (c) and (e) on the other change the signal’s path and wavelength and hence affect localization.

### 2.3 ReMix System Setup

ReMix is a backscatter design for deep-tissue implants or sensors (e.g., smart capsule). ReMix has two main components: a transceiver system comprising of multiple antennas and an in-body backscatter mod-

ule (i.e., RFID). The transceiver system has two transmit antennas, one for each frequency being transmitted and multiple receive antennas.

This setup can be placed below or above a person's bed or on the side. The transceiver does not need to touch the body. It works in the far field and can be anywhere between 0.5 meters to a few meters. The in-body module is a small unit that can be attached to standard in-body devices that need to communicate data or be localized. An example arrangement is presented in Fig. 2-3. While it is common to perform medical procedures while the patient is lying on a bed, the operation of ReMix does not require the patient to be in a particular position.

## 2.4 In-Body Backscatter Communication

We discuss the challenges that face in-body backscatter communication and how ReMix overcomes them.

### 2.4.1 Understanding Surface Interference

As explained earlier, the signal reflected off the surface of the human body is much more than the small backscatter signal from a deep tissue device. To understand the power ratio between these two signals, let us consider a transceiver placed outside the human body and compare the reflected power observed from human body with the reflected power from an antenna in deep tissue (5cm below the skin), connected to a perfect backscatter system (i.e., no loss in the backscatter circuit).

There are four major sources of power differences between skin reflections and reflections from the backscatter device. First, as discussed in Sec. 2.2, a portion of the incident power is reflected at the interface between air and body, and thus never ends up reaching the backscatter system. Second, since the human body is a lossy medium, the signal faces large attenuation (exponential in distance) during in-body propagation (see Eq. 2.3). Third, the antenna efficiency in-body is reduced by 10-20 dB [122]. Finally, the effective area of radiation of an in-body antenna is much smaller than the skin area.

We do a back of the envelope calculation to combine these factors and compare the power reflected off the skin with the power reflected by the backscatter device. The attenuation, reflection and antenna efficiency loss from Sec. 2.2 produce a combined loss of at least 30 dB in one direction. Since backscatter is two way, the total loss is at least 60 dB. When we include the effect of the effective area of the skin



which is much larger than an in-body antenna (typically the size of grain of rice [72]), the total loss is  $\approx 80$  dB. Similar loss has been empirically observed by several experiments in past work [11, 130].

Thus, the signal reflection measured from the backscatter system is at least 80 dB lower than the signal measured from the surface. Such a huge difference in power will overwhelm the receiver's ADC and prevent it from capturing the backscatter signal. The problem is exacerbated by the fact that the human body is not static. Breathing, pulsing, and bowel movements cause the skin to move and vibrate. As a result the signal reflected by the body surface changes in unpredictable way. Thus traditional approaches used to eliminate self-interference in backscatter systems do not apply.<sup>1</sup>

## 2.4.2 Exploiting Non-Linear Behavior

How does one go about isolating the signal received from inside the body when the skin reflections are 80 dB (100 million times) stronger than the signal from inside the body? One approach to solve this problem is to enable the in-body backscatter device to shift the frequency. If the signal reflection from the skin and the in-body device are received at sufficiently different frequencies, one can filter the skin reflections out and receive just the backscatter signal.

In state-of-the-art backscatter systems ([96, 120, 258]), the standard way to shift frequencies is to use a small battery and operate a local high-frequency clock (with frequency  $f_c$ ) on the backscatter device. Thus, if the incident signal has frequency  $f$ , the backscatter device uses the local clock to generate frequency  $f_c$ , and RF mixers to mix these two frequencies and generate  $f + f_c$  or  $f - f_c$ . This is usually done to shift existing Wi-Fi signals to adjacent Wi-Fi frequency bands to avoid interference. This approach works well for IoT devices equipped with batteries to run the high-frequency (tens of MHz) clock, and with no size-restrictions. However, their battery requirement, size and complexity make them undesirable for in-body use.

So, the question becomes how we can shift frequencies without using any active wireless components. Our insight is to leverage non-ideal circuit behavior to generate frequency shifts. In traditional RF design, circuits are designed to exhibit linear behavior, i.e., if a circuit (like an RF amplifier) receives an input

---

<sup>1</sup>Gating methods used in radar to reject nearby reflections do not apply because the distance between implant and skin is only a few cm and hence the gating circuit has to operate at excessive speeds, e.g., picoseconds. Further, due to breathing the skin may move by more than a few centimeters, making it impossible to set a gating threshold that captures the implant's signal yet rejects skin reflections.

signal  $s$ , it's output defined by  $f_{linear}(s)$  is given by:

$$f_{linear}(s) = \gamma s \quad (2.6)$$

where  $\gamma$  is a complex number. Thus, the system just scales signal magnitude and adds a constant phase to the signal.

In contrast to linear systems which apply a linear transformation to the input signal, non-linear systems can apply generally polynomial transforms to the input signal. Specifically, for an input signal  $s$ , the output signal  $f_{non-linear}(s)$  is given by:

$$f_{non-linear}(s) = \gamma_0 s + \gamma_1 s^2 + \gamma_2 s^3 + \dots \quad (2.7)$$

where  $\gamma_0, \gamma_1, \dots$  are complex numbers. To understand what this non-linear behavior means for the frequencies contained in a signal, consider a signal  $s$  that is a sum of two sine waves of different frequencies  $f_1$  and  $f_2$ , i.e.  $s = \sin(2\pi f_1 t) + \sin(2\pi f_2 t)$ , where  $t$  is the time.

$$\begin{aligned} \gamma_1 s^2 &= \gamma_1 (\sin(2\pi f_1 t) + \sin(2\pi f_2 t))^2 \\ &= \gamma_1 (\sin^2(2\pi f_1 t) + \sin^2(2\pi f_2 t) + 2 \sin(2\pi f_1 t) \sin(2\pi f_2 t)) \\ &= \frac{\gamma_1}{2} ((2 - \cos(2\pi 2f_1 t) - \cos(2\pi 2f_2 t)) \\ &\quad + \cos(2\pi(f_1 - f_2)t) - \cos(2\pi(f_1 + f_2)t)) \end{aligned} \quad (2.8)$$

Here the last equation is derived using standard trigonometric formulae. The final step in equation 2.8 shows that the output signal contains different linear combinations of the input frequencies, i.e.,  $2f_1, 2f_2, f_1 - f_2$  and  $f_1 + f_2$ . Each of these frequency components are referred to as harmonics. Similarly, it can be shown that the third order term contains third order combinations like  $3f_1, 3f_2, 2f_1 + f_2, 2f_2 + f_1, 2f_2 - f_1, \dots$

Note that, this behavior is typically non-ideal in circuits and RF designers try to minimize non-linear behavior. However, in the context of in-body backscatter, this presents us with an opportunity to shift the signal in frequency using a non-ideal RF component. This allows us to transmit a signal composed of two sine waves at different frequencies  $f_1$  and  $f_2$ . When the in-body implant receives the signal, it

uses the non-linear behavior to create signals at frequencies which are combinations of the two-incident frequencies. While the human body reflects signals at just  $f_1$  and  $f_2$ , the signal reflected by the implant contains signals at frequencies  $f_1 + f_2$ ,  $f_1 - f_2$ ,  $2f_1 - f_2$ , and so on. This allows us to filter out reflected signals from the human body at  $f_1$  and  $f_2$  and isolate the signal reflected by the in-body device.

Finally, how can we produce non-linearities using simple passive components, i.e., components with no battery? As we said before, most RF-circuits are designed to exhibit linear behavior. Thus, to produce non-linearities, we turn to a fundamental non-linear component, the diode. A diode is the simplest non-linearity producing component, that is completely passive and is typically used as building block in larger, more complex circuits. By using a diode connected to an antenna, we can design a simple RF front-end for the implant. It can receive signals at two frequencies,  $f_1$  and  $f_2$ , and backscatter combinations of these frequencies, thereby allowing a receiver outside the body to isolate the backscattered signal.

### 2.4.3 System Design

We use the design shown in the inset in Fig. 2-3, which is similar to standard passive RFIDs except for the nonlinear diode connecting the antenna to the rest of the system. Given an input signal, the diode mixes the frequencies in the signal creating second and third order frequencies. These frequencies are fed to a switch, which modulates the signal. Specifically, the device can turn the switch on or off to communicate its data using on-off keying, as in passive RFIDs. The backscatter signal can be demodulated by the receiver using standard demodulation techniques. Note that, even for high data applications like smart capsules which take images of the GI tract, the data requirements are few 100 kbps since they typically transmit one or two small frames per second [19]. Thus, a simple encoding like on-off keying suffices for communication. Notice that our design is completely passive since a diode is a passive component that does not need to be powered. Furthermore, a diode is a very small and simple component that can be added to an in-body transceiver with negligible size change.

**Frequency Selection:** One important system design consideration for in-body backscatter is the choice of frequencies to be transmitted. There are two constraints that must be met.

- *Safety Limits:* The safety standards for in-body transmissions need to be met. As past work has shown [11], it is safe to transmit up to 28 dBm for an on-body antenna at frequencies around 1 GHz. This power level suffices for ReMix. Our system experiences a loss of around 80 to 100 dB in the

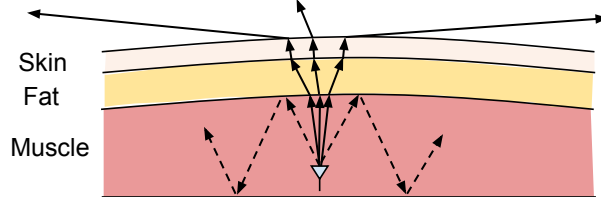


Figure 2-4: Due to high  $\epsilon_r$  for muscle, only the signal travelling along a small angle can exit the body. All other signals get reflected.

human body at frequencies around 1 GHz. But to be on the conservative side, even if one assumes the loss is up to 130 dB, the expected received signal strength is  $\approx -100$  dBm. This low power can support 1MHz wide on-off keying transmissions using standard receiver design [214, 47].

- *FCC Regulations:* The FCC regulates which frequencies can be used - specifically it has set aside several bands around 1 GHz (174 MHz-216 MHz, 470 MHz-668 MHz, 1395 MHz-1400 MHz, 1427 MHz-1432 MHz) for biomedical telemetry services, in addition to the ISM bands (FCC 15.241, 15.242, part 95, subpart H ([67])). ReMix transmitters can use any two frequencies in this range. For example, one can transmit at 570 MHz ( $f_1$ ) in the biomedical telemetry band and 920 MHz ( $f_2$ ) in the ISM band. The backscattered signals can then, for example, be received at 1490 MHz ( $f_1 + f_2$ ) and 1270 MHz ( $2f_2 - f_1$ ). The frequencies transmitted by the backscatter device inside the body do not violate FCC regulations because their power levels are much lower than FCC spurious transmission limits ([104]). In part 15.209, FCC limits the effective radiated power of spurious transmissions (like harmonics) of unlicensed devices to be less than -52 dBm (for frequency bands over 100 MHz), which is much larger than our backscattered power.

## 2.5 In-Body Backscatter Localization

We discuss the key challenges in positioning in-body devices using RF signals and how ReMix overcomes them. While we explain our insights in the context of backscatter, they naturally apply to non-backscatter RF signals.

## 2.5.1 Understanding the Positioning Challenge

State-of-the-art RF localization systems [244, 124, 231, 222] operate in two steps. In the first step, they use the phase of the wireless channel between the transmitter and the receiver to measure the angle-of-arrival of the signal or distance between the transmitter and the receiver. In the second step, they assume the path travelled by the signal is straight, and apply basic geometry to locate the transmitter. For in-body RF signals, both these steps are bound to fail if applied as is.

First, let us consider the phase of the wireless channel. As shown in Eq. 2.1, the phase in air or vacuum,  $\phi$ , is given by:  $\phi = -2\pi f \frac{d}{c} \bmod 2\pi$ . Here,  $f$  is the frequency of the signal and  $c$  is the speed of EM waves in vacuum. Hence, the phase of the wireless channel linearly depends on the distance travelled by the signal. In contrast, and as discussed in Sec. 2.2, the phase accumulated by the signal in a biomaterial is scaled by  $\alpha = Re(\sqrt{\epsilon_r})$ , where  $\epsilon_r$  is the electrical permittivity of the material. Thus, when the signal traverses the body, its phase is:

$$\phi = -2\pi \frac{f}{c} \left( \sum_i \alpha_i d_i \right) \bmod 2\pi, \quad (2.9)$$

where the sum is over the various materials traversed by the signal (air, skin, fat, muscle, etc.),  $\alpha_i$  is the real square-root of the electrical permittivity of material  $i$ , and  $d_i$  is the distance traveled in that material. This means that when the signal traverses multiple materials, the phase is no longer a simple function of the distance between the transmitter and the receiver.

Second, even if we could measure the distance traversed or angle-of-arrival of the signal, how does one map it to a location? As discussed in Sec. 2.2, the signal experiences refraction (bending) at the interface between different material (e.g., the interface between air and skin). Thus, the assumption that the signal travels between two points along straight lines no longer holds. As a result, the geometric model of intersecting distances or angles from multiple viewpoints doesn't apply.

Third, we need to take into account that our signal travels the entire distance at different frequencies. The signal from the transmitter to the backscatter device consists of two frequencies ( $f_1$  and  $f_2$ ), while the signal measured at the receiver after being backscattered is at  $f_1 + f_2$ ,  $2f_1 + f_2$ , and other frequency combinations. Together, these three factors are the main challenges facing in-body localization.

## 2.5.2 Principles of Positioning

Before we introduce ReMix’s positioning algorithm, we discuss a few key insights that we build our algorithm on.

**(a) RF signals exit the body from a small region on the surface:** In Sec. 2.2(e), we made the observation that it does not matter how the signal arrives from air, it enters the body only close to the direction of the normal on the surface. Since RF propagation is reversible, this means that it is also not possible for the RF signal to escape from the body through all possible directions. In fact, it can escape only from a small region around the normal on the surface, as shown in Fig. 2-4. The reason is the property of refraction. Specifically, when an RF signal travels from a high permittivity material (like human body) to a low permittivity material (like air), it bends *away* from the direction perpendicular to the interface between the materials. Substituting the electrical permittivity values for body tissues in Eq. 2.5 shows that the cone in Fig. 2-4 is about  $8^\circ$ . In-body signals that arrive more than  $8\text{-}10^\circ$  away from the normal on the surface typically reflect internally and do not escape to the air.

**(b) No in-body multipath:** As a corollary of the first point, in-body multipath either does not exist or is very weak compared to the direct path. Any signal that is reflected back into the body has to traverse multiple cm of human tissue and face multiple reflections before it can escape the human body. Because of the exponential attenuation caused by human tissue, this signal will be very low power compared to the direct path emanating out of the body. This is quite in contrast to large scale in-air localization systems where the line-of-sight path can be much weaker than multipath because of obstructions.

**(c) Material order and interleaving can be ignored:** The human body has multiple layers of tissues interleaved with each other. For example, skin and muscle are alike in electrical properties but are separated by fat which is closer to air. Further, the same material can appear in multiple layers (e.g., air-skin-fat-muscle-fat-muscle). This complex layering structure makes it challenging to model refraction at various interfaces. We can prove, however, that for parallel layers, order and interleaving can be changed with no impact on the total phase of the signal.<sup>2</sup> Since human tissues tend to be layered on top of each other, the assumption of parallelism is a reasonable approximation. We prove this observation in the appendix. We also verify this property empirically using multiple layers of pork belly in Sec. 2.9. This observation implies that the multiple layers of the human body can be rearranged for modeling and approximated to be

---

<sup>2</sup>Reordering of layers affects the amplitude due to more reflections.

grouped in two major layers: one layer comprising oil based tissues (like fat) and another layer comprising water based tissues (like skin and muscle).

## 2.6 ReMix's Localization Algorithm

We introduce a localization algorithm that is particularly customized for in-body RF signals. The algorithm works in two steps. First, it estimates the distances traveled by the signal as if it were traveling in air. We call such values *the effective-in-air distances*. Second, it models signal paths with linear splines (piecewise linear segments). The length of each segment refers to the stretch of the path in a particular material (air, fat, muscles). It then solves an optimization problem that maps the effective distances to the correct splines that match the actual paths traveled by the signal. (For simplicity, all phase equations are expressed ignoring the initial difference in oscillator phase between transmitter and receiver which can be measured during the calibration phase.)

### 2.6.1 Measuring Effective In-Air Distances

Consider a signal traveling from a transmitter to a receiver through  $L$  different biomaterials. Assume that it travels distance  $d_i$  in biomaterial  $i$ , with phase scaling factor  $\alpha_i = Re(\sqrt{\epsilon_{ri}})$ . We define *effective in-air distance*,  $d_{eff}$ , traveled by the signal as:

$$d_{eff} = \sum_{i=0}^{N-1} \alpha_i d_i \quad (2.10)$$

Combining Eq. 2.10 with Eq. 2.9, the phase,  $\phi$ , of the signal observed by the receiver is:

$$\phi = -\frac{2\pi f d_{eff}}{c} \bmod 2\pi. \quad (2.11)$$

Thus, an alternative definition for the effective in air distance is that, if travelled in air, it would result in the received phase.

So, how do we compute the effective distances? Recall that ReMix has two transmit antennas that transmit  $f_1$  and  $f_2$ , and a number of receive antennas. Let  $d_1$  and  $d_2$  be the effective distances from the two transmitters to the backscatter device, and  $d_r$  the effective distance from the backscatter device to

receiver  $r$ . The transmitters are transmitting frequencies  $f_1$  and  $f_2$ , while the receivers receive the non-linear mixing of these two signals at frequencies  $f_1 + f_2$ ,  $2f_1 - f_2$ , and other linear combinations. Let us consider the phase of  $f_1 + f_2$  measured at receive antenna  $r$ , which can be given by:

$$\phi_i = -\frac{2\pi}{c}(f_1d_1 + f_2d_2 + (f_1 + f_2)d_r) \bmod 2\pi \quad (2.12)$$

This phase equation is a combination of three components. The first two components correspond to the phase of the signal from the transmit antenna to the device. They combine based on the particular non-linear component of the signal that we receive. Since, we are considering just the non linear component  $f_1 + f_2$ , which is just the sum of the frequencies, the corresponding phases also add up (this follows directly from Eq. 2.8). Eq. 2.12 gives us one equation in terms of three unknowns,  $d_1$ ,  $d_2$ , and  $d_r$ . We need more equations to solve for these unknowns. Note, now, that the non-linearity generates various frequency mixes, which provide additional equations. For example, we can write a similar equation for  $2f_1 - f_2$ . The phase,  $\psi_i$ , measured at this frequency is given by:

$$\psi_i = -\frac{2\pi}{c}(2f_1d_1 - f_2d_2 + (2f_1 - f_2)d_r) \bmod 2\pi \quad (2.13)$$

Once again, note that the phase accumulated by the signal combines in the same way as the frequencies.

To simplify Eq. 2.12 and Eq. 2.13, we combine them as:

$$\begin{aligned} \phi_i + \psi_i &= -\frac{2\pi}{c}3f_1(d_1 + d_r) \bmod 2\pi \\ 2\phi_i - \psi_i &= -\frac{2\pi}{c}3f_2(d_2 + d_r) \bmod 2\pi \end{aligned} \quad (2.14)$$

Thus, we get equations expressed as summed distances from each of the transmitters to the receivers. At this point, we cannot use more harmonics to solve for individual distances since they will just yield equations that are linearly dependent on these two harmonics. However, we can use another receiver  $r'$  to get two additional equations that are functions of  $d_1$ ,  $d_2$  and  $d_{r'}$ . Thus, given at least two receive antennas, these four equations can be solved to obtain  $d_1$ ,  $d_2$ ,  $d_r$  and  $d_{r'}$ . More antennas can be used to improve accuracy of distance estimates.<sup>3</sup>

---

<sup>3</sup>We note that all phase equations are mod  $2\pi$ . To resolve ambiguity due to the phase wrapping around, ReMix, like past work [222], uses a small frequency band around each of the transmitted frequencies – i.e., instead of just transmitting  $f_1$  and  $f_2$ , ReMix sweeps through its transmission in a small band of 10 MHz around  $f_1$  and  $f_2$ .



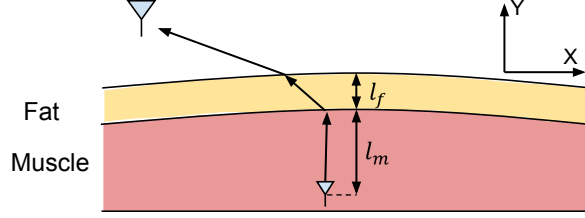


Figure 2-5: **Localization Model:** The signal from the in-body device travels through two layers (fat, muscle) to reach ReMix antennas.

## 2.6.2 Mapping Effective Distance to Actual Location

Now that we have the effective in-air distances between the in-body backscatter device and the transmit and receive antennas outside the body, we want to map those effective distances to the actual physical location of the backscatter device. At this stage we can drop the distinction between the transmit and receive antennas and treat all effective distances in the same way.

As we said before, the in-air effective distance does not translate into physical distances directly. Furthermore, since the signals do not travel in a straight line, the distances cannot simply be intersected from the different transmitters and receivers to get the right location. To solve this problem, we model signal propagation inside the human body as linear splines (instead of a straight line). Thus, propagation in each layer is linear, but across layers, it can change directions.

**Model:** As discussed before, human tissues can be classified as either oil (like fat) or water based (like muscle). Furthermore, different layers can be rearranged such that the muscle-based tissues occur together and the fat-based tissues occur together. Thus, we model the human body as a two-layer system, as shown in Fig. 2-5. For ease of exposition and visualization, we discuss the algorithm in the 2D XY plane. An extension to 3D is straightforward. The in-body backscatter implant is located at  $X$ , where  $X$  is a tuple of its  $(x, y)$  coordinates. The implant is covered by a layer of muscle with depth  $l_m$  (relative permittivity  $\epsilon_{rm}$ ). Then, there is a layer of fat, which has depth  $l_f$  (relative permittivity  $\epsilon_{rf}$ ). Thus, our model,  $\mathcal{M}$  has three latent variables  $(X, l_m, l_f)$ . In addition to these latent variables, the model has fixed parameters,  $\Theta$ : the position of each antenna,  $X_i$  for  $i = 1, \dots, N$ , and the permittivity of biomaterials  $\epsilon_{rf}$  and  $\epsilon_{rm}$ . The observations made by the model are the effective distance measurements,  $d_i$ , from the implant to each of the antennas. Then, the goal of the model is to estimate the hidden variable  $(X, l_m, l_f)$  given a set of observations.

**Constraints:** Next, how does the model constrain the structure of the splines for each path? Let us

consider the effective in-air distance  $d_i$  measured at the  $i^{th}$  antenna. In our model, the effective in-air distance is modeled by a spline comprised of 3 different segments: an in-air segment of length  $d_a^i$ , in-fat segment of length  $d_f^i$  and in-muscle segment,  $d_m^i$ . Together, when these physical distances are scaled by their respective scaling factors and summed together, they should yield the effective-in-air distance  $d_i$ . The estimation of the individual segments of these splines is governed by two sets of constraints:

- **Refraction Constraints:** Let us say the angle of incidence inside fat, muscle and air is  $\theta_f^i, \theta_m^i, \theta_a^i$  respectively. Then:

$$\begin{aligned} Re(\sqrt{\epsilon_{ra}}) \sin \theta_a^i &= Re(\sqrt{\epsilon_{rf}}) \sin \theta_f^i \\ Re(\sqrt{\epsilon_{rm}}) \sin \theta_m^i &= Re(\sqrt{\epsilon_{rf}}) \sin \theta_f^i \end{aligned} \quad (2.15)$$

- **Geometric Constraints:** If  $(X_i - X)_1$  denotes the horizontal dimension of the difference between two positions, then:

$$\begin{aligned} d_a^i &= \frac{l_a}{\cos \theta_a^i}, d_f^i = \frac{l_f}{\cos \theta_f^i}, d_m^i = \frac{l_m}{\cos \theta_m^i} \\ d_a^i \sin \theta_a^i + d_f^i \sin \theta_f^i + d_m^i \sin \theta_m^i &= (X_i - X)_1 \end{aligned} \quad (2.16)$$

where  $l_a$  is the depth of air which is equal to the total distance along the vertical dimension minus the depth of muscle and fat combined.

We now have a system with 6 variables ( $d_a^i, d_m^i, d_f^i, \theta_m^i, \theta_f^i, \theta_a^i$ ) and 6 independent equations (Eq. 2.15 and 2.16). This is solvable numerically using ray tracing methods. Finally,  $d_a^i, d_f^i$  and  $d_m^i$ , thus obtained are functions of the latent variables  $(X, l_m, r_f)$ . Thus, we denote the length of the segments of the spline corresponding to antenna  $i$  in air, fat and muscle by  $d_a^i(X, l_m, l_f), d_f^i(X, l_m, l_f), d_m^i(X, l_m, l_f)$  respectively.

**Optimization:** Now that we modeled the individual segments of the splines as functions of the latent variables in the model, we want to leverage the observed effective in-air distances to estimate the latent variables. Specifically, for each antenna, we minimize the L2-norm of the observed effective in-air distances ( $d_i$ ) and the distance obtained by the scaled sum of the spline segments ( $d_a^i(\cdot) + \alpha_f d_f^i(\cdot) + \alpha_m d_m^i(\cdot)$ ). As before,  $\alpha = Re(\sqrt{\epsilon_r})$ . Thus, combining distance measurements from multiple antennas, our optimiza-

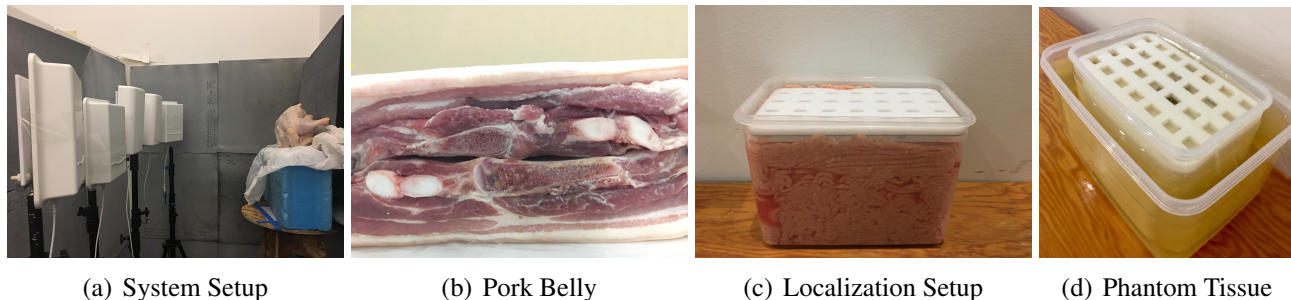


Figure 2-6: **ReMix Implementation:** ReMix’s system setup is shown in (a). To emulate human tissue behavior, we conduct experiments with animal tissues like (a) whole chicken , (b) pork belly, and (c) ground chicken in addition to (d) human tissue phantoms. In (d), the outer layer is fat phantom and the inner box is filled with muscle phantom.

tion function can be written as:

$$\hat{X}, \hat{l}_m, \hat{l}_f = \arg \min_{X, l_m, l_f} \sum_{i=1}^N \|d_a^i(\cdot) + \alpha_f d_f^i(\cdot) + \alpha_m d_m^i(\cdot) - d_i\|^2 \quad (2.17)$$

where  $\hat{X}, \hat{l}_m, \hat{l}_f$  are the optimal values of the latent variables. This optimization problem is convex in each of the hidden variables  $(X, l_a, l_f)$  for  $\epsilon_r$  value ranges of human tissues. Furthermore, it has one local maximum. It can be framed as a standard convex optimization problem and solved using convex optimization techniques. By doing so, this optimization can accurately estimate the position of the device by modeling the spline structure.

## 2.7 Implementation

ReMix’s implementation has two primary components: the in-body backscatter device and the out-of-body ReMix transceiver that generates the two frequencies and listens for the backscatter signal. For our implementation of the in-body backscatter device, we use a Schottky detector diode from Skyworks Solutions [206]. The size of the diode is 1.6 mm by 0.8 mm. We connect this diode to the PC30 dipole antenna from Taoglas [215]. We note that this antenna is 7.5 cm long and its gain is around 0 dB in-air for the band of interest. However, its an off-the-shelf antenna optimized for in-air transmissions and its design is not optimized for in-body transmissions. Smaller antennas the size ([72, 155, 144]) of a grain of rice have been used in in-body RFIDs and one could leverage these designs for better performance. However,

the exploration of antenna design is out of the scope of this work.

For out-of-body transceiver, we use two patch antennas for transmissions and three patch antennas for reception. We use this set of five antennas for localization. For communication, a single receive antenna is sufficient. A picture of this setup is shown in Fig. 2-6(a). To avoid mixing of the two transmitted frequencies in the transmission circuit, we use separate transmit chains for both the transmitted frequencies. The antennas are typically placed from 50 cm to 2 m away from the subject for our experiments. These antennas are connected to USRP software radios ([66]). The transmit frequencies used are 830 MHz ( $f_1$ ) and 870 MHz ( $f_2$ ). We use two harmonics 910 MHz ( $2f_2 - f_1$ ) and 1700 MHz ( $f_1 + f_2$ ). Our choice of frequencies is illustrative and another set of frequencies can be chosen as discussed in Sec. 2.4. Our choice was governed by the availability of off-the-shelf hardware in these bands.

The signal transmission and reception were done using USRP X300 software radios and UBX daughterboards. The USRPs were synchronized using an external 10 MHz clock. The USRPs were programmed using the UHD library in C. The signals received from the USRPs were collected on a desktop PC over ethernet and the received signals were processed in Matlab.

## 2.8 Human Tissue Emulation

We emulate human tissues using two methods commonly used in the literature on in-body imaging and power transfer [160, 192, 11, 49, 178]. First, we use animal tissues like whole chicken (Fig. 2-6(a)), ground chicken meat (Fig. 2-6(c)) and pork belly (Fig. 2-6(b)). These tissues have complex layer structures and contain skin, muscle, fat, bones, etc. Further, the EM properties of these tissues is similar to human tissues [103, 209]. Thus, they provide an ideal environment for emulating human tissues.

Second, we use phantom tissues have been proven to exhibit electrical properties similar to human tissues in frequencies up to 2500 MHz [170, 8, 108, 134]. These phantoms can emulate both water-based (muscle) and oil-based (fat) tissues. The muscle phantoms are made using Polyethylene powder and Agarose [108], and the fat phantoms are made mainly of vegetable oils and gelatin [134]. We vary the amount of fat and muscle to emulate human tissues with more fat or more muscle.

An in-body implant placed inside chicken tissue or human phantom is not visible to the naked eye. Thus, for localization experiments where precise ground truth location is required, we use the setup shown in Fig. 2-6(c). The ground chicken meat or human phantom is packed in a plastic container and covered

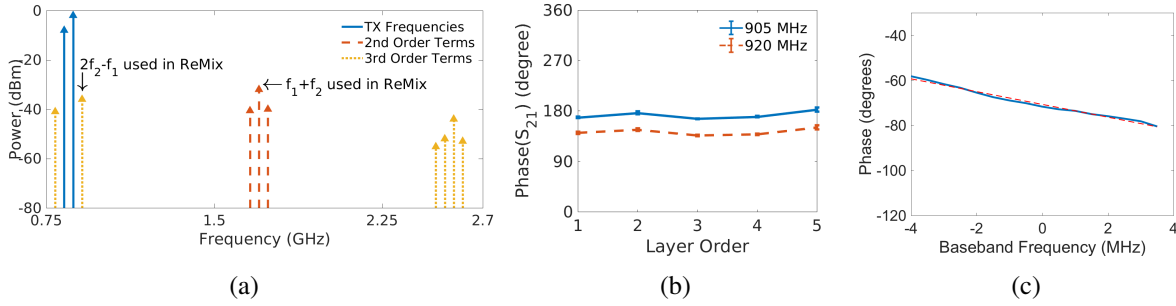


Figure 2-7: **ReMix Microbenchmarks:** (a) ReMix’s design produces non-linear mixing of transmitted frequencies which can be used to filter out strong human body reflections, (b) Interchange of layers (Tab. 2.1) does not impact signal phase, (c) Linear relationship between phase (in blue) and frequency shows non-existent or mild multipath in human body.

with a lid. The lid has slits cut into it using precise (sub-millimeter) laser cutting methods. The implant is placed through the slits at a fixed depth below the surface. By varying the slit that the implant is inserted into, the implant’s location in the human phantom or chicken tissue can be changed.

## 2.9 Empirical Results

We discuss our evaluation of ReMix below.

### 2.9.1 Microbenchmarks

**Non-linear Behavior of Diode:** ReMix exploits the non-linear behavior of a diode for isolating the in-body backscatter signal. A mathematical explanation of the non-linear behavior is given in Sec. 2.4. To understand the practical manifestation of the non-linear behavior, we design a simple experiment. A diode is connected to an antenna and placed in the air. Two transmitters, each transmitting a single frequency signal, are placed at a distance of 1 m from the diode-antenna system. A receive antenna is placed 1 m from the diode on the same side as the transmitter. The power observed by this receiver on different frequencies is plotted in Fig. 2-7(a).

As can be seen in the figure, the diode produces multiple non-linear combinations of the input signal. The second order harmonics ( $2f_1, 2f_2, f_1 + f_2$ ) are lower in power than the main signal peaks, but higher in power than the third order harmonics ( $2f_1 - f_2, 3f_1$ , etc.). The variation in power in different frequency bands is caused by: (a) the diode conversion loss, i.e., different harmonics have different power after the

Config.	Layer Structure
1	Skin, Fat, Muscle, Fat, Muscle, Muscle, Bone
2	Muscle, Fat, Muscle, Fat, Skin, Muscle, Bone
3	Skin, Fat, Muscle, Fat, Muscle, Bone, Muscle
4	Muscle, Fat, Muscle, Fat, Skin, Bone, Muscle
5	Bone, Muscle, Skin, Fat, Muscle, Fat, Muscle

Table 2.1: Configurations for Tissue-Layer Interchange Experiment

non-linearity of the diode is factored in, (b) the propagation loss over air – since different frequencies have different loss over air, the observed power is higher for lower frequencies.

This experiment indicates that a diode’s non-linear behavior produces reflected signal at multiple frequency bands. ReMix uses these signals for in-body backscatter.

**Interchange Layers:** In Sec. 2.5.2, we observed that the order of layers can be changed without impacting the phase of the signal. In order to validate this observation, we use a big chunk of pork belly. Pork belly is composed of layers of fat, muscle and bones. A big chunk of meat is placed below the transmit antenna so as to avoid any leakage over the air. Then, the transmit antenna is covered by different sequences of multiple layers given in Tab. 2.1. Finally, a receive antenna is placed on top of this setup, about 10 cm away. Each configuration of the layers given in Tab. 2.1 is repeated 5 times. The signal phase is measured at the receive antenna at two different frequencies.

The phase observed on the receive antenna and its standard deviation are plotted in Fig. 2-7(b). As seen in the figure, the phase remains almost constant across these measurements. The standard deviation in these measurements is 8 degrees which is within quite small and within the measurement errors. This shows that re-ordering different tissue layers in-body do not impact the phase of the received signal.

**Lack of in-body Multipath:** In 2.5.2, we discussed why the direct path for a signal from inside the body is much stronger than any reflected paths. While mapping the multipath directly would either need a large antenna array or a large frequency bandwidth, we use an approximate method to verify this assumption. Specifically, for a fixed distance between transmitter and receiver, signal phase changes linearly with frequency if there is no multipath. When there is significant multipath, the phase is no longer linear. Thus, to verify whether this observation holds, we put ReMix’s backscatter device inside a box full of chicken meat and observe the phase of the reflected signal across a 8 MHz frequency band. Each of the transmitter frequencies is moved by up by 8 MHz in steps of 0.5 MHz and the received signal phase is observed and plotted in Fig. 2-7(c). As can be seen in this figure, the phase has a linear relationship with distance. This

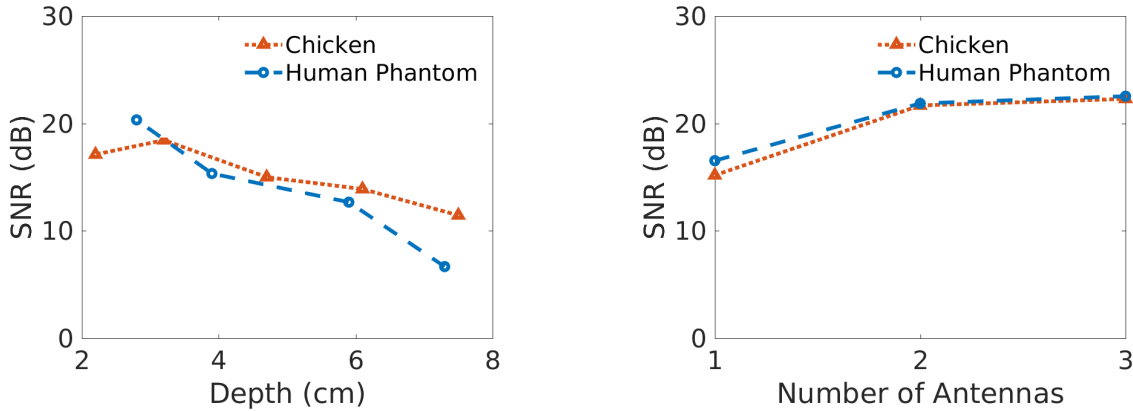


Figure 2-8: **ReMixBackscatter**: ReMix can achieve an average SNR of over 15 dB for 1 MHz frequency band even when the tissue depth is up to 8 cm (human muscle depth is 2-3 cm [62]). This SNR can be improved by combining multiple antennas.

indicates that the in-body multipath for ReMix is mild to non-existent.

## 2.9.2 Backscatter Communication

To evaluate the backscatter communication capabilities of ReMix, we use three different setups: ground chicken meat, human phantom tissues and whole (dead) chicken. The human phantom is designed to have 1.5 cm fat, followed by muscle. In each of these setups, we measured the Signal to Noise ratio (SNR) for depths up to 8 cm of tissue. Typically, human abdomen muscle can be as deep as 1.6 cm and the small intestine is around 1 cm deep [62]. Since muscle and intestine have high attenuation for RF signals, our tests including 7-8 cm muscle should present an insight into the worst case behavior of ReMix’s communication capabilities. Finally, we operate the system at a 1 MHz bandwidth for all these measurements.

We plot the SNR observed at a single harmonic in Fig. 2-8. As can be seen in the figure, the SNR of the signal decreases as the depth in tissue increases. The average SNR obtained in chicken and phantom are 15.2 dB and 16.5 dB respectively. Furthermore, even at depths of 8 cm, a depth which is larger than human muscle depth, the SNR is as high as 7 to 11 dB. Notice that the human phantom and the chicken meat exhibit similar behavior. This is along expected lines since they have similar dielectric properties.

Finally, while we cannot make precise measurements and incisions in whole dead chicken, we measure the SNR for 5 random locations inside the chicken body. The mean SNR observed was around  $23\text{dB}$ . This is higher than ground chicken or human phantom because the muscle thickness in the whole chicken is just

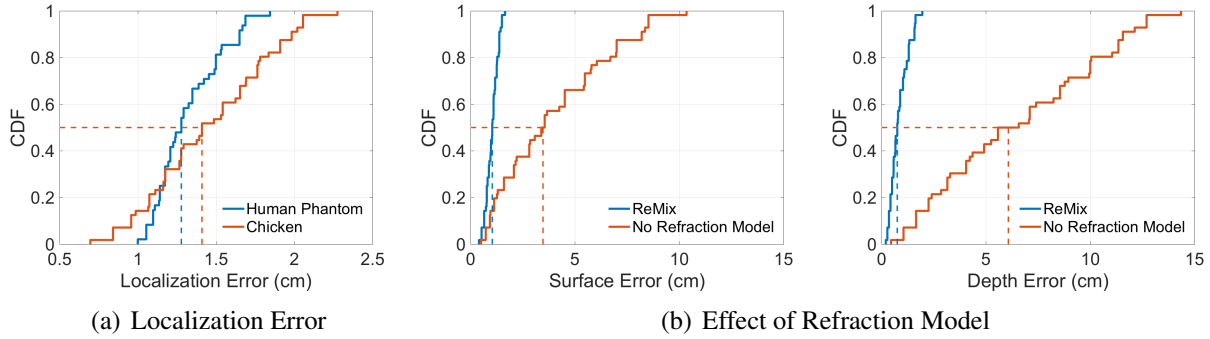


Figure 2-9: **ReMix Localization Results:** ReMix can achieve high localization accuracy of 1.3 cm in human phantom and 1.4 cm in chicken.

2-5 cm, which is lower than the muscle thickness we analyzed for ground chicken and human phantom.

**Combining Across Antennas:** The SNR reported above is the SNR for a single antenna. Since ReMix has multiple receive antennas, it can combine the signal across these antennas using techniques like maximal ratio combining (MRC) [216] to achieve higher SNR. Fig. 2-8 plots the SNR after combining the signal received at our 3 receive antennas using MRC. As expected, the combination gives us an average gain of 5 – 6 dB with 3 antennas. The performance is similar for both, the chicken tissue and the human phantom. This implies that we can use the multiple antennas to improve SNR in-body.

**Data rates:** ReMix uses On-off keying (OOK) for its data transmission. OOK is robust to noise and can achieve a datarate of  $1\text{Mbps}$  with a bit error rate of  $10^{-4}$  at SNR of around 12 dBm and bit error rate of  $10^{-5}$  at SNR of around 14 dBm [214, 47]. In realistic cases (muscle depth < 5 cm), the SNR of ReMix is in the 12-20 dB range even with a single receive antenna. Thus, ReMix can provide efficient communication for in-body devices like a wireless capsule endoscope, which have a requirement of few hundreds of kbps.

### 2.9.3 In-Body Localization

We evaluate ReMix’s localization accuracy using two different arrangements. First, we use chicken meat, covered with fixed size slits in a plastic cover (shown in Fig. 2-6(c)). The slits are 1 inch apart along each dimension. In the second setup, we use human phantoms. The human phantom tissues are designed to emulate layers of fat and muscle *a la* the human body. As shown in Fig. 2-6(d), the muscle phantom is placed inside the smaller box covered with slits spaced 1 inch apart along each dimension. The muscle phantom is surrounded by a layer of fat. The thickness of the fat layer is varied between 1 – 3 cm randomly



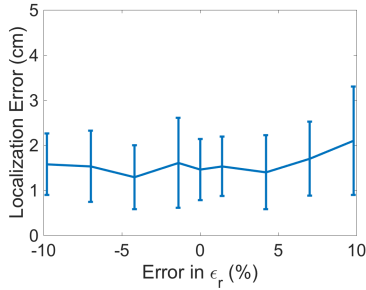


Figure 2-10:  $\epsilon_r$  **Variance:** For typical variance in human tissue properties, ReMix’s localization error continues to be  $< 2$  cm.

to emulate variation in body structure. The goal of this setup is to be true to human tissue layers.

We measure localization error of the backscatter device placed through the slits in the cover. We make 50 different measurements in each of the two setups, and report the CDF of localization error in Fig. 2-9(a). As the figure shows, the median localization error for ReMix is 1.4 cm in chicken tissue and 1.27 cm in human phantom, and the maximum is 2.2 cm in chicken tissue and 1.8 cm in human phantom tissue. Note that, this error is much lower than the 5 cm accuracy required for depositing biomarkers in the colon for wireless capsule endoscopy [178] and can enable this application. Furthermore, this error is 2X lower than the theoretical lower bound on RSS based in-body localization achievable with 32 antennas [250].

To delve deeper into the results, we discuss two aspects of the localization. First, we study how the use of refraction model impacts the localization error. Thus, we use ReMix’s distance based model without the refraction model and observe the error along two axis: lateral localization error along the surface of the body and error in measuring the depth of the device in the body. The CDF of the surface error and depth error are shown in Fig. 2-9(b). The median surface and depth error for ReMix are 1.04 cm and 0.75 cm respectively. Without ReMix’s refraction model, however, the error increases. The surface error and depth errors in that case are 3.4cm and 6.1cm respectively. Note that, without the refraction model, the error in estimating depth of the in-body device is higher than the error in estimating position along the surface. A simple way to understand is to compare this with refraction in visible light. When one observes a coin placed in water, it appears at the same lateral position, but appears much closer than its actual position in terms of depth. This is similar to what happens when one fails to account for the refraction model and hence, there is larger error in depth measurement as compared to the lateral error measurement.

Finally, there is some natural variation between people in terms of the value of  $\epsilon_r$  for the same tissue type. In our evaluation we use the average value of  $\epsilon_r$ . We want to understand how much of an error

can the use of this value  $\epsilon_r$  introduce. We change the value of  $\epsilon_r$  by up to 10%, which is similar to the natural variation observed in past work [212]. We report the localization error as a function of the change in Fig. 2-10. As can be seen, the error increases as a function of perturbation of this value. However, even when the change in  $\epsilon_r$  is 10%, the error in positioning the in-body device is less than 2.5cm. This shows that the natural variation of human beings can be accommodated by ReMix’s design. In addition to that, it also indicates that there is a potential for improving the accuracy by customizing the parameters for each patient.

## 2.10 Limitations and Future Work

Finally, a few points are worth elaborating:

- ReMix made a few approximations, like grouping skin and muscle in a single layer to reduce model complexity, and using an average value for  $\epsilon_r$  though different people may have mild variations. In making these assumptions, we strived to balance model complexity with the accuracy requirements. We believe these approximations are acceptable for applications like smart capsules. Future work can extend the model to eliminate these approximations and address applications with more stringent localization requirements (e.g., mm level accuracy for tumor localization in radiation therapy).
- We made our best attempt to emulate human tissues by using phantoms and dead animal tissues for our evaluation, but we did not conduct measurements in live animals. While the analysis and experiments should stand to be correct in living tissue, experiments in live animals fall in the realm of future work.
- In medical applications, typically, there is availability of side channel information, like one-time MRI scans, that can throw more light on the exact composition of the human body. ReMix does not rely on such extra information, but future work that targets specific applications with more stringent requirements might benefit from multi-modal input data.

## 2.11 Conclusion

We present ReMix, the first system to demonstrate deep-tissue backscatter communication and localization. ReMix achieves this goal using two design principles. First, it uses circuit non-linearities to shift

the frequency of the backscatter signal to avoid interference from surface reflections. Second, it presents a time-of-flight localization algorithm that accounts for in-body signal refraction. Experiments with animal tissues and human phantoms demonstrate ReMix's effectiveness.

## Chapter 3

# FarmBeats: An IoT Platform for Data-driven Agriculture

The demand for food is expected to double by 2050, primarily fueled by an increase in population and upward social mobility[219]. Achieving this increase in food production is even more challenging because of receding water levels, climate change and shrinking amount of arable land. According to International Food Policy Research Institute, data-driven techniques can help us achieve this goal by increasing farm productivity by as much as 67% by 2050 and cutting down agricultural losses [80].

In fact, field trials have shown that techniques that use sensor measurements to vary water input across the farm at a fine granularity (precision irrigation) can increase farm productivity by as much as 45% while reducing the water intake by 35%[18]. Similar techniques to vary other farm inputs like seeds, soil nutrients, etc. have proven to be beneficial [121, 161]. More recently, the advent of aerial imagery systems, such as drones, has enabled farmers to get richer sensor data from the farms. Drones can help farmers map their fields, monitor crop canopy remotely and check for anomalies. Over time, all this data can indicate useful practices in farms and make suggestions based on previous crop cycles; resulting in higher yields, lower inputs and less environmental impact.

While these techniques for agriculture have shown promising results, their adoption is limited to less than 20 percent farmers owing to the high cost of manual sensor data collection (according to US Department of Agriculture [147]). Automating sensor data collection requires a dense deployment of sensors, which is prohibitively expensive for farmers as well as logistically infeasible as they inhibit the workflow of farmers who need to drive tractors or bullock carts through a dense maze of sensors. Furthermore, one

needs to establish network connection to these sensors. However, existing connectivity solutions [53, 74] require a cellular data logger to be attached to each sensor (see Table 3.1 for a detailed comparison). These loggers cost around \$1000 each in equipment cost with over \$100 in subscription fee. They are also limited in the amount of data that they can send to few kilobytes per day. Clearly, these solutions do not scale up for large farms and cannot support high bandwidth sensors like cameras and drones, which rely on sending all their data to the cloud for processing [197, 51]. This situation is further worsened by the fact that farms typically have limited cellular coverage [119] and are prone to weather-based Internet outages.

In this chapter, we present FarmBeats, an end-to-end IoT platform for data-driven agriculture, that enables seamless data collection from various sensor types, i.e., cameras, drones and soil sensors, with very different bandwidth constraints. FarmBeats operates with a sparse deployment of sensors and can ensure system availability even in the face of power and Internet outages caused by bad weather; scenarios that are fairly common for a farm. FarmBeats enables cloud connectivity for the sensor data to enable persistent storage as well as long-term or cross-farm analytics. We have deployed FarmBeats in two farms in the US over a period of six months and used FarmBeats to enable three applications for the farmer: precision agriculture, monitoring temperature and humidity in food storage, and monitoring animal shelters. In designing FarmBeats, we solve four key challenges.

First, we observe that combining data from multiple data streams can allow us to operate with a sparse sensor deployment. Specifically, we work with two data streams: visual data from UAVs (unmanned aerial vehicles, an increasingly common farm tool), and sparse sensor data from ground sensors. UAVs provide measurements that are dense in space (continuous visual maps), but sparse in time. On the other hand, ground sensors are sparse in space, but dense in time – the sensors can be frequently polled to provide measurements. We built a new graphical model that combined the two streams to provide dense space-time sensor maps of the farm. Specifically, our system operates on the teacher-student model where we train the graphical model to map visual features to sensor values using the sparse sensor deployments as training data. This model, once trained, is used to predict sensor values in parts of the farm which have no sensors, thereby reducing the requirement of sensor deployment by one to two orders of magnitude.

Second, to enable connectivity within the farm, FarmBeats leverages recent work in unlicensed TV White Spaces (TVWS) [68, 28, 187] to setup a high bandwidth link from the farmer’s home Internet connection to an IoT base station on the farm. Sensors, cameras and drones can connect to this base station over a Wi-Fi front-end. This ensures high bandwidth connectivity within the farm. However, due

to the lack of power on the farm, the base station is powered by battery-backed solar power which suffers from power unreliability depending on weather conditions. As shown in past work [94, 200], cloudy weather can reduce solar power output significantly and drain the batteries of the base station to shut it down. To solve this problem, FarmBeats uses a novel weather-aware IoT base station design. Specifically, it uses weather forecasts to appropriately duty cycle different components of the base station. To the best of our knowledge, this is the first weather-aware IoT base station design.

Third, Internet connection to the farm is typically weak making it challenging to ship high bandwidth drone videos (multiple GBs) to the cloud. Furthermore, farms are prone to weather-related network outages that last weeks. Such system unavailability impedes a farmer’s ability to take adequate preventive actions, do UAV inspections and leads to loss of valuable sensor data. Thus, FarmBeats uses a Gateway based design, wherein a PC at the farmer’s home serves as a gateway for the farm data. The FarmBeats Gateway serves two purposes: a) it performs significant computation locally on the farm data to consolidate it into summaries that can be shipped to the cloud for long-term and cross-farm analytics, and b) the gateway is capable of independent operation to handle periods of network outage, thus leading to continuous availability for the farmer.

Finally, while drones are one of the most exciting farm sensors today, they suffer from poor battery life. Getting aerial imagery for a farm requires multiple drone flights and a long wait time in between when the batteries are being charged. We use the fact that farms are typically very windy, since they are open spaces. Thus, we incorporate a novel path planning algorithm in the FarmBeats gateway, that leverages wind to help the drone accelerate and decelerate, thereby conserving battery. This algorithm is motivated by how sailors use winds to navigate sailboats.

We use the FarmBeats system to enable precision agriculture applications on two farms: one in Washington state and the other in upstate New York. While traditional farming treats the farm as a homogeneous piece of land, precision agriculture adapts the farm inputs over different parts of the farm depending on the requirement. Precision agriculture techniques require a precision map with information about each location in the farm, for example, the soil temperature, soil moisture, nutrient levels, etc. To construct this precision map, existing solutions for precision agriculture require a dense deployment of in-ground sensors [147]. A dense deployment of sensors becomes expensive (as well as cumbersome to manage) as the size of the farm grows. Unless these sensors are deployed densely within a farm, the estimated precision map can be very inaccurate, as we show in Section 3.6. Since FarmBeats’s gateway has access

to both the drone videos and sensor data, it enables a novel low-cost mechanism that uses drone videos in combination with sparse ground sensors to generate precision maps for the farm. To the best of our knowledge, this is the first system that can combine the temporal data from sensors, with the spatial data from drones to construct an instantaneous precision map of the farm, such as the one in Figure 3-5.

Beyond FarmBeats's application in precision agriculture, farmers have so far used FarmBeats for two other applications. First, the farmers have been using FarmBeats to monitor temperature and humidity in storage spaces to ensure that the produce does not go bad. Second, the farmers have plugged in cameras at different locations, to monitor cow sheds, selling stations etc<sup>1</sup>.

**Contributions:**To summarize, FarmBeats makes the following key contributions:

- **Long-term large scale deployment:** Our deployments have run over 6 months in each of the farms and collected over 10 million sensor measurements, 1 million camera images and 100 drone videos
- **Novel Weather-Aware IoT Base Station Design:** Adding weather awareness into the IoT base station reduced the base station down time to zero as opposed to greater than 30% downtime during the same month in the previous year in an earlier version of our deployment
- **Novel Inference Techniques for Compression of Aerial Imagery Data:** FarmBeats's multi-modal inference method allows reduction in sensor deployment by one to two orders of magnitude while allowing the gateway to achieve a median compression of 1000 times from an aerial drone video to the sensor summaries sent to the cloud. Further, the gateway remained available even when the Internet connectivity to the farm faced a week-long outage
- **Wind-Assisted Drone Flight Planning Algorithm:** FarmBeats's flight planning algorithm improves the area covered by a single drone flight by 30%

### 3.1 IoT Platform: Objectives

In building FarmBeats, we target the following goals:

- **Availability:** The platform should have negligible downtime. When there is an outage (for example, due to power or network failure), data collection from the sensors should not stop and the platform should continue to deliver services to the farmers.

---

<sup>1</sup>Supplementary Material includes detailed description of FarmBeats applications and usage.

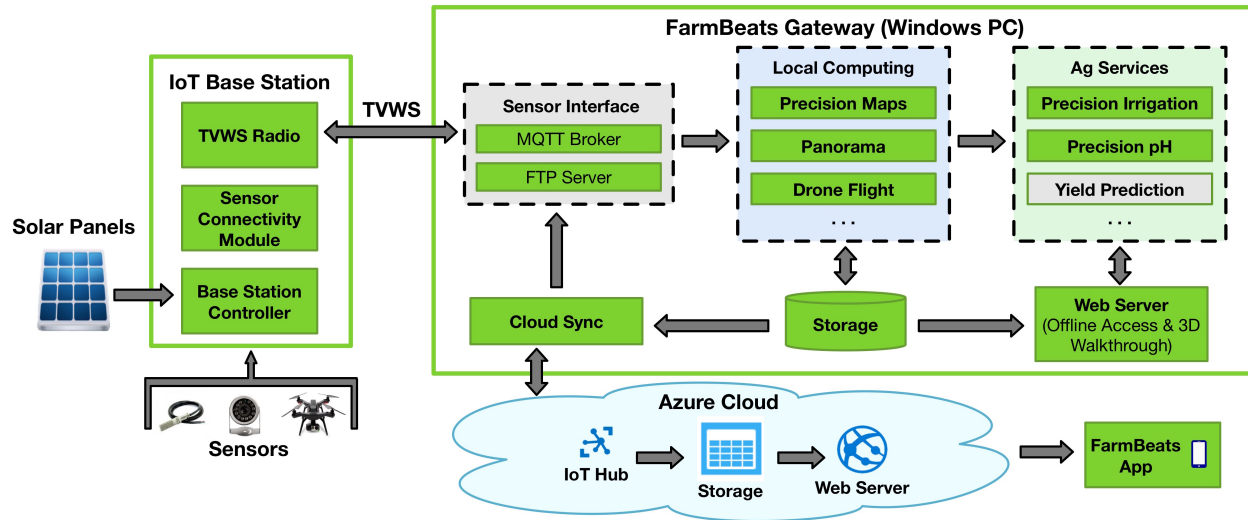


Figure 3-1: FarmBeats System Overview

- Capacity:** It should support sensors with widely varying requirements: pH sensors reporting few bytes of data to drones sending gigabytes of video. Similarly, the system should be capable of supporting end-user applications with varying needs: from a precision irrigation application that needs the latest sensor data for the entire farm to a crop suggestion application that needs just high level productivity data but across several growing seasons
- Cloud Connectivity:** Several farming applications, such as crop cycle prediction, seeding suggestions, farming practice advisory, etc. rely on long term data analytics. Besides, a farmer may want to access some applications even when he is not on the farm. Thus, the IoT platform must enable pushing data to the cloud.
- Data Freshness:** Stale sensor data from the farm can make applications suggest incorrect courses of action to the farmer. Gaps in historical data can also cause applications to misbehave. Moreover, stale data leads to bad user experience. Thus, the platform must strive to maintain maximum data freshness.

## 3.2 The FarmBeats IoT Platform

While these objectives have been fairly successfully achieved by home IoT platforms like Amazon Echo, achieving these objectives in an agricultural setting introduces several challenges for two main reasons: access and environmental variability. As discussed before (and as shown in Table 3.1), farms do not have



Technology	Cost	Data Restriction
Cellular Connection (Decagon Devices)	Per sensor fee: 1000\$ + 100\$ annual fee	Restricted to sensor data; Uploads every 15 mins at best
Mesh Networks (Ranch Systems)	Base station: 3500\$ + 750\$ annual fee; Per sensor fee: 1100\$ + 60\$ annual fee	Maximum 25 mesh nodes per base station
Satellite (Iridium)	Per sensor fee: 800\$ + 100\$ monthly fee	Restricted to 2.5 Kbps

Table 3.1: Cost Comparison of Farm Sensor Networking Solutions

access to power and high-bandwidth Internet connectivity unlike indoor IoT systems. Furthermore, energy harvested from the environment and weak network connectivity to the farm is susceptible to failures due to weather variability. So, the key question for the design of FarmBeats is: how does one design an IoT platform to meet the objectives in a highly variable, resource constrained environment?

### 3.2.1 Design Decisions

An overview of the system is given in Figure 3-1. Here, we discuss the main design decisions.

To achieve farm connectivity over long range, we leverage recent work in the TV White Spaces [28, 187, 68] to setup a high-bandwidth connection from the farmer’s home to the farm. However, sensors, drones and cameras typically do not support TVWS. Thus, in order to maintain compatibility with sensors along with long-range high bandwidth connectivity, we deploy a two-layer hybrid network. We use a TVWS link to connect the farmer’s home Internet connection to a few IoT base stations on the farm. Since it is a high bandwidth backhaul link, each base station can accommodate sensors, as well as cameras and drones. At the second layer, the IoT base station provides a Wi-Fi interface for connections from sensors and other devices. The Wi-Fi interface ensures that the farmer can not only connect most off-the-shelf farming sensors, cameras and drones; but they can also use their phone to access farming productivity apps.<sup>2</sup>

Variability in harvested solar energy leads to IoT base station downtime in overcast conditions. In fact, in our early deployments, power failures due to environmental factors were the major cause of unavailability. While past work has dealt with this problem in the context of single sensors [94, 226, 200] by duty cycling the sensors, the same approach does not work for a base station. Specifically, the base

<sup>2</sup>Future iterations of the systems would add multiple interfaces to the base station to enable compatibility with more sensor types.

Data Requirement	Applications
Immediate Descriptive Data	Precision irrigation, virtual walkthroughs, productivity apps, farm monitoring, ...
Long-term Summarized Data	Crop suggestions, seed distribution, yield monitoring, financial management, animal health statistics, ...

Table 3.2: Application classification based on requirements

station has multiple components with different power requirements and duty cycling costs. For example, a farmer is typically inactive at night and is unlikely to check the farm data. So, turning the TVWS device off (which consumes 5x more power than the rest of the base station) can enable the base station to collect data (in a cache) from the sensors more frequently. Further, FarmBeats enables the farmer to turn the base station on to access Wi-Fi for productivity applications, while they are on the farm. This adds another layer of uncertainty in the duty cycling plan. Thus, we propose a novel duty cycle policy (in Section 3.3) wherein the different components of the base station are duty cycled at different rates; while explicitly accommodating these constraints.

Finally, given the weak internet connectivity to the farm, a naive approach of pushing all the data to the cloud does not work. We make the key observation that the data requirements of the farming applications can be broadly classified into two main categories: immediate detailed data and long-term summarized data. Table 3.2 summarizes how the industrial and research applications of farm data can be classified into these two categories. This categorization enables a gateway based IoT design for FarmBeats. The local gateway sits at the farmer’s home at the other end of the White Space link and performs two functions: a) creates summaries for future use and ships them to the cloud and b) delivers applications that can be provided locally. The summaries are several orders of magnitude lower in size than the raw farm data (3-4 orders of magnitude smaller in case of the precision agriculture application discussed later) and hence, respect the harsh bandwidth constraints.

### 3.2.2 Architecture

The FarmBeats system has the following components:

**Sensors & Drones:** FarmBeats uses off-the-shelf sensors for its applications. Each sensor measures specific characteristics of the farm, such as soil moisture and soil pH, and reports this data to the IoT base station over a Wi-Fi connection. In addition to soil sensors, FarmBeats supports cameras for farm monitoring and drones. The cameras are either connected to the IoT base station over Ethernet or report

data over Wi-Fi. They take periodic snapshots and transmit this data to the IoT base station. UAV flights are either periodically scheduled or manually initiated using the FarmBeats app on the farmer's phone.

**IoT Base Station:** The IoT base station on the farm is powered by solar panels, backed by batteries and has three components:

- The TVWS device ensures that the base station on the farm can send the data to the gateway, which then, sends it up to the cloud.
- The sensor connectivity module establishes a connection between the base station and the sensors deployed on the farm. In FarmBeats's current implementation, this module is just a Wi-Fi router.
- Finally, the Base Station Controller is responsible for two functions. First, it serves as a cache for the sensor data collected by the sensor module and syncs this data with the IoT gateway when the TVWS device is switched on. Second, it plans and enforces the duty cycle rates depending on the current battery status and weather conditions.

**IoT Gateway:** As mentioned before, the goal of the IoT gateway is to enable local services and create summaries from existing data to be sent to the cloud. We use a PC form factor device as the FarmBeats gateway, which is typically placed in the farmer's house or office, whichever has Internet access. The gateway provides an interface for applications to run and create summaries to be sent to the cloud as well as to post data to the local web server. Furthermore, it includes a web service for the farmer to access detailed data when they are on the farm network. This also ensures that FarmBeats remains available even when the cloud connection is not present. Finally, it includes built-in algorithms for drone path planning and for compressing drone data before being sent to the cloud (described in Section 3.4). We illustrate in Section 3.4.3 how applications function on the gateway with the example of precision agriculture applications.

Three aspects of the FarmBeats gateway differentiate it from prior IoT gateways. First, the FarmBeats gateway implements a web service, providing unique services that are different from the FarmBeats web service in the cloud. Second, the gateway can operate offline, and still offer the most important services. Finally, as shown later in the context of precision agriculture, having access to data from multiple types of sensors enables unique feature-based summarization technologies for the drone videos and sensor data.

**Services & the Cloud:** The Gateway ships data summaries to the cloud, which provides a storage system for long-term data and a web interface for the farmer. The cloud enables three functions: data access

outside the farm network (e.g. when traveling), long term applications like crop suggestions, and cross-farm analytics.

### 3.3 Duty Cycling the Base Station

As discussed before, FarmBeats’s solar-powered IoT base station on the farm is duty cycled to explicitly account for weather forecasts and current charge state of the batteries. Two aspects of the base station make this problem challenging: a) The sensor connectivity module has significantly lower power requirements than the TVWS device. Thus, we need to intelligently proportion power between these components to achieve optimum performance. b) FarmBeats allows farmers to manually turn the base station on to connect to the Internet to use productivity apps on their phone. This adds a variable component to power consumption.

#### 3.3.1 Duty Cycling Goals

The key goals for the duty cycling algorithm are:

- **Energy Neutrality:** Like past work in the context of duty-cycling sensors backed by energy harvesting sensor systems [94, 226], FarmBeats aims to achieve the objective of energy neutrality. For a given planning period, the goal is to consume at max as much power as can be harvested from the solar panels.
- **Variable Access:** FarmBeats allows farmers to access Wi-Fi connectivity on-demand. This power consumption is usage-driven and varies across days. FarmBeats must plan ahead for this variable delay.
- **Minimize Data Gaps:** We use the term ‘Data Gaps’ to denote continuous time-intervals with no sensor measurements available. Such gaps need to be minimized to avoid missing out on interesting data trends. So, FarmBeats’s duty cycling algorithm aims to minimize the length of the largest data gaps, under the constraints of energy neutrality and variable access.

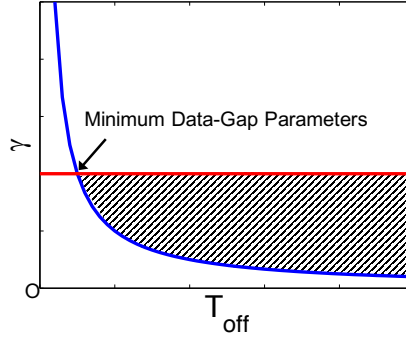


Figure 3-2: **Duty Cycling Approach:** The shaded region shows the feasibility region of Equations 3.2 and 3.4. The latency is minimized when both the equations are satisfied on the boundaries.

### 3.3.2 Power Budget

The sole power source for the base station is a set of solar panels (backed by a battery). The solar power output varies with the time of day and the weather conditions. We use standard methods [200] to estimate the output of the solar panels, given the weather conditions. Let us say that the energy output from the solar panels over the next planning period is  $S_I$ . Because the estimation is not perfect and there is usage variability, there maybe some credit or debit from the previous planning period. Let us denote this credit by  $C_I$ . So, the total power budget for the base station over the next planning period is  $S_I + C_I$ .

### 3.3.3 Duty Cycling Approach

The duty cycle decisions are made on the order of a planning period,  $T_p$ . Since our deployments use solar powered base stations, we set  $T_p$  to be one day. We define the average energy loss due to battery leakage and the very low power base station controller during one  $T_p$  to be  $E_D$ . For the farmer to have on-demand Wi-Fi access, we allocate a fixed time budget of  $T_v$ . If we denote the power consumption of the TVWS device by  $P_T$  and the power consumption of the sensor connectivity module by  $P_S$ , then, we need to allocate  $T_v(P_T + P_S)$  for variable Wi-Fi access. Now the key question is, how do we proportion the remaining power budget?

**Duty Cycling the TVWS device:** The TVWS module is needed to sync the data in the base station cache with the gateway. Let us assume that we have a schedule,  $S$ , the set of sync times advised for the base-station to sync with the FarmBeats gateway. This could depend on the farmer’s usage patterns, sensor types and can be either manually programmed or automatically inferred. The sync times in the set  $S$  have

a corresponding set of weights given by set  $W$ . An example of a high-weighted sync time could be sunrise, as that is when the farmer begins their day. Thus, they would like to access the latest sensor data when the activities of the day are planned.

To ascertain the subset of syncs that need to be performed, we make a simple observation. If the sensors haven't sent any data to the base station, the base station need not turn on the TVWS device. Specifically, it uses the following greedy algorithm to identify the syncs to be executed. Let us denote by,  $S_1 \subset S$ , the subset of syncs that are to be executed. This subset is initialized as an empty set. FarmBeats starts by adding the highest priority sync to  $S_1$ . After it has done that, it subtracts  $|S_1|P_T T_S$  from the power budget, where  $|\cdot|$  denotes set cardinality and  $T_S$  denotes the time to perform a sync operation. Then, FarmBeats computes the corresponding duty-cycle rate for the sensor connectivity module. If this rate ensures that the second highest weighted sync in  $S$  will have additional data from the sensors to sync with the gateway, it adds this sync operation to the set  $S_1$ . It repeats this process in decreasing order of weights until it reaches a state where one of the syncs in  $S_1$  has no new data to share. As we add more sync operations to  $S_1$ , the power budget for the sensor connectivity modules decreases. With a lower power budget, the sensor connectivity module can collect data from the sensors less often and hence it becomes less likely for frequent syncs to see new data. Thus, the algorithm implicitly regulates the sync operations between the gateway and the base station.

**Duty Cycling the Sensor Connectivity Module:** We denote the duty cycling rate for the sensor connectivity module by  $\gamma$ . In particular, it is turned off for a time period,  $T_{off}$ , followed by an on period of,  $T_{on}$  and  $\gamma = T_{on}/T_{off}$ . Using the notation we have established so far, the energy expenditure of the system is  $E_D + (P_S + P_T)T_v + P_T T_S |S_1| + P_S T_p \gamma$ . Since the goal of the planning algorithm is to estimate  $T_{on}$  and  $T_{off}$  such that the energy expenditure does not exceed the energy budget during the planning period, this imposes the following constraint:

$$S_I + C_I \geq E_D + (P_S + P_T)T_v + P_T T_S |S_1| + P_S T_p \gamma \quad (3.1)$$

$$\implies \gamma \leq \frac{S_I + C_I - E_D - (P_S + P_T)T_v - P_T T_S |S_1|}{P_S T_p} \quad (3.2)$$

Let us denote  $T_{connect}$  as the time taken for the sensor connectivity module to turn on and establish a connection to the sensors. Further, let  $T_{sensor}$  be the time that it takes for all the sensors to wake up and transmit to the base station. Since the ON time of the module has to be long enough for the sensors to be

able to communicate their data to the base station, this imposes a further constraint:

$$T_{ON} \geq T_{connect} + T_{transfer} \quad (3.3)$$

$$\implies \gamma T_{off} \geq T_{connect} + T_{transfer} \quad (3.4)$$

Since our goal is to minimize the data gap under the power constraints imposed by Equations 3.2 and 3.4, we aim to minimize  $T_{off}$ . The inequalities from Equations 3.2 and 3.4 define a convex region in the 2-dimensional space of  $(\gamma, T_{off})$ , shown as the shaded region in Figure 3-2. Since the cost function  $T_{off}$  is linear, the minimum occurs on a corner of the intersection region defined by the two inequalities. Specifically, the minimum latency is achieved when the two inequalities are exactly met. The solution is shown graphically in Figure 3-2.

### 3.3.4 Discussion

At this point, it is worth noting that:

- By explicitly accounting for the credit term,  $C_I$ , the formulation absorbs the variability in on-demand Wi-Fi usage patterns. If the on-demand Wi-Fi usage patterns are stable, the term  $C_I$  goes down to zero.
- By incorporating flexibility in sync times between the gateway and the base station, FarmBeats can easily adapt to farm applications with different requirements.
- We have not yet discussed the duty cycling of sensor nodes. In our implementation, we set the duty cycle off time for sensors to be less than  $T_{transfer}$  to ensure that the sensor can transfer data when the sensor connectivity module is on. An alternative implementation would allow the base station to send wake-up times to sensors. Our design choice was motivated by the availability of very low-power sensors that consume 3-4 orders of magnitude less power than the base station on average.

## 3.4 The FarmBeats Gateway

In this section, we discuss three key components of the FarmBeats gateway: UAV path planning, stitched imagery (orthomosaic) generation from UAV videos, and precision map inference from sparse sensor data.

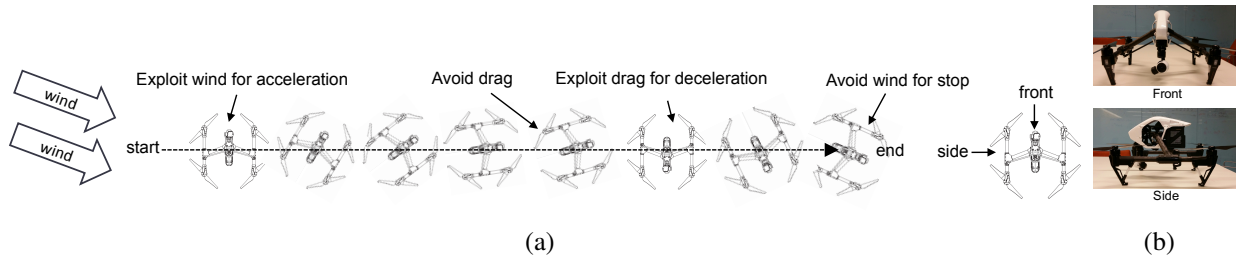


Figure 3-3: FarmBeats’s path planning algorithm uses the asymmetry in front and side profiles of a drone like DJI Inspire 1 (in (b)) to leverage wind to its advantage

### 3.4.1 UAV Path Planning

Most UAVs operate in line sweep patterns. Specifically, given a sequence of waypoints defined by their GPS coordinates, they move from one waypoint to the next, in order. However, in the context of agriculture, our objective is to optimize for the area covered in a single flight. Thus, we aim to minimize the time taken to cover a given area. To that end, we make the observation that increasing the number of waypoints to cover the same area increases the time taken to cover it, even though the total path length may be the same. This is because the quadrotor has to decelerate at each waypoint and come to a halt before it can turn around and accelerate again. We present a novel flight planning algorithm that minimizes the number of waypoints required to cover a given part of the farm.

Existing commercial systems like Pix4D [179], DroneDeploy [63], etc. offer area coverage services, these systems cover a given area using an east-to-west flight path, without any regards to the number of waypoints required. Recent research proposals like [73] do not guarantee the minimum number of waypoints either. Our area coverage algorithm *Min-waypoint* described below guarantees that the UAV covers an area with the minimum number of waypoints:

- Given an area, construct its convex hull.
- Determine the direction of sweeping lines. For each edge and its antipodal (diametrically opposite) vertex [199], draw two parallel lines and measures the distance between them. The slope of the edge corresponding to the minimum distance between the edge and the antipodal vertex becomes the direction of the sweeping lines.
- Determine the waypoints depending on the flight altitude, the camera’s field of view, and desired image quality.



- Given a start-point and end-point of the flight path, order the waypoints to minimize the total travel distance.

**Adaptive Wind-assisted Yaw Control:** Since farms are large open spaces and typically very windy, we observed that quadrotors that have an asymmetric physical profile can exploit the wind either for more efficient propulsion or deceleration. Figure 3-3(b) shows an example of a quadrotor (DJI Inspire 1) that has an asymmetrical profile, where its front and the side are considerably different; thus, it can exploit the wind similar to sailboats. Intuitively, when the quadrotor is flying downwind (i.e. wind is helping the quadrotor), the side profile of the quadrotor should face the wind since the side profile has a larger area and hence, will be able to extract the maximum assistance from the wind. In our experiments on the farm, the quadrotor requires significantly more energy (80% higher at 4m/s) to maintain its speed upwind in comparison to the downwind flight.

To leverage this observation, we designed a novel yaw control algorithm to exploit the wind energy on the farm. Specifically, yaw is the angle of the quadrotor with respect to the vertical axis. While we don't describe the algorithm in detail, on a high level, Figure 3-3 describes how the yaw control algorithm would operate for a quadrotor that has a larger area on the sideways profile. For the downwind segment from the start point to the first waypoint, the adaptive control starts by making the yaw perpendicular to the flight path, thereby maximally utilizing the favorable wind as the quadrotor accelerates. However, as the velocity increases, the air drag generated by the quadrotors profile also increases. Consequently, once the quadrotor accelerates the yaw is reduced so as to maximally exploit the wind, while minimizing the parasitic drag due to the side profile. Similarly, the deceleration phase can very effectively exploit the air drag by making its yaw perpendicular to the flight path. This action is analogous to the action that a skier takes to stop.

### 3.4.2 Generating Orthomosaics from UAV Videos

UAVs generate a prohibitive amount of video that is difficult to transfer to the cloud due to poor network connectivity on farms. For example, a 4 minute flight with a UAV capturing 1080p video at 30 frames per second generates almost a Gigabyte of video data. We make the observation that the unit of interest for the farmer is not the drone video itself, but an overview of the farm that can be provided by a geo-referenced panoramic overview, which is one-two order of magnitude more compact than the full resolution video

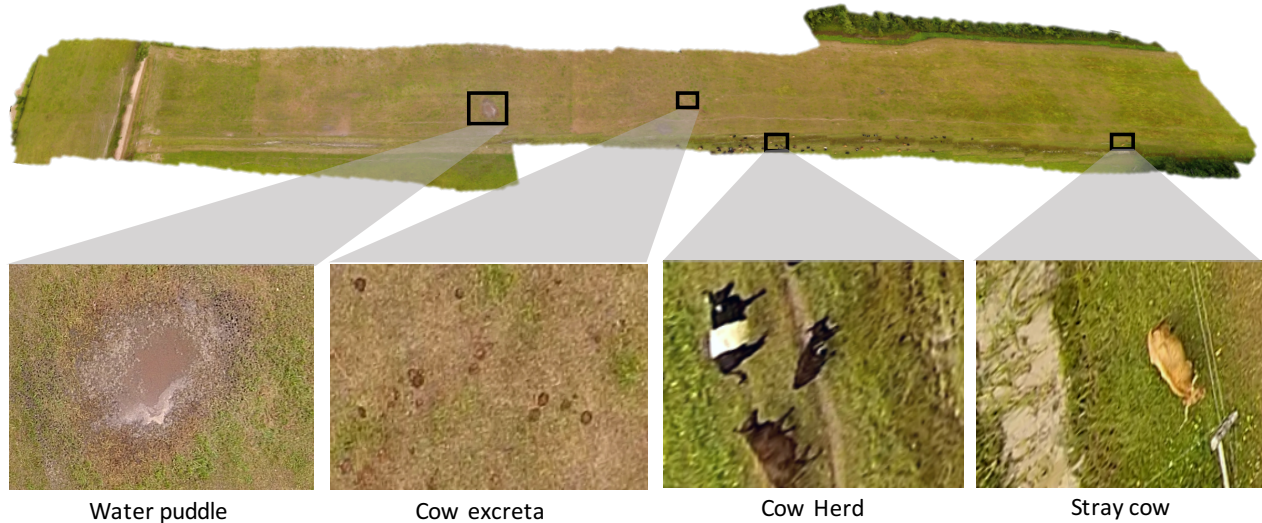


Figure 3-4: **Orthomosaic Generation:** The high resolution orthomosaic generated by FarmBeats for a 5 acre patch in the large farm reveals important visual details to the farmer, such as those shown in the insets – puddles that can make part of the land unavailable for agriculture, cow excreta that becomes manure and enriches the soil, location of individual cows grazing on the farm and their distance from the nearby electric fence.

(see Figure 3-4). The stitched orthomosaic generated from the drone video provides a high resolution visual summary of the farm from a low altitude vantage point, revealing minute details. In fact, existing agricultural drone solutions ([51, 197]) ship the videos to the cloud and convert them into orthomosaics to show to the farmer. Thus, we incorporate the orthomosaics processing pipeline into the FarmBeats Gateway, to process the drone videos locally.

Broadly speaking, the panoramic views can be constructed from the UAV video using two approaches, based on either (i) aerial 3D mapping [179, 196] or (ii) image stitching and mosaicking [159, 21, 36, 213, 205]. While the aerial 3D mapping is a general-purpose method to reconstruct high resolution 3D surface maps of the environment from aerial videos, the image stitching methods treat the world as planar and simply stitch the different images together by finding their relative positions.

Computing high-resolution surface maps is both compute and memory intensive and is not suitable for the resource-constrained farm gateway. On the other hand, while image stitching methods can be incorporated into the gateway, the planar terrain assumption becomes invalid on the farm. Uneven ground geometry, trees, animals or man-made structures observed in the video generates parallax which cannot be handled by the image registration algorithms that assume a planar scene. As we show later in Section 3.6 and as observed in prior work [140], existing image stitchers – Microsoft ICE [159], AutoPano [21] tend

to produce distorted orthomosaics in such scenarios. This presents us with an uncomfortable tradeoff: either fly high such that the farm appears planar and sacrifice fine details of the farm, or ship the large aerial videos to the cloud for processing.

**Our approach:** In order to break this tradeoff, we have developed a hybrid technique which combines key components from both 3D mapping and image stitching methods. On a high level, we use techniques from the aerial 3D mapping systems, just to estimate the relative position of different video frames; without computing the expensive high resolution digital surface maps. Since this process can be performed at a much lower resolution, this allows us to get rid of the harsh compute and memory requirements, while removing the inaccuracies due to non-planar nature of the farm. Once these relative positions have been computed, we can then use standard stitching software (like Microsoft ICE) to stitch together these images. The performance achievements of this hybrid approach are evaluated further in Section 3.6.

### 3.4.3 Generating Precision Maps

As discussed before, precision agriculture relies on accurate precision maps of the farm that indicate the distribution of a specific characteristic throughout the farm. The FarmBeats gateway naturally enables a novel approach to precision map generation that can use the aerial imagery from drones to perform spatial inference of sensor values from sparsely deployed sensors.

Specifically, FarmBeats uses the orthomosaic generated from the drone videos together with the sensor values observed by the sensors planted in the soil, and generates predictions for the entire farm. For example, sensors that observed soil temperature at the discrete locations can inform the machine learning pipeline to make predictions about every location in the farm by considering spatial proximity as well visual similarity of the locations to the sites with the sensors.

FarmBeats's gateway embeds a machine learning pipeline that draws on probabilistic graphical models that embed Gaussian processes [186]. The key intuition in the proposed model is *spatial and visual smoothness*: areas that are similar should observe similar sensor readings. Specifically, the model relies on two kinds of similarities:

- **Visual Smoothness:** Areas that look similar have similar sensor values. For example, a recently irrigated area would look darker and hence, has more moisture.

- **Spatial Smoothness:** Since we are measuring physical properties of the soil and the environment, the sensor readings for locations that are nearby should be similar.

We encode these two intuitions into a graphical model using standard techniques and formulate it as a Gaussian process regression model [186]. Let us denote the farm by a set of  $N$  discrete locations, where the visual features of a location  $i$  are denoted by  $n$ -dimensional vector,  $\mathbf{x}_i$ . Further, let us denote the sensor value (like temperature, pH, etc.) corresponding to the location  $i$  as  $y_i$ , where  $y_i$  could either be directly observed through a sensor measurement or needs to be inferred by FarmBeats. Then, we use  $\mathbf{w}$ , a  $n \times 1$  vector as a linear transformation from visual features to the outputs. Now, our intuition about visual smoothness described above can be incorporated by defining the corresponding loss function as  $\|\mathbf{w}^T \mathbf{x}_i - y_i\|^2$ .

Finally, we use  $\mathbf{D}$  to denote the  $N \times N$  matrix such that  $\mathbf{D}_{ij} = e^{-\frac{\|p_i - p_j\|^2}{2\alpha^2}}$ , where  $p_i$  is the two-dimensional position coordinates for location  $i$ . Thus, in order to incorporate the spatial smoothness constraint, we add another term  $\sum_{i,j} D_{ij}(y_i - y_j)^2$  to our loss function. Thus, the joint loss function of the system can be written as a weighted sum:

$$\begin{aligned} \mathcal{L}(\{\mathbf{x}_i\}_{i=1}^N, \{y_i\}_{i=1}^N, \mathbf{W}, \mathbf{K}; \alpha, \beta) = \\ \|\mathbf{W}\|^2 + \alpha \sum_i \|\mathbf{w}^T \mathbf{x}_i - y_i\|^2 + \beta \sum_{i,j} D_{ij}(y_i - y_j)^2 \end{aligned} \quad (3.5)$$

where  $\|\mathbf{W}\|^2$  is added as a regularization term and  $\alpha$  and  $\beta$  are weight parameters that can be determined at training time. Training and inference on this model are pretty straightforward and follows standard Gaussian process procedure [186]. In the training phase, the value of  $\mathbf{w}$  is learned from the training data (sparse sensor data) and can then be used to predict the sensor output  $y_i$  across all points on the farm, thus generating a precision map.

In our current design, FarmBeats uses the precision maps as units of summarization for the UAV data and ships them to the cloud. This has two advantages over the using orthomosaics as the unit of summary. First, they can perform dense spatial inference from sparse sensor data. Second, they can be compressed to two to three orders of magnitude smaller size than a orthomosaic. So, while the orthomosaic is good for giving the farmer a detailed overview of the farm, precision maps are better for long term storage and shipping. We envision that for other machine learning applications as well, feature maps like the precision maps of the field would be the summaries that get shipped to the cloud, while the descriptive data delivers

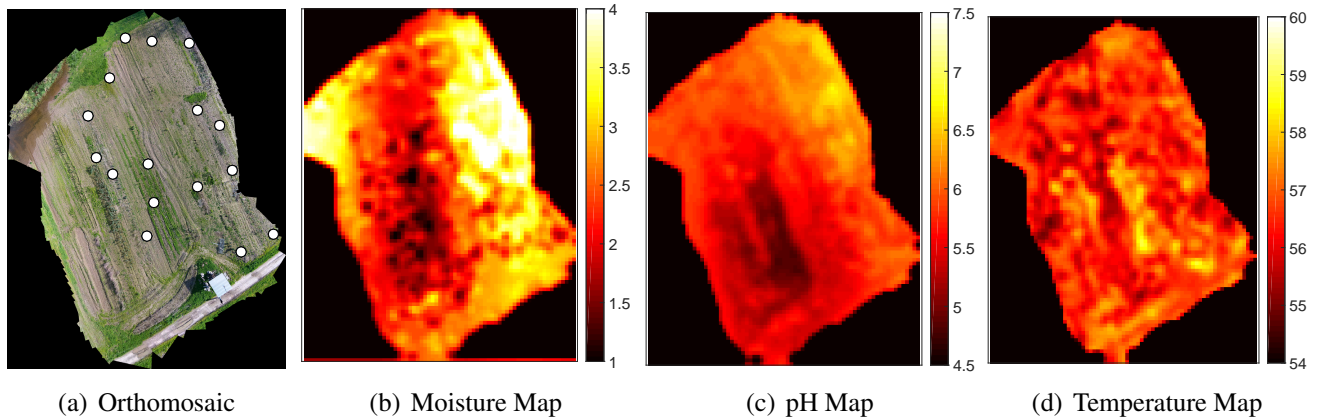


Figure 3-5: **Precision Maps:** (a) A 40 MPixel orthomosaic created from a 3 minute flight over 2 acre area of a farm. Our system infers dense sensor measurements from very few sensors deployed on the farm (indicated by white circles). (b) The predicted soil moisture map (our sensors measures moisture on a scale of 1 to 5). Note that the top left region in the image where the ground appears wet was correctly predicted to have high moisture even though no moisture sensors were present in that part of the farm. (c) The predicted pH map (pH is measured from 0-14, 7 is neutral and 0 is the most acidic). Our system identified that the whole field is slightly acidic, but the bottom left/center is more acidic than the rest. (d) The predicted soil temperature map (in Fahrenheit scale).

short-term applications on the gateway.

### 3.5 Deployment

We deployed FarmBeats in two farms located in Washington (WA) state and in upstate New York (NY), with an area of 5 acres and 100 acres, respectively. The farmer in WA grows vegetables that he sells in the local farmers market. The farm in upstate NY follows the community supported agriculture (CSA) model, and grows vegetables, fruits, grains, as well as dairy, poultry, and meat. Our deployments consist of: sensors, cameras, UAV, the IoT base station, a gateway PC, the cloud service and a dashboard (mobile app and a web page).

**Sensors:** Each farm was equipped with sensors that measure soil temperature, pH, and moisture. In case of sensors without Wi-Fi support, we interfaced them with Arduinos, Particle Photons or NodeMCUs to add Wi-Fi capability. While the exact number of sensors varied over the deployments and the application of interest, we have deployed over 100 different sensors. Additionally, We deployed Microseven IP [156] cameras in different parts of the field to monitor the farm, as well as to capture IR images of crops. To avoid potential damage from environmental impacts, each sensing platform was encased in a weatherproof

box. An example of a sensor deployment can be seen in Figure 3-6(a).

**Drones:** We used the DJI Phantom 2, Phantom 3 and Inspire 1 for our drone flights.<sup>3</sup> We created an auto-pilot application using the DJI Mobile SDK [58] to interface with FarmBeats. The user can use the app to first select the flight altitude and determine the area to be covered on an interactive map. FarmBeats’s app then plans a flight path using the algorithm proposed in Section 3.4.1. After the drone completes its mission, it automatically returns to its home position and transfers the video recording during the flight to the gateway, through the IoT base station.

**IoT Base Station:** At each IoT base station deployment, we set up a TVWS network using the FCC certified Adaptrum ACRS 2 radios [12] operating at 20 dBm, and 11 dBi directional antennas with 90 degree sectors. The internet connectivity was provided by the home internet connection of the farmers. To power the base station we setup a solar charging system, which comprised of two 60 Watt solar panels connected to a solar charge controller. The powering system is backed by four 12V-44Ah batteries connected in parallel. The power output goes through an 8-port Digital Logger PoE switch[145]. This provides us the capability to turn on or off individual components of the base station. A Raspberry Pi 3 with 64 GB SD card serves as the base station controller. The sensors interfaced with the base station through a 802.11b router, with a range of over 100 m.

**Gateway:** The gateway is a Lenovo Thinkpad in the WA farm and a Dell Inspiron laptop in the upstate NY farm.

**Cloud:** We use the Azure IoT Suite ([157]) for FarmBeats. The sensor readings, camera images, and drone video summaries are populated through the Azure IoT Hub ([158]), to storage. We use blobs for images, and tables for the sensor readings. Although in our current implementation, the different farms share the Azure account, with table-level access control, we plan to have different cloud service accounts for the different farms, as FarmBeats scales up.

## 3.6 Results

We evaluate the components of FarmBeats below:

---

<sup>3</sup>We received an exemption from the FAA to fly the UAV.

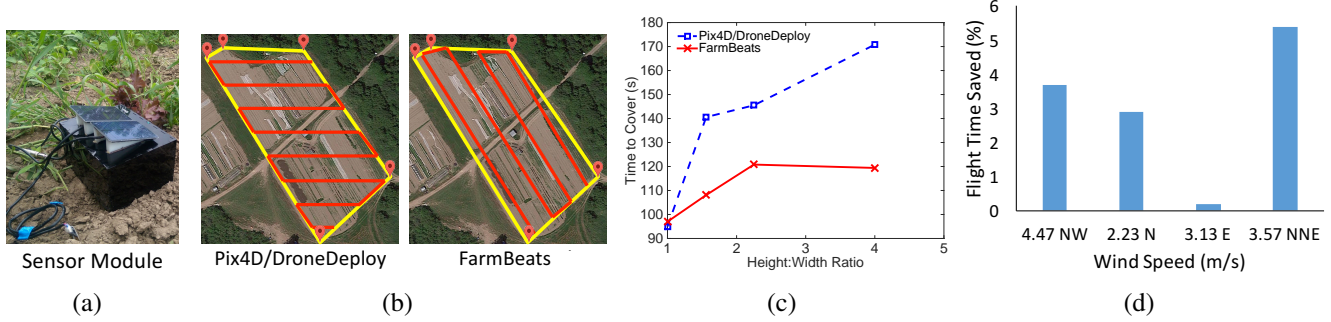


Figure 3-6: (a) A weather-resistant, solar-powered FarmBeats sensor module.(b,c,d) Drone Flight Planning: (b) FarmBeats’s flight planning algorithm minimizes the number of waypoints to cover a region. (c) Depending on the aspect ratio of the field, flights without FarmBeats’s algorithm take upto 42% more time. This improves the time by a factor of 1.26 in the average case for our farms. (d) In addition, the yaw control algorithm described in Section 3.4.1 achieves a gain of up to 5% based on the wind velocity.

### 3.6.1 Weather Aware Base Station

The FarmBeats base station leverages the algorithm in Section 3.3 to duty cycle different components. It uses the OpenWeather API[171] to get the weather forecasts and plans the duty cycling scheme for the next day. The weather information gives us the cloudiness percentage for each period of three hours. The cloudiness percentage over three days is plotted in Figure 3-7(a).

Over this set of three days, we compare three power-awareness schemes. We define the start of the day as 6AM local time. We periodically record the state-of-charge of our solar power backed batteries. First, we let the base station be always on. As shown in Figure 3-7(b), the battery charge goes up during a sunny day and down during the night. While the base station remains energy neutral during the first day, during subsequent days its battery drains because of cloudy weather, leading to unavailability on the third day. Then, we evaluate the alternate approach. We set the base station to a conservative duty cycling period. While this ensures that the base station is available on cloudy days, the base station battery charges up to 100% during the sunny days thus wasting solar power that could have been utilized. Moreover, its duty cycling interval collects *15 times* less data than the optimal FarmBeats solution, plotted in 3-7(d).

FarmBeats collects data on the first two days more frequently owing to high availability of solar power. However, on the third day, it switches to a conservative duty cycling schedule to save power. Of the 15x gain in data collection frequency achieved over a fixed duty cycle, a factor of 2 is because of the TVWS client being duty cycled at a different rate than the Wi-Fi router. An earlier version of our deployment which did not duty cycle the base station faced a downtime of 30% in a cloudy month as opposed to

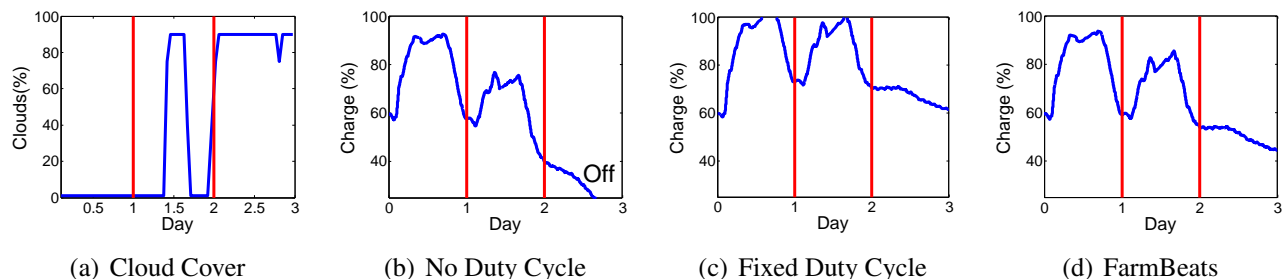


Figure 3-7: **Power-aware Base Station:** The cloudiness percentage over 3 days. (b) With no duty-cycling, the base station shuts down on a cloudy day. (b) A fixed conservative duty cycle can prevent the base station from going down, but it collects 15 times less sensor data. (d) FarmBeats’s Power-aware basestation can keep the base station on by reducing the duty-cycling on days are expected to be cloudy.

zero downtime for our power-aware design in the same month. Thus, FarmBeats’s power-aware design achieves its goal of maximizing data-freshness while maintaining energy neutrality.

### 3.6.2 UAV Flight Planning

As mentioned in Section 3.4.1, we use an efficient area coverage algorithm in addition to leveraging wind assistance to extend drone battery life. To understand the impact of area coverage algorithms on drone flight time, we compare performance of FarmBeats in covering a given area as compared to the state-of-the-art *East-to-west* algorithm (used by Pix4D, DroneMapper, etc.). As shown in Figure 3-6(b), the east-to-west algorithm generates sweeping patterns from the east to the west or vice-versa regardless of the area shape. However, FarmBeats generates a path that minimizes the number of waypoints.

Next, we compare the time taken to complete flights planned by the two algorithms to cover a given area. The maximum speed was set to 10m/s and the altitude was set to 20m. Figure 3-6(c) plots the time taken to complete a flight with the two algorithms in different area geometries defined by their height to width ratio, where height is the distance along the North-South direction and width is measured along East-West. As expected, the gain achieved by FarmBeats increases as the height-width ratio increases. This is because FarmBeats algorithm generates fewer waypoints to cover the same area. In general, for the average case of our deployments, FarmBeats reduced the time taken to cover an area by 26%.

Finally, we evaluate the impact of our yaw control algorithm under different wind conditions. The maximum speed was set to 10m/s and the altitude was set to 30m. For every flight, we fully charged the battery. We measure the percentage of time saved by FarmBeats’s yaw control algorithm for each flight





Figure 3-8: **Orthomosaic Generation:** (a) The Google Earth image for the farm in Figure 3-4. (b) Microsoft ICE image stitching pipeline fails to reconstruct it accurately. (c) Pix4D takes about 2.2x longer on average compared to our approach.

and plot it in Figure 3-6(d). As seen in the figure, FarmBeats can save up to 5% time depending on the wind velocity. Moreover, as the north-south component (the principal direction of motion for this set of experiments) of the wind increases, FarmBeats can leverage it better.

### 3.6.3 Orthomosaic Generation

The novel orthomosaic generation algorithm proposed in this chapter advances the state-of-the-art on two fronts. First, our approach of combining sparse 3D reconstruction techniques from video with image stitching techniques is more robust than existing techniques based on either aerial 3D mapping or aerial image stitching. In addition, our approach is computationally more efficient and runs considerably faster than Pix4D [179], an aerial 3D mapping-based tool catering to Precision Agriculture.

**Qualitative Results:** We show two representative orthomosaics constructed by FarmBeats and Microsoft ICE in Figure 3-4 and 3-8(b) respectively. Figure 3-8(a) shows what the farm looked like in Google Earth in the past. The orthomosaic generated by Microsoft ICE failed in this case, while our result is consistent and accurate. Our geo-referenced image covers about 5 acres of farmland and provides a detailed visual summary to the farmer. By visually inspecting the high-resolution image, they can discover anomalies such as the water puddle that can render a part of the field unsuitable for agriculture for a couple of seasons. Moreover, the farmer can see where cows are grazing during the day and make a decision about whether they want to move them to another spot for the next day. The decision is based on how much grass they want to leave on the field to be converted into manure.

**Processing Time:** As shown in Figure 3-8(c), our implementation is 2.2 times faster than Pix4D on average. Specifically, our method took 14 minutes to construct an orthomosaic on average whereas Pix4D

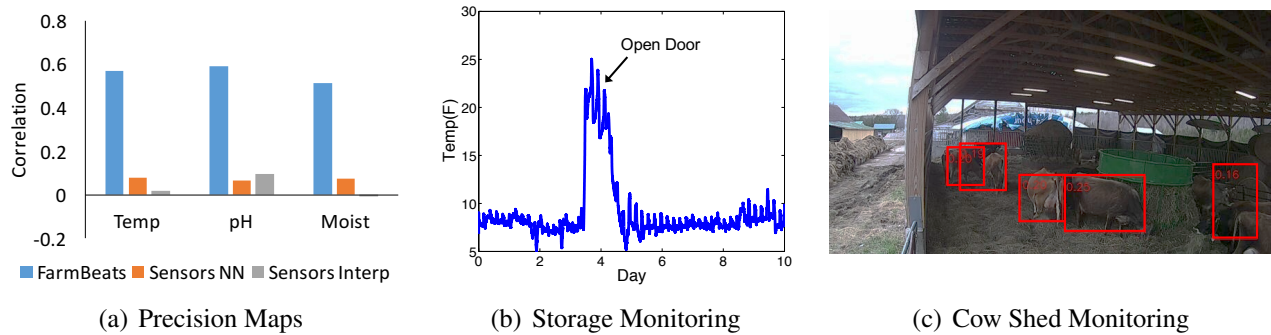


Figure 3-9: **FarmBeats Applications:** (a) FarmBeats’s precision maps are more accurate than standard sensor based interpolation techniques. (b) Temperature (measured in F) in a storage unit can raise an alarm when an employee leaves a door open. (c) Cows being monitored in a cow shed. The red boxes indicate a standard cow detector output.

took 32 minutes on average on a set of videos captured by our drones at 1080p resolution at 30 frames per second. This demonstrates the improved running time of our method.

Finally, the orthomosaic generated by our system are approximately 5 times smaller than the original video size at full resolution (in .png format) before applying lossy compression. A single pixel in the geo-referenced orthomosaic is about 2 cm in size which is equivalent to a single penny on the ground. The image resolution and compression quality are parameters that can be tuned to meet any target file size.

### 3.6.4 Generating Precision Maps

As described in Section 3.4.3, FarmBeats uses the visual features from the orthomosaic overview to extrapolate the sensor values and generate precision maps for soil temperature, soil moisture and pH.

**Qualitative Evaluation:** We show a representative set of these precision maps in Figure 3-5. As shown in the figure, based on sensor values in the rest of the farm, the moisture prediction pipeline can estimate that the top left part of the farm has high moisture content even though that part has no sensor there. Similarly, the pH map generates an actionable input in the sense that the bottom left and center of the farm have very low pH and are highly acidic. As a result of this map, the farmer applied lime to enhance the pH and make the soil more neutral.

Note that the pH of the farm varies within the farm at fine granularity. As seen in Figure 3-5, within a couple of acres, the pH can vary from 4 (very acidic) to 7 (neutral). Soil moisture variance is even higher, with variance seen within a few meters. Precision maps generated by FarmBeats capture this variance

accurately, by using the drone videos to extrapolate the sensor data.

**Quantitative Evaluation:** In order to evaluate the accuracy of the precision maps generated by FarmBeats using the approach described in Section 3.4.3, we evaluated our system on 5 datasets constructed from the drone videos and sensor data. Each dataset corresponds to a drone flight over the farm (covering 2 acres) and one set of sensor measurements from the sparse sensor deployment. The hyperparameters are learned by doing 5 fold cross validation. As an accuracy metric, we measure the correlation between the predicted sensor values and the ground truth sensor values to see how well the variations in the field are captured by FarmBeats. We compare against two techniques, which do not use the drone video based extrapolation of the sensor values:

- **Nearest Neighbor (SensorsNN):** We assign the value from the nearest sensor to each point in the field.
- **Inverse Distance based Interpolation (SensorsInterp):** We linearly interpolate known sensor values in the field, by using inverse distance as a weight. This technique has been previously been proposed in the context of precision agriculture [208, 60].

For all the analysis, we use leave-one-out evaluation, i.e., we generate a precision map after leaving one of the sensors out of the training set and evaluate the map on the left out sensor. We repeat this process for all the ground sensors and report the averaged results.

The comparison of correlation across the different schemes is shown in Figure 3-9(a). As shown in the figure, FarmBeats outperforms existing sensor based interpolation techniques. In particular, FarmBeats can accurately estimate the variations of the different sensor values in the field. While sensor based methods do not mirror the variations and hence have nearly zero correlation with the sensor values, FarmBeats's estimates have high positive correlation with the true sensor values, thus indicating the utility of using the drone video in conjunction with the drone estimates.

Note that, the mean error achieved by FarmBeats's algorithm is 0.61 F (temperature), 0.22 (pH), and 0.42 (moisture measured on a scale of 0 to 6). The standard deviations are 0.12, 0.05, and 0.12 respectively. These errors are roughly equal to the least count of our sensors, indicating that FarmBeats can estimate accurate ground truth values even with a sparse deployment (we estimate sensor values at two orders of magnitude higher density than the deployment density). Finally, the precision maps generated by FarmBeats are 3 orders of magnitude smaller in size on average than the video and can be easily shipped

to the cloud during periods of connectivity.

### 3.6.5 Other Applications

Figure 3-9 highlights two other applications that the farmers used FarmBeats for. First, the farmer in NY used FarmBeats sensors to monitor his storage freezers. The temperature in these freezers is carefully regulated below 10° F to prevent produce from going bad. As shown in Figure 3-9(b), an employee leaving the door open could lead to this temperature going up causing loss to the farmer. This problem is solved by FarmBeats by enabling automated notifications based on these sensor readings in the FarmBeats phone application.

Second, the farmers plugged in cameras at different locations like cow sheds and connected them to the nearest FarmBeats base station. One frame of the camera is shown in Figure 3-9(c). While the intent of the current application is to manually monitor the cows, one can potentially build an application that can detect anomalies in cow behavior or use cow motion to track animal health[163]. As a preliminary result, we ran a deep neural network based cow detector on the data. The identification boxes are overlaid on the figure.

### 3.6.6 End-to-end Deployment Statistics

**Data Aggregation:** FarmBeats’s deployments at both farms have been running for over six months. Over these deployments, FarmBeats interfaced with around 10 different sensor types, three different camera types, three versions of drones and the farmers’ phones. It collected more than 10 million sensor measurements, half million images and 100 drone surveys.

**Resilience to Outages:** FarmBeats’s deployments faced one week-long Internet outage due to a thunderstorm and several smaller term Internet outages. The FarmBeats gateway continued to be available during these times.

**Cost:** The TVWS client radios cost \$200,<sup>4</sup> and there are no additional data charges, than the farmer’s existing internet connection. The Particle Photons cost about \$20 and can add Wi-Fi support to each sensor. Thus, use of the hybrid networking approach reduces the system cost by an order of magnitude as

---

<sup>4</sup>With the standardization of IEEE 802.11af [2] standard, we expect the price to the client and base station to be similar to Wi-Fi, of less than 10\$. We are testing one such multi-mode TVWS/Wi-Fi chip from a major Wi-Fi vendor.

compared to existing systems which cost over \$1000 in equipment cost per sensor and over 100\$ annual subscription fee (see Table 3.1).

**Applications:** Farmers used FarmBeats’s precision agriculture system to guide their precision irrigation units. The precision pH maps generated were used by farmers to apply lime in the more acidic regions. As mentioned before, farmers also used FarmBeats for storage monitoring with sensors and animal shelter monitoring, selling station monitoring with cameras. Beyond that, farmers also used FarmBeats base stations to access Wi-Fi while on the farm to run productivity applications like Trello.

### 3.7 Related Work

FarmBeats builds on past work in wireless sensor networks, precision agriculture and ICTD.

**Wireless Sensor Networks:** Past work has used multi-hop networks [75, 234, 136, 190, 26, 113, 169, 237] to gather data from sensors in the farm. However, all these systems suffer from bandwidth constraints that make them unable to support sensors, cameras and drones. Further, these systems do not account for constraints imposed by weak cloud connectivity and weather related power and Internet outages. The same is true for recent advances in LPWAN technologies [146, 203]. In contrast, FarmBeats includes support for sensors, cameras and drones; is backed by cloud connectivity and has mechanisms to adapt to weather variability.

**Agriculture:** Agronomists have studied various aspects of precision agriculture, from defining more accurate management zones [153], to improving prescription [161], to leveraging soil science [207] and plant physiology [40] techniques. Prior work has also looked at applications of precision agriculture to irrigation, variable seeding, nutrient application, and others. There has been prior work on developing technology for enabling precision agriculture. Researchers have built specialized sensors for measuring nutrients [121], water levels [89], and other such sensors, and we build on top of this work. FarmBeats’s work is complementary to this body of work as it facilitates the automation of data collection using these sensors and enables the precision agriculture systems.

**ICTD:** ICTD solutions focus on user interfaces to make existing technologies more accessible[59] enhanced access to information [45] and better communications. The mechanisms of data collection is manual in most scenarios. The few attempts at automated data collection, like [45], fall into the same pitfalls as discussed before. We believe FarmBeats is complimentary to this work and will aid the proliferation of

ICTD by enabling end-to-end IoT connectivity in weakly connected scenarios.

### **3.8 Conclusion & Contributions**

FarmBeats started as a Microsoft Research project in 2015. Over the next two years, we developed FarmBeats to become a low-cost, highly available IoT platform for agriculture. It incorporates TV White Space connectivity for sensors, a weather-aware solar-powered IoT base station, an intelligent gateway that ensures high availability of services, a multi-modal inference mechanism, and a drone path planning algorithm to extend drone battery life. While I contributed to many of these components, my primary responsibilities in the project were the design of the weather-aware base station, the edge gateway to interface the different components, and the multi-modal inference mechanism. In this chapter, we discussed FarmBeats's deployment in two farms in United States. Over the years, FarmBeats has been deployed in many more farms in United States, India, and Africa. Across these deployments, farmers have self-reported increased yields and reduced input costs.

## Chapter 4

# Decimeter-Level Localization with a Single WiFi Access Point

Recent years have seen significant advances in indoor positioning using wireless signals [244, 112]. State-of-the-art systems have achieved an accuracy of tens of centimeters, even using commodity Wi-Fi chipsets [124, 127, 79]. Existing proposals however target enterprise networks, where multiple Wi-Fi access points can combine their information and cooperate together to locate a user. However, the vast majority of homes and small businesses today have a single Wi-Fi access point. Consequently, this large constituency of wireless networks has been left out of the benefits of accurate indoor positioning.

Developing a technology that can locate users and objects using a single Wi-Fi access point would enable a range of important applications:

- (i) *Smart Home Occupancy*: In particular, indoor positioning can play a crucial role in the smart home vision, where Wi-Fi enabled home automation systems like NEST are gaining increasing popularity [165]. Accurate localization addresses a long-standing problem in home automation: reliable occupancy detection [162, 20]. With Wi-Fi based localization, one can track the number of users per room using their phones or wearables, and accordingly adapt heating and lighting. Knowing the identity of these occupants can then help personalize heating and lighting levels based on user preferences.
- (ii) *Wi-Fi Geo-fencing*: Beyond the home, indoor positioning can benefit small businesses that use a single access point to offer free Wi-Fi to attract customers. But with increasingly congested networks, business owners seek to restrict Wi-Fi connectivity to their own customers, given that 32% of users in

the US admit to have accessed open Wi-Fi networks outside the premises they serve [240]. Yet securing these networks with passwords is inconvenient, both to customers that connect to these networks and the business owners who must frequently change the passwords. Indoor positioning with a single access point provides a natural solution to this problem because it can automatically authenticate customers based on their location.

- (iii) *Device-to-device Location*: More generally, enabling two Wi-Fi nodes to localize each other without additional infrastructure support has implications in areas where Wi-Fi networks may not exist altogether. Imagine traveling with friends or family in countries where Wi-Fi is not as prevalent as in the US, yet still be able to find each other in a mall, museum, or train station, without the need to connect to a Wi-Fi infrastructure.

Our goal is to design a system that enables a single Wi-Fi node (e.g., an access point) to localize another, without support from additional infrastructure. Further, we would like a design that works on commodity Wi-Fi NICs and does not require any additional sensors (cameras, accelerometers, etc.).

As we design for the above goal, it helps to first examine why past systems need multiple access points. The most direct approach to RF-based positioning estimates the time-of-flight (i.e., propagation time) and multiplies it by the speed of light to obtain the distance [91, 76]. However, past proposals for Wi-Fi based positioning cannot measure the *absolute* time-of-flight. They measure only *differences* in the time-of-flight across the receiver's antennas. Such time differences allow those systems to infer the direction of the source with respect to the receiver, known as the angle of arrival (AoA) [244]. But they don't provide the distance between the source and the receiver. Thus, past work has to intersect the direction of the source from multiple access points to localize it. In fact, past proposals typically use four or five access points to achieve tens of centimeters accuracy [124, 127, 244, 246]. Even the few recent proposals to localize using one Wi-Fi access point [151, 260] require users to walk to multiple locations to emulate the presence of multiple access points. They then intersect signal measurements across these locations coupled with accelerometer readings to infer the user's trajectory.

There are however non-Wi-Fi systems that can accurately measure the absolute time-of-flight, and hence localize using a single receiver. Such systems use specialized ultra wideband radios that span multiple GHz [14, 198]. Since time resolution is inversely related to the radio bandwidth, such devices can measure time-of-flight at sub-nanosecond accuracy, and hence localize an object to within tens of



centimeters. In contrast, directly measuring time with a 20MHz or 40MHz Wi-Fi radio results in errors of 7 to 15 meters [124].

Motivated by the above analysis, we investigated whether a Wi-Fi radio can emulate a wideband multi-GHz radio, for the purpose of localization. Our investigation led to Chronos, an indoor positioning system that enables a pair of Wi-Fi devices to localize each other. It runs on commodity Wi-Fi cards, and does not require any external sensor (e.g., accelerometer, or camera). Chronos works by making a Wi-Fi card emulate a very wideband radio. In particular, while each Wi-Fi frequency band is only tens of Megahertz wide, there are many such bands that together span a very wide bandwidth. Chronos therefore transmits packets on multiple Wi-Fi bands and stitches their information together to give the illusion of a wideband radio.

Yet, emulating a wideband radio using packets transmitted on different frequency bands is not easy. Stitching measurements across such packets requires Chronos to overcome three challenges:

**Resolving Phase Offsets:** First, to emulate a wideband radio, Chronos needs to stitch channel state information (CSI) captured by multiple packets, transmitted in different Wi-Fi frequency bands, at different points in time. However, the very act of hopping between Wi-Fi frequency bands introduces a random initial phase offset as the hardware resets to each new frequency (i.e., PLL locking). Chronos must therefore recover time-of-flight to perform positioning despite these random phase offsets.

**Eliminating Packet Detection Delay:** Second, any measurement of time-of-flight of a packet necessarily includes the delay in detecting its presence. Different packets however experience different random detection delays. To make matters worse, this packet detection delay is typically orders-of-magnitude higher than time-of-flight. For indoor Wi-Fi environments, time-of-flight is just a few nanoseconds, while packet detection delay spans hundreds of nanoseconds [181]. Chronos must tease apart the time-of-flight from this detection delay.

**Combating Multipath:** Finally, in indoor environments, signals do not experience a single time-of-flight, but a time-of-flight spread. This is because RF signals in indoor environments bounce off walls and furniture, and reach the receiver along multiple paths. As a result, the receiver obtains several copies of the signal, each having experienced a different time-of-flight. To perform accurate localization, Chronos therefore must disentangle the time-of-flight of the direct path from all the remaining paths.

The body of this chapter explains how Chronos overcomes these challenges, computes the absolute

time-of-flight, and enables localization using a single access point.

**Summary of Results:** We have implemented Chronos and evaluated its performance on devices equipped with Intel 5300 Wi-Fi cards. Our results reveal the following:

- Chronos computes the time-of-flight with a median error of 0.47 ns in line-of-sight and 0.69 ns in non-line-of-sight settings. This corresponds to a median distance error of 14.1 cm and 20.7 cm respectively.
- Chronos enables a Wi-Fi device (e.g., an AP) to localize another with a median error of 65 cm in line-of-sight and 98 cm in non-line-of-sight settings.

To demonstrate Chronos’s capabilities, we use it for three applications:

- *Smart Home Occupancy:* Chronos can be used to track the number of occupants in different rooms of a home using a single access point – a key primitive for smart homes that adapt heating and lighting. Experiments conducted in a 2-bedroom apartment with 4 occupants show that Chronos maps residents in a home to the correct room they are in with an accuracy of 94.3%.
- *Wi-Fi Geo-fencing:* Chronos can be used by small businesses with a single access point to restrict Wi-Fi connectivity to customers within their facility. Experiments in a coffee house reveal that Chronos achieves this to an accuracy of 97%.
- *Personal Drone:* Chronos’s ability to locate a pair of user devices can directly benefit the navigation systems of personal robots such as recreational drones. Chronos enables personal drones that can maintain a safe distance from their user by tracking their owner’s handheld device. Our experiments using an AscTec Quadrotor reveal that it maintains the required distance relative to a user’s device with a root mean-squared error of 4.2 cm.

**Contributions:** To our knowledge, Chronos is the first system that enables a node with a commercial Wi-Fi card to locate another at tens of centimeters accuracy without any third party support, be it other Wi-Fi nodes or external sensors (e.g., accelerometers). Chronos also contributes the first algorithm for measuring the absolute time-of-flight on commercial Wi-Fi cards at sub-nanosecond accuracy.

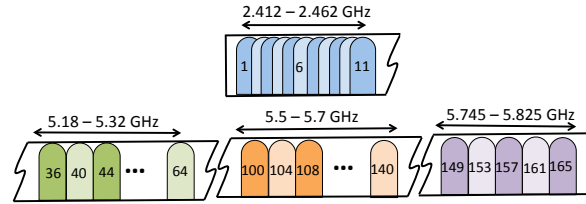


Figure 4-1: **Wi-Fi Bands:** Depicts Wi-Fi bands at 2.4 GHz and 5 GHz. Note that some of these frequencies (e.g. 5.5-5.7 GHz) are DFS bands in the U.S. that many 802.11h compatible 802.11n radios like Intel 5300 support.

## 4.1 Overview

We briefly outline the organization of the rest of this chapter. Chronos localizes a pair of Wi-Fi devices without third party support by computing time of flight of signals between them. Sec. 4.2 describes our approach to compute time-of-flight by stitching together information across multiple Wi-Fi frequency bands. It is followed by a description of the challenges faced by Chronos and how it addresses them. Specifically:

- **Eliminating Packet Detection Delay:** First, Chronos disentangles the time-of-flight from packet detection delay, since the latter has no connection to the distance between transmitter and receiver (See Sec. 4.3).
- **Combating Multipath:** Second, Chronos separates the time-of-flight of the direct path of the wireless signal from that of all the remaining paths (See Sec. 4.4).
- **Resolving Phase Offsets:** Finally, Chronos removes arbitrary phase offsets that are introduced as the Wi-Fi receiver hops between frequency bands (See Sec. 4.5).

## 4.2 Measuring Time of Flight

In this section, we describe how Chronos measures accurate time-of-flight of signals between a pair of Wi-Fi devices without third party support. For clarity, the rest of this section assumes signals propagate from the transmitter to a receiver along a single path with no detection delay or phase offsets. We address challenges stemming from packet detection delay, multipath and phase offsets in 4.3, 4.4 and 4.5 respectively.

Chronos’s approach is based on the following observation: Conceptually, if our receiver had a very wide bandwidth, it could readily measure time-of-flight from a single receiving device at a fine-grained resolution (since time and bandwidth are inversely related). Unfortunately, today’s Wi-Fi devices do not have such wide bandwidth. But there is another opportunity: Wi-Fi devices are known to span multiple frequency bands scattered around 2.4 GHz and 5 GHz. Combined, these bands span almost one GHz of bandwidth. By making a transmitter and receiver hop between these different frequency bands, we can gather many different measurements of the wireless channel. We can then “stitch together” these measurements to compute the time-of-flight, as if we had a very wideband radio.

However, our method for stitching time measurements across Wi-Fi frequency bands must account for the fact that many Wi-Fi bands are non-contiguous, unequally spaced, and even multiple GHz apart (Fig. 4-1). Chronos overcomes these issues by exploiting the relation between the time-of-flight and the phase of wireless channels. Specifically, we know from basic electromagnetics that as a signal propagates in time, it accumulates a corresponding phase depending on its frequency. The higher the frequency of the signal, the faster the phase accumulates. To illustrate, let us consider a transmitter sending a signal to its receiver. Then we can write the wireless channel  $h$  as [216]:

$$h = ae^{-j2\pi f\tau}, \tag{4.1}$$

where  $a$  is the signal magnitude,  $f$  is the frequency and  $\tau$  is the time-of-flight. The phase of this channel depends on time-of-flight as:

$$\angle h = -2\pi f\tau \pmod{2\pi} \tag{4.2}$$

Notice that the above equation depends directly on the signal’s time-of-flight and hence, we can use it to measure the time-of-flight  $\tau$  as:

$$\tau = -\frac{\angle h}{2\pi f} \pmod{\frac{1}{f}} \tag{4.3}$$

The above equation gives us the time-of-flight modulo  $1/f$ . Hence, for a Wi-Fi frequency of 2.4 GHz, we can only obtain the time-of-flight *modulo* 0.4 nanoseconds. Said differently, transmitters with times-of-flight 0.1 ns, 0.5 ns, 0.9 ns, 1.3 ns, etc. all produce identical phase in the wireless channel. In terms of

physical distances, this means transmitters at distances separated by multiples of 12 cm (e.g., 3 cm, 15 cm, 27 cm, 39 cm, etc.) all result in the same channel phase. Consequently, there is no way to distinguish between these transmitters using their phase on a single frequency band.

Indeed, this is precisely why Chronos needs to hop between multiple frequency bands  $\{f_1, \dots, f_n\}$  and measure the corresponding wireless channels  $\{h_1, \dots, h_n\}$ . The result is a system of equations, one per frequency, that measure the time-of-flight modulo different values:

$$\forall i \in \{1, 2, \dots, n\} \quad \tau = -\frac{\angle h_i}{2\pi f_i} \pmod{\frac{1}{f_i}} \quad (4.4)$$

Notice that the above set of equations has the form of the well-known Chinese remainder theorem [235]. Such equations can be readily solved using standard modular arithmetic algorithms, even amidst noise [57] and have been used in prior work, in the context of range estimation ([228, 227]).<sup>1</sup> The theorem states that solutions to these equations are unique modulo a much larger quantity – the Least Common Multiple (LCM) of  $\{1/f_1, \dots, 1/f_n\}$ .

To illustrate how the above system of equations works, consider a source at 0.6 m whose time-of-flight is 2 ns. Say the receiver measures the channel phases from this source on five candidate Wi-Fi frequency bands as shown in Fig. 4-2. We note that a measurement on each of these channels produces a unique equation for  $\tau$ , like in Eqn. 4.4. Each equation has multiple solutions, depicted as colored vertical lines in Fig. 4-2. However, only the correct solution of  $\tau$  will satisfy all equations. Hence, by picking the solution satisfying the most number of equations (i.e., the  $\tau$  with most number of aligned lines in Fig. 4-2), we can recover the true time-of-flight of 2 ns.

Note that our solution based on the Chinese remainder theorem makes no assumptions on whether the set of frequencies  $\{f_1, \dots, f_n\}$  are equally separated or otherwise. In fact, having unequally separated frequencies makes them less likely to share common factors, boosting the LCM. Thus, counter-intuitively, the scattered and unequally-separated bands of Wi-Fi (Fig. 4-1) are not a challenge, but an opportunity to resolve larger values of  $\tau$ .

While the above provides a mathematical formulation of our algorithm, we describe below important systems considerations when dealing with commercial Wi-Fi cards:

- Chronos must ensure both the Wi-Fi transmitter and receiver hop synchronously between multiple

---

<sup>1</sup>Algorithm 1 in 4.4 provides a more general version of Chronos’s algorithm to do this while accounting for noise and multipath

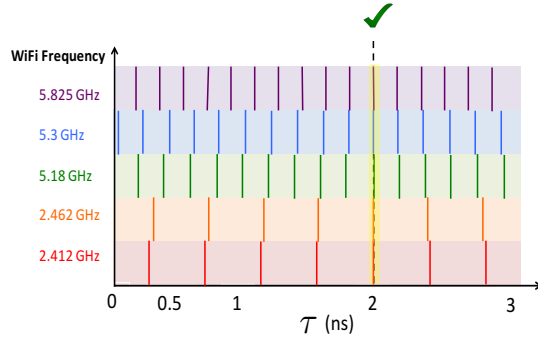


Figure 4-2: **Measuring Time-of-Flight:** Consider a wireless transmitter at a distance of 0.6 m, i.e. a time-of-flight of 2 ns. The phase of each Wi-Fi channel results in multiple solutions, depicted as colored lines, including 2 ns. However, the solution that satisfies most equations, i.e. has the most number of aligned colored lines is the true time-of-flight (2 ns).

Wi-Fi frequency bands. Chronos achieves this using a frequency band hopping protocol driven by the transmitter. Before switching frequency bands (every 2-3 ms in our implementation), the transmitter issues a control packet that advertises the frequency of the next band to hop to. The receiver responds with an acknowledgment and switches to the advertised frequency. Once the acknowledgment is received, the transmitter switches frequency bands as well. As a fail-safe, transmitters and receivers revert to a default frequency band if they do not receive packets or acknowledgments from each other for a given time-out duration on any band.

- Our implementation of Chronos sweeps all Wi-Fi bands in 84 ms (12 times per second). This is within the channel coherence time of indoor environments [182] and can empirically localize users at walking speeds (4.9.3).
- Finally, we discuss and evaluate the implications of Chronos’s protocol on data traffic in 4.8.3.

### 4.3 Eliminating Packet Detection Delay

So far, we computed time-of-flight based on the channels  $h_i$ , that signals experience when transmitted over the air on different frequencies  $f_i$ . In practice however, there is a difference between the channel over the air,  $h_i$ , and the channel as measured by the receiver,  $\tilde{h}_i$ . Specifically, the *measured* channel at the receiver,  $\tilde{h}_i$ , experiences a delay in addition to time-of-flight: the delay in detecting the presence of a packet. This delay occurs because Wi-Fi receivers detect the presence of a packet based on the energy

of its first few time samples. The number of samples that the receiver needs to cross its energy detection threshold varies based on the power of the received signal, as well as noise. While this variation may seem small, packet detection delays are often an *order-of-magnitude* larger than time-of-flight, particularly in indoor environments, where time-of-flight is just a few tens of nanoseconds (See 4.8.1). Hence, accounting for packet detection delay is crucial for accurate time-of-flight and distance measurements.

Thus, our goal is to derive the true channel  $h_i$  (which incorporates the time-of-flight alone) from the measured channel  $\tilde{h}_i$  (which incorporates both time-of-flight and packet detection delay). To do this, we exploit the fact that Wi-Fi uses OFDM. Specifically, the bits of Wi-Fi packets are transmitted in the frequency domain on several small frequency bins called OFDM subcarriers. This means that the wireless channels  $\tilde{h}_i$  can be measured on each subcarrier. We then make the following claim:

**Claim 4.3.1** *The measured channel at subcarrier-0 does not experience packet detection delay, i.e., it is identical in phase to the true channel at subcarrier 0.*

To see why this claim holds, note that while time-of-flight and packet detection delay appear very similar, they occur at different stages of a signal’s lifetime. Specifically, time-of-flight occurs while the signal is transmitted over the air (i.e., in passband). In contrast, packet detection delay stems from energy detection that occurs in digital processing once the carrier frequency has been removed (in baseband). Thus, time-of-flight and packet detection delay affect the wireless OFDM channels in different ways.

To understand this difference, consider the Wi-Fi frequency band,  $i$ . Let  $\tilde{h}_{i,k}$  be the measured channel of OFDM subcarrier  $k$ , at frequency  $f_{i,k}$ .  $\tilde{h}_{i,k}$  experiences two phase rotations in different stages of the signal’s lifetime:

- A phase rotation in the air proportional to the over-the-air frequency  $f_{i,k}$ . From Eqn. 4.2 in 4.2, this phase value for a frequency  $f_{i,k}$  is:

$$\angle h_{i,k} = -2\pi f_{i,k}\tau \pmod{2\pi},$$

where  $\tau$  is the time-of-flight.

- An additional phase rotation due to packet detection after the removal of the carrier frequency. This

additional phase rotation can be expressed as:

$$\Delta_{i,k} = -2\pi(f_{i,k} - f_{i,0})\delta_i,$$

where  $\delta_i$  is the packet detection delay.

Thus, the total measured channel phase at subcarrier  $k$  is:

$$\angle \tilde{h}_{i,k} = (\angle h_{i,k} + \Delta_{i,k}) \pmod{2\pi} \quad (4.5)$$

$$= (-2\pi f_{i,k}\tau - 2\pi(f_{i,k} - f_{i,0})\delta_i) \pmod{2\pi} \quad (4.6)$$

Notice from the above equation that the second term  $\Delta_{i,k} = -2\pi(f_{i,k} - f_{i,0})\delta_i = 0$  at  $k = 0$ . In other words, at the zero-subcarrier of OFDM, the measured channel  $\tilde{h}_{i,k}$  is identical in phase to the true channel  $h_{i,k}$  over-the-air which validates our claim.

In practice, this means that we can apply the Chinese Remainder theorem as described in Eqn. 4.4 of 4.2 at the zero-subcarriers (i.e. center frequencies) of each Wi-Fi frequency band. In the U.S., Wi-Fi at 2.4 GHz and 5 GHz has a total of 35 Wi-Fi bands with independent center frequencies.<sup>2</sup> Therefore, a sweep of all Wi-Fi frequency bands results in 35 independent equations like in Eqn. 4.4, which we can solve to recover time-of-flight.

One problem still needs to be addressed. So far we have used the measured channel at the zero-subcarrier of Wi-Fi bands. However, Wi-Fi transmitters do not send data on the zero-subcarrier, meaning that this channel simply cannot be measured. This is because the zero-subcarrier overlaps with DC offsets in hardware that are extremely difficult to remove [90, 4]. So how can one measure channels on zero-subcarriers if they do not even contain data?

Fortunately, Chronos can tackle this challenge by using the remaining Wi-Fi OFDM subcarriers, where signals are transmitted. Specifically, it leverages the fact that indoor wireless channels are based on physical phenomena. Hence, they are continuous over a small number of OFDM subcarriers [107]. This means that Chronos can interpolate the measured channel phase across all subcarriers to estimate the missing phase at the zero-subcarrier.<sup>3</sup> Indeed, the 802.11n standard [4] measures wireless channels on as many

<sup>2</sup>Including the DFS bands at 5 GHz in the U.S. which are supported by many 802.11h-compatible 802.11n radios, e.g., the Intel 5300.

<sup>3</sup>Our implementation of Chronos uses cubic spline interpolation.



as 30 subcarriers in each Wi-Fi band. Hence, interpolating between the channels not only helps Chronos retrieve the measured channel on the zero-subcarrier, but also provides additional resilience to noise.

To summarize, Chronos applies the following steps to account for packet detection delay: (1) It obtains the measured wireless channels on the 30 subcarriers on the 35 available Wi-Fi bands; (2) It interpolates between these subcarriers to obtain the measured channel phase on the zero-subcarriers on each of these bands, which is unaffected by packet detection delay. (3) It retrieves the time-of-flight using the resulting 35 channels.

## 4.4 Combating Multipath

So far, our discussion has assumed that a wireless signal propagates along a single direct path between its transmitter and receiver. However, indoor environments are rich in multipath, causing wireless signals to bounce off objects in the environment like walls and furniture. Fig. 4-3(a) illustrates an example where the signal travels along three paths from its sender to receiver. The signals on each of these paths propagate over the air incurring different time delays as well as different attenuations. The ultimate received signal is therefore the sum of these multiple signal copies, each having experienced a different propagation delay. Fig. 4-3(b) represents this using a *multipath profile*. This profile has peaks at the propagation delays of signal paths, scaled by their respective attenuations. Hence, Chronos needs a mechanism to find such a multipath profile, so as to separate the propagation delays of different signal paths. This allows it to then identify the time-of-flight as the least of these propagation delays, i.e. the delay of the most direct (shortest) path.

### 4.4.1 Computing Multipath Profiles

Say that wireless signals from a transmitter reach a receiver along  $p$  different paths. The received signal from each path corresponds to amplitudes  $\{a_1, \dots, a_p\}$  and propagation delays  $\{\tau_1, \dots, \tau_p\}$ . Observe that Eqn. 4.1 considers only a single path experiencing propagation delay and attenuation. In the presence of multipath, we can extend this equation to write the measured channel  $\tilde{h}_{i,0}$  on center-frequency  $f_{i,0}$  as the

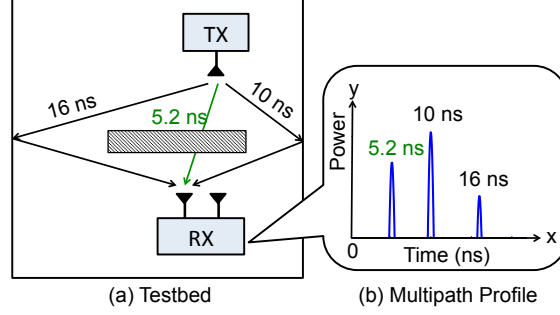


Figure 4-3: **Combating Multipath:** Consider a signal propagating from a transmitter to a receiver along 3 paths as shown in (a): an attenuated direct path and two reflected paths of lengths 5.2 ns, 10 ns and 16 ns respectively. These paths can be separated by using the inverse discrete Fourier Transform as shown in (b). The plot has 3 peaks corresponding to the propagation delays of the paths, with peak magnitudes scaled by relative attenuations.

sum of the channels on each of these paths, i.e.:

$$\tilde{h}_{i,0} = \sum_{k=1}^p a_k e^{-j2\pi f_{i,0} \tau_k} \quad , \text{ for } i = 1, \dots, n \quad (4.7)$$

Now, we need to disentangle these different paths and recover their propagation delays. To do this, notice that the above equation has a familiar form – it is the well-known Discrete Fourier Transform. Thus, if one could obtain the channel measurements at many uniformly-spaced frequencies, a simple inverse-Fourier transform would separate individual paths. Such an inverse Fourier transform has a closed-form expression that can be used to obtain the propagation delay of all paths and compute the multipath profile (up to a resolution defined by the bandwidth).

Wi-Fi frequency bands, however, are not equally spaced – they are scattered around 2.4 GHz and multiple non-contiguous chunks at 5 GHz, as shown in Fig. 4-1. While we can measure  $\tilde{h}_{i,0}$  at each Wi-Fi band, these measurements will not be at equally spaced frequencies and hence cannot be simply used to compute the inverse Fourier transform. In fact, since our measurements of the channels are not uniformly spaced, we are dealing with the *Non-uniform* Discrete Fourier Transform or NDFT [25]. To recover the multipath profile, we need to invert the NDFT.

#### 4.4.2 Inverting the NDFT

The NDFT is an under-determined system, where the responses of multiple frequency elements are unavailable [84, 64]. Thus, the inverse of such a Fourier transform does not have a single closed-form

solution, but several possible solutions. So how can Chronos pick the best among those solutions to find the true times-of-flight?

Chronos adds another constraint to the inverse-NDFT optimization. Specifically, this constraint favors solutions that are sparse, i.e., have few dominant paths. Intuitively, this stems from the fact that while signals in indoor environments traverse several paths, a few paths tend to dominate as they suffer minimal attenuation [29].<sup>4</sup> Indeed other localization systems make this assumption as well, albeit less explicitly. For instance, antenna-array systems can resolve a limited number of dominant paths based on the number of antennas they use.

We can formulate the sparsity constraint mathematically as follows. Let the vector  $\mathbf{p}$  sample inverse-NDFT at  $m$  discrete values  $\tau \in \{\tau_1, \dots, \tau_m\}$ . Then, we can introduce sparsity as a simple constraint in the NDFT inversion problem that minimizes the L-1 norm of  $\mathbf{p}$ . Indeed, it has been well-studied in optimization theory that minimizing the L-1 norm of a vector favors sparse solutions for that vector [24]. Thus, we can write the optimization problem to solve for the inverse-NDFT as:

$$\min \|\mathbf{p}\|_1 \tag{4.8}$$

$$\text{s.t. } \|\tilde{\mathbf{h}} - \mathcal{F}\mathbf{p}\|_2^2 = 0 \tag{4.9}$$

where,  $\mathcal{F}$  is the  $n \times m$  Fourier matrix, i.e.  $\mathcal{F}_{i,k} = e^{-j2\pi f_{i,0}\tau_k}$ ,  $\tilde{\mathbf{h}} = [\tilde{h}_{1,0}, \dots, \tilde{h}_{n,0}]^T$  is the  $n \times 1$  vector of wireless channels at the  $n$  different center-frequencies  $\{f_{1,0}, \dots, f_{n,0}\}$ ,  $\|\cdot\|_1$  is the L-1 norm, and  $\|\cdot\|_2$  is the L-2 norm. Here, the constraint makes sure that the Discrete Fourier Transform of  $\mathbf{p}$  is  $\tilde{\mathbf{h}}$ , as desired. In other words, it ensures  $\mathbf{p}$  is a candidate inverse-NDFT solution of  $\tilde{\mathbf{h}}$ . The objective function favors sparse solutions by minimizing the L-1 norm of  $\mathbf{p}$ .

We can re-formulate the above optimization problem using the method of Lagrange multipliers as:

$$\min_{\mathbf{p}} \|\tilde{\mathbf{h}} - \mathcal{F}\mathbf{p}\|_2^2 + \alpha \|\mathbf{p}\|_1 \tag{4.10}$$

Notice that the factor  $\alpha$  is a sparsity parameter that enforces the level of sparsity. A bigger choice of  $\alpha$  leads to fewer non-zero values in  $\mathbf{p}$ .

This objective function is convex but not differentiable. Our approach to optimize for it borrows from

---

<sup>4</sup>We empirically evaluate the sparsity of indoor multipath profiles in typical line-of-sight and non-line-of-sight settings in 4.8.1.

---

## 1 Algorithm to Compute Inverse NDFT

---

▷ Given: Measured Channels,  $\tilde{\mathbf{h}}$   
 ▷  $\mathcal{F}$ : Non-uniform DFT matrix, such that  $\mathcal{F}_{i,k} = e^{-j2\pi f_{i,0}\tau_k}$   
 ▷  $\alpha$ : Sparsity parameter;  $\epsilon$ : Convergence Parameter  
 ▷ Output: Inverse-NDFT,  $\mathbf{p}$   
 ▷ Initialize  $\mathbf{p}_0$  to a random value,  $t = 0$ ,  $\gamma = \frac{1}{\|\mathcal{F}\|_2}$ .  
**while** *converged = false* **do**  
      $\mathbf{p}_{t+1} = \text{SPARSIFY}(\mathbf{p}_t - \gamma \mathcal{F}^*(\mathcal{F}\mathbf{p}_t - \tilde{\mathbf{h}}), \gamma\alpha)$   
     **if**  $\|\mathbf{p}_{t+1} - \mathbf{p}_t\|_2 < \epsilon$  **then**  
         *converged = true*  
          $\mathbf{p} = \mathbf{p}_{t+1}$   
     **else**  
          $t = t + 1$   
     **end if**  
**end while**  
**function** SPARSIFY( $\mathbf{p}, t$ )  
     **for**  $i = 1, 2, \dots, \text{length}(\mathbf{p})$  **do**  
         **if**  $|\mathbf{p}_i| < t$  **then**  
              $\mathbf{p}_i = 0$   
         **else**  
              $\mathbf{p}_i = \mathbf{p}_i \frac{|\mathbf{p}_i| - t}{|\mathbf{p}_i|}$   
         **end if**  
     **end for**  
**end function**

---

proximal gradient methods, a special class of optimization algorithms that have provable convergence guarantees [93]. Specifically, our algorithm takes as inputs the measured wireless channels  $\tilde{\mathbf{h}}$  at the frequencies  $\{f_{1,0}, \dots, f_{n,0}\}$  and the sparsity parameter  $\alpha$ . It then applies a gradient-descent style algorithm by computing the gradient of differentiable terms in the objective function (i.e., the L-2 norm), picking sparse solutions along the way (i.e., enforcing the L-1 norm). Algorithm 1 summarizes the steps to invert the NDFT and find the multipath profile.<sup>5</sup>

Inverting the NDFT provides Chronos with the time-of-flight on all paths. Chronos still needs to identify the direct path to compute the distance between transmitter and receiver. To do this, Chronos leverages that: of all the paths of the wireless signal, the direct path is the shortest. Hence, the time-of-flight of the direct path is the time corresponding to the first peak in the multipath profile.

It is worth noting that by making the sparsity assumption, we lose the propagation delays of extremely weak paths in the multipath profile. However, Chronos only needs the propagation delay of the direct path. As long as this path is among the dominant signal paths, Chronos can retrieve it accurately. Of course, in

---

<sup>5</sup>MATLAB implementation of this algorithm takes 3.1 s (standard deviation 0.6 s) for Chronos's implementation in Sec. 4.7.

some unlikely scenarios, the direct path may be too attenuated, which leads to poorer localization in that instance. Our results in 4.8.1 depict the sparsity of representative multipath profiles, and show its impact on overall accuracy.

## 4.5 Correcting for Phase Offsets

To work with practical Wi-Fi radios, Chronos has to address their inherent phase and frequency offsets:

- PLL Phase Offset:** Frequency hopping causes a random phase offset in the measured channel. This is because the phase-locked loop (PLL) responsible for generating the center frequency for the transmitter and the receiver starts at random initial phase (say,  $\phi_{i,0}^{tx}$  and  $\phi_{i,0}^{rx}$  respectively). As a result, the channel measured at the receiver is corrupted by an additional phase offset  $\phi_{i,0}^{tx} - \phi_{i,0}^{rx}$ . This phase offset, if left uncorrected, could render the phase information uncorrelated with the time-of-flight of the signal.
- Carrier Frequency Offset:** This offset occurs due to small differences in the carrier frequency of the transmitting and receiving radio. This leads to a time varying phase offset across each frequency band. Such differences accumulate quickly over time and need to be corrected for every Wi-Fi packet. Mathematically, in the  $i^{th}$  Wi-Fi frequency band, the receiver center frequency  $f_{i,0}^{rx}$  is slightly different from the transmitter center frequency,  $f_{i,0}^{tx}$ . As a result, the channel measurements at the receiver have an additional phase change which is proportional to  $f_{i,0}^{tx} - f_{i,0}^{rx}$ .

Let us refer to the channel values that incorporate phase and frequency offsets as CSI (channel state information), which is the typical term use in communication systems. Then, the CSI measured at the receiver for the  $i^{th}$  frequency band can be written as:

$$\text{CSI}_{i,0}^{rx}(t) = \tilde{h}_{i,0} e^{j(f_{i,0}^{tx} - f_{i,0}^{rx})t + j(\phi_{i,0}^{tx} - \phi_{i,0}^{rx})} \quad (4.11)$$

So how do we remove the phase and frequency offsets from CSI? To address this issue, Chronos exploits that, the phase and frequency offsets measured on one node with respect to another change sign when measured on the second node with respect to the first. Thus, if one would measure the CSI on the

transmitter with respect to the receiver, it would take the following value:

$$\text{CSI}_{i,0}^{tx}(t) = \tilde{h}_{i,0} e^{j(f_{i,0}^{rx} - f_{i,0}^{tx})t + j(\phi_{i,0}^{rx} - \phi_{i,0}^{tx})}. \quad (4.12)$$

Note that the channel,  $\tilde{h}_{i,0}$ , in equations 4.11 and 4.12 is the same due to reciprocity [85]. We can therefore multiply the CSI measurements at the receiver and the transmitter to recover the wireless channel as follows:

$$\tilde{h}_{i,0}^2 = \text{CSI}_{i,0}^{rx}(t) \text{CSI}_{i,0}^{tx}(t) \quad (4.13)$$

One may wonder how Chronos measure the CSI at the transmitter. Note however that as part of our channel hopping protocol both nodes have to transmit packets to each other. Hence, the CSI can be measured on both sides and exchanged to apply Eqn. 4.13.

The above formulation helps us only retrieve the square of the wireless channels  $\tilde{h}_{i,0}^2$ . However, this is not an issue: Chronos can directly feed  $\tilde{h}_{i,0}^2$  into its algorithm (Alg. 1 in 4.4) instead of  $\tilde{h}_{i,0}$ . Then the first peak of the resulting multipath profile will simply be at twice the time-of-flight.

To see why, let us look at a simple example. Consider a transmitter and receiver obtaining their signals along two paths, with propagation delays 2 ns and 4 ns. We can write the square of the resulting wireless channels from Eqn. 4.7 for frequency band  $i$  in a simple form:

$$\begin{aligned} \tilde{h}_{i,0}^2 &= (a_1 e^{-j2\pi f_{i,0} \times 2} + a_2 e^{-j2\pi f_{i,0} \times 4})^2 \\ &= a_1^2 e^{-j2\pi f_{i,0} \times 2 \times 2} + 2a_1 a_2 e^{-j2\pi f_{i,0} \times (2+4)} + a_2^2 e^{-j2\pi f_{i,0} \times 4 \times 2} \\ &= b_1 e^{-j2\pi f_{i,0} \times 4} + b_2 e^{-j2\pi f_{i,0} \times 6} + b_3 e^{-j2\pi f_{i,0} \times 8} \end{aligned}$$

Where  $b_1 = a_1^2$ ,  $b_2 = 2a_1 a_2$ ,  $b_3 = a_2^2$ . Clearly, the above equation has a form similar to a wireless channel with propagation delays 4 ns, 6 ns and 8 ns respectively. This means that applying Chronos's algorithm will result in peaks precisely at 4 ns, 6 ns and 8 ns. Notice that in addition to 4 ns and 8 ns that are simply twice the propagation delays of genuine paths, there is an extra peak at 6 ns. This peak stems from the square operation in  $\tilde{h}_{i,0}^2$  and is a sum of two delays. However, the sum of any two delays will always be higher than twice the lowest delay. Consequently, the smallest of these propagation delays is still at 4 ns – i.e., at twice the time-of-flight. A similar argument holds for larger number of signal paths, and can be

used to recover time-of-flight.

Finally, we make a few observations: (1) In practice, the forward and reverse channels cannot be measured at exactly the same  $t$  but within short time separations (tens of microseconds), resulting in a small phase error. However, this error is significantly smaller than the error from not compensating for frequency offsets altogether (for tens of milliseconds). The error can be resolved by averaging over several packets. (2) Delays in the hardware result in a constant additive value to the time-of-flight. This constant can be pre-calibrated once in the lifetime of a Wi-Fi-card, by measuring time-of-flight to a device at a known distance. (3) Standard Fourier Transform properties dictate that a minimum separation of  $\Delta f$  in frequencies of measured CSI values, leads to an ambiguity by multiples of  $\frac{1}{\Delta f}$  in the time estimates (i.e the delay is measured modulo  $\frac{1}{\Delta f}$ ). Since, Chronos uses CSI measurements at center frequencies, the minimum frequency separation is  $5 \text{ MHz}$ <sup>6</sup>. Hence, the time domain ambiguity is  $200 \text{ ns}$  which corresponds to a distance of  $60 \text{ m}$ , i.e., distance measurements are modulo  $60 \text{ m}$ . Thus, for indoor settings and typical Wi-Fi propagation, one can ignore the modulo factor.

## 4.6 Computing Distances and Location

So far, we have explained how Chronos measures the time-of-flight between two antennas on a pair of Wi-Fi cards. One can then compute the distance between the two antennas (i.e., the two devices) by multiplying the time-of-flight by the speed of light.

In order to get the location of the client from the distance measurements, Chronos follows a two-step procedure. In the first step, Chronos refines the distance measurements by utilizing geometric constraints, imposed by the relative locations of the antennas on the access point and the client. In the second step, Chronos formulates a quadratic optimization problem, based on the refined distances to get the accurate location of the client with respect to the access point.

Mathematically, we denote the separation between antenna  $i$  and antenna  $j$  on the access point by  $l_{ij}^{ap}$ . Similarly, antenna  $i$  and antenna  $j$  on the client are separated by  $l_{i,j}^{cl}$ . By using standard triangle inequality, we know that  $|d_{ij} - d_{i'j}| < l_{i,i'}^{ap}$ , where  $d_{ij}$  is the distance measured by Chronos between antenna  $i$  on the access point and antenna  $j$  on the client. When a pair of distances measured by Chronos violates this constraint; clearly, one or both of the distance measurements must be declared invalid. Chronos uses a

---

<sup>6</sup>The frequency separation is less than the channel bandwidth of 20 MHz due to overlapping Wi-Fi bands.

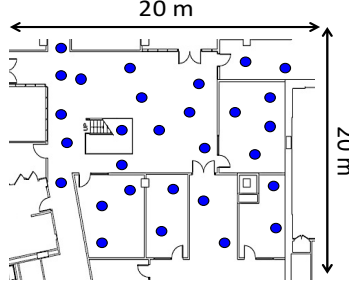


Figure 4-4: **Lab Testbed:** The figure depicts our testbed with candidate locations for the nodes marked with blue dots.

relaxed version of triangle inequality to eliminate erroneous distance measurements. Specifically, if we denote the maximum distance between any pair of antennas on a device by  $\alpha$ , Chronos chooses the largest cluster,  $C$ , of distance measurements such that each measurement in this cluster is at most  $\alpha$  away from at least one other distance measurement in the cluster. Chronos, then, discards the distance measurements that do not belong to  $C$ .

Finally, Chronos formulates the following constrained optimization problem to find the accurate position of the client. We denote the position of the  $i^{th}$  antenna on the access point by  $(x_i^{ap}, y_i^{ap})$ . Our goal is to optimize for the position of the client which we denote by  $(x, y)$ , where  $x$  and  $y$  are  $3 \times 1$  vectors of antenna coordinates:

$$\begin{aligned} & \min_{\epsilon > 0, x, y} \epsilon \\ & \text{such that} \\ & \forall (i, j) \in C, |dist((x_i^{ap}, y_i^{ap}), (x_j, y_j)) - d_{ij}| < \epsilon \\ & \forall (i, j) \in \{1, 2, 3\}, dist((x_i, y_i), (x_j, y_j)) = l_{i,j}^c \end{aligned}$$

where  $dist((x_1, y_1), (x_2, y_2))$  denotes the euclidean distance between points  $(x_1, y_1)$  and  $(x_2, y_2)$ . On a high level, Chronos optimizes for the minimal violation of the distance constraints while still maintaining the relative position of the antennas on the client. We formulate this problem as a quadratic-constrained optimization in MATLAB and use the fmincon solver to find the optimum solution. The average execution time for this algorithm is 0.09 s (standard deviation 0.01 s).



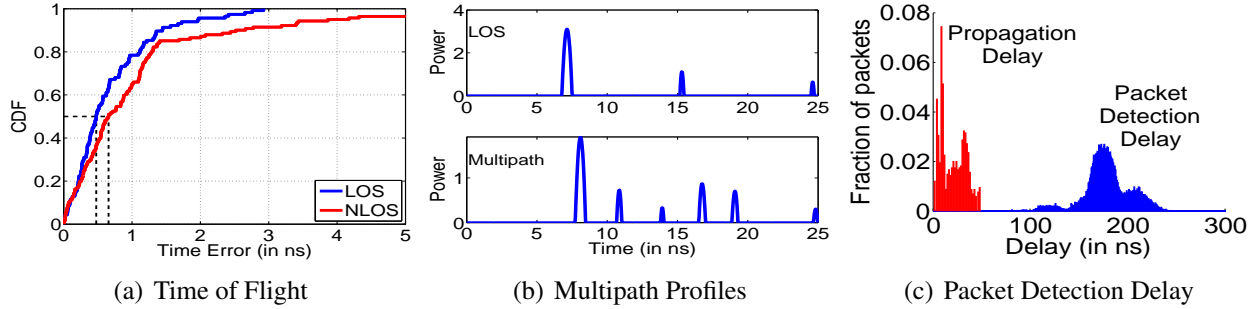


Figure 4-5: **Accuracy in Time of Flight:** (a) The CDF of error in time-of-flight between two devices in Line of Sight (LOS) and Non-Line of Sight (NLOS). (b) Representative multipath profiles. (c) Histograms of time-of-flight and packet detection delay.

## 4.7 Implementation

We implemented Chronos as a software patch to the `iwlwifi` driver on Ubuntu Linux running the 3.5.7 kernel. To measure channel-state-information, we use the 802.11 CSI Tool [87] for the Intel 5300 Wi-Fi card. We measure the channels on both 2.4 GHz and 5 GHz Wi-Fi bands.<sup>7</sup>

Unless specified otherwise, we pair two Chronos devices by placing each device in monitor mode with packet injection support on the same Wi-Fi frequency. We implemented Chronos’s frequency band hopping protocol (see 4.2) in the `iwlwifi` driver using high resolution timers (`hrtimers`), which can schedule kernel tasks such as packet transmits at microsecond granularity. Since the 802.11 CSI Tool does not report channel state information for Link-Layer ACKs received by the card, we use packet-injection to create and transmit special acknowledgments directly from the `iwlwifi` driver to minimize delay between packets and acknowledgments. These acknowledgments are also used to signal the next channel that the devices should hop to, as described in 4.2. We process the CSI to infer time-of-flight and device locations purely in software written in part in C++, MEX and MATLAB.

We note that all our experiments are conducted in naturally dynamic environments, specifically, an office building, a coffee shop and a home with four occupants. Chronos requires no modifications based on the changes in the environment. The environments have ambient Wi-Fi traffic. We could sense 3 to 19 different access points across our testbeds. Chronos disables the contention mechanism during hopping in order to enable fast switching across different Wi-Fi bands. This causes noise in Chronos’s measurements

<sup>7</sup>The Intel 5300 Wi-Fi card is known to have a firmware issue on the 2.4 GHz bands that causes it to report the phase of the channel  $\angle \tilde{h}_{i,0}$  modulo  $\pi/2$  (instead of the phase modulo  $2\pi$ ) [79]. We resolve this issue by performing Chronos’s algorithm at 2.4 GHz on  $\tilde{h}_{i,0}^4$  instead of  $\tilde{h}_{i,0}$ . This does not affect the fact that the direct path of the signal will continue being the first peak in the inverse NDFT (like in 4.5).

when there is a collision with other Wi-Fi packets. However, Chronos is resilient to noise on a small subset of the measurements. Moreover, since Chronos sends few packets on each Wi-Fi band, it does not adversely effect the Wi-Fi traffic.

## 4.8 Results

We evaluate Chronos’s ability to measure the time-of-flight, and compute a client’s position using a single access point.

### 4.8.1 Time-of-Flight Accuracy

We examine whether Chronos can deliver on its promise of measuring sub-nanosecond time-of-flight between a pair of commodity Wi-Fi devices.

**Method:** We run our experiments in the testbed in Fig. 4-4. In each experiment, we randomly pick a location for the AP. We then randomly pick a client location that is within 15 meter from the AP. We experiment with both line-of-sight and non-line-of-sight settings. We perform our experiments using a 10” ASUS EEPc netbook as a client and a Thinkpad W300 Laptop emulating a Wi-Fi AP via hostapd. Both devices are equipped with the 3-antenna Intel 5300 chipset. The antennas are placed at the corner of each device, which results an average antenna spacing of 30cm for the Thinkpad AP and 12cm for the ASUS client.

Using the above setup, we have run 400 localization experiments for different AP-client pairs. For each pair, we run Chronos channel hopping protocol. We compute the time of flight between each transmit antenna and receive antenna. We measure the ground-truth location using a combination of architectural drawings of our building and a Bosch GLM50 laser distance measurement tool [1], which measures distances up to 50 m with an accuracy of 1.5 mm. The ground truth time-of-flight is the ground truth distance divided by the speed of light.

**Time-of-Flight Results:** We first evaluate Chronos’s accuracy in time-of-flight. Fig. 4-5(a) depicts the CDF of the time-of-flight of the signal in line-of-sight settings and non-line-of-sight. We observe that the median errors in time-of-flight estimation are 0.47 ns and 0.69 ns respectively. These results show that Chronos achieves its promise of computing time-of-flight at sub-nanosecond accuracy. To put this in

perspective, consider SourceSync [181], a state-of-the-art system for time synchronization. SourceSync achieves 95<sup>th</sup> percentile synchronization error up to 20 ns, using advanced software radios. In contrast, the figure shows that Chronos’s 95<sup>th</sup> percentile error is 1.96 ns in line-of-sight and 4.01 ns in non-line-of-sight. Thus, Chronos achieves 5 to 10 fold lower error in time-of-flight, and runs on commodity Wi-Fi cards as opposed to software radios.

**Multipath Profile Results:** Next, we would like to examine whether multiple path profiles are indeed sparse. Thus, we plot candidate multipath profiles computed by Chronos in the above experiments. Fig. 4-5(b) plots representative multipath profiles in line-of-sight and multipath environments. We note that both profiles are sparse, with the profile in multipath environments having five dominant peaks. Across all experiments, the mean number of dominant peaks in the multipath profiles is 5.05 on average, with standard deviation 1.95 — indicating that they are indeed sparse. As expected, the profile in line-of-sight has even fewer dominant peaks than the profile in multipath settings. In both cases, we observe that the leftmost peaks in both profiles correspond to the true location of the source. Further, we observe that the peaks in both profiles are sharp due to two reasons: 1) Chronos effectively spans a large bandwidth that includes all Wi-Fi frequency bands, leading to high time resolution; 2) Chronos’s resolution is further improved by exploiting sparsity that focuses on retrieving the sparse dominant peaks at much higher resolution, as opposed to all peaks.

**Packet Detection Delay Results:** Past work on Wi-Fi time measurement and/or synchronization cannot measure the time-of-flight of a packet separately from its detection delay [181]. ([151] measures the distribution of detection delays but not the detection delay of a particular packet.) In contrast, Chronos has a novel way for separating the detection delay from the time-of-flight. We would like to understand the importance of this capability for the success of Chronos. Thus, we use the measurements from the above experiments to compare time-of-flight in indoor environments against packet detection delay.

Fig. 4-5(c) depicts histograms of both packet detection delay and time-of-flight across experiments. Chronos observes a median packet detection delay of 177 ns across experiments. We emphasize two key observations: (1) Packet detection delay is nearly  $8\times$  larger than the time-of-flight in our typical indoor testbed. (2) Packet delay varies dramatically between packets, and has a high standard deviation of 24.8 ns. In other words, packet detection delays are large, highly variable, and hard to predict. This means that if left uncompensated, these delays could lead to a large error in time-of-flight measurements. Hence, our results reinforce the importance of accounting for these delays and demonstrate Chronos’s ability to do so.

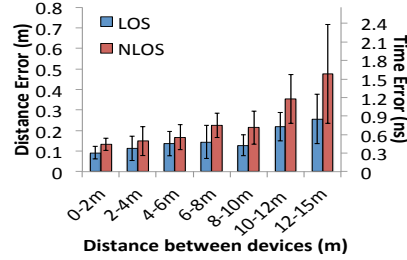


Figure 4-6: **Ranging Accuracy:** Plots error in distance across the true distance separating the transmitter from the receiver.

## 4.8.2 Localization Accuracy

We evaluate Chronos’s accuracy in measuring distance and location using a single access point.

**Method:** We compute the time-of-flight between the AP and user client in the testbed as described in 4.8.1 above. We use the measured time-of-flight to compute the distance between antennas and localize the client with respect to the AP as described in 4.6. We repeat the experiment multiple times in line-of-sight and non-line-of-sight.

**Location Results:** Fig. 4-7 plots a CDF of localization error using Chronos in different settings. The device’s median positioning error is 65 cm and 98 cm in line-of-sight and non-line-of-sight respectively. This result shows that Chronos’s accuracy is comparable to state-of-the-art indoor localization that use multiple AP’s [124, 127, 244].

**Ranging Results:** In some applications, it is important to maintain a particular distance between objects but the exact location is not necessary (e.g., preventing robot collision). Thus, here we plot the ranging results of Chronos. Fig. 4-6 plots the median and standard deviation of error in distance computed between the transmitter and receiver against their true distance. We observe that this error is initially around 10 cm and increases to at most 26 cm at 12-15 meters. The increase is primarily due to reduced signal-to-noise ratio at further distances. Note that the ranging accuracy is higher than the localization accuracy because ranging is a simpler problem (no need to find the exact direction) and Chronos’s time-of-flight computation naturally yields the range between devices.

## 4.8.3 Impact on Network traffic

Chronos enables localization between a pair of Wi-Fi devices without third party support. In many cases, these are user devices that do not otherwise communicate data between each other directly. However, an

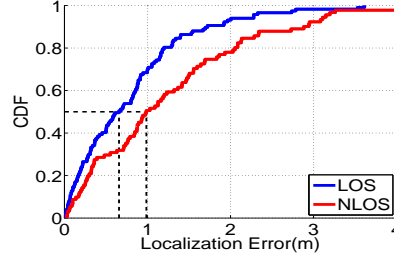


Figure 4-7: **Localization Accuracy:** Plots CDF of localization error in Line-of-Sight (LOS) and Non-Line-of-Sight (NLOS).

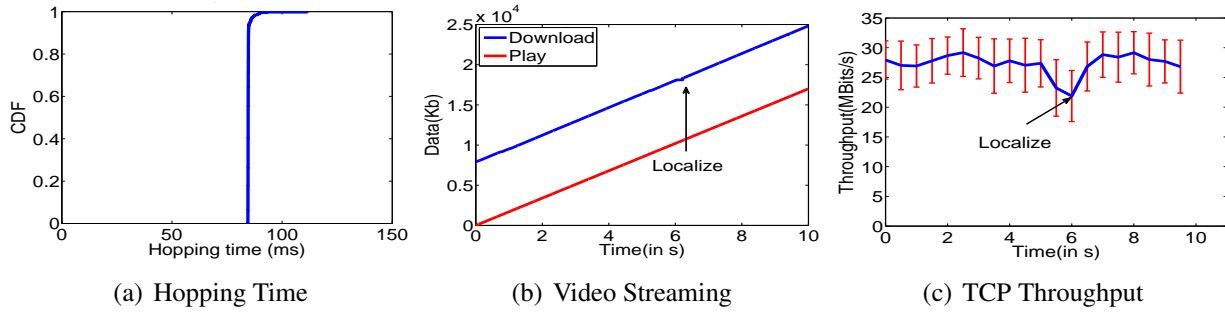


Figure 4-8: **Impact on Network Traffic:** (a) measures the CDF of time taken by Chronos to hop between all Wi-Fi bands – a small value of 84 ms. Consider a client-1 with a long-running traffic flow to an AP. The AP is asked to localize another client-2 at  $t = 6$  s. (b) depicts a representative trace of the number of bytes of data downloaded and data played over time if the client-1 views a VLC video stream. (c) measures the throughput if client-1 runs a TCP flow using iperf. In either case, the impact of client-1’s flow is minimal at  $t = 6$  s.

interesting question is the impact of Chronos on network traffic, if one of the devices is serving traffic, such as a Wi-Fi AP. This experiment answers three questions in this regard: (1) How long does Chronos take to hop between all Wi-Fi bands? (2) How does Chronos impact real-time traffic like video streaming applications? (3) How does Chronos affect TCP? We address these questions below:

**Method:** We consider a Thinkpad W530 Laptop emulating an AP and two ASUS EEPc netbook clients. We assume client-2 requests the AP for indoor localization at  $t = 6$  s. We measure the time Chronos incurs to hop between the 35 Wi-Fi bands. Meanwhile, client-1 runs a long-lasting traffic flow. We consider two types of flows: (1) VLC video stream over RTP; (2) TCP flow using iperf. We run the experiment 30 times and find aggregate results.

**Results:** Fig. 4-8(a) depicts the CDF of the time that Chronos incurs to hop over all Wi-Fi bands. We observe that the median hopping time is 84 ms for the Intel 5300 Wi-Fi card, like past work on commercial Wi-Fi radios [118].

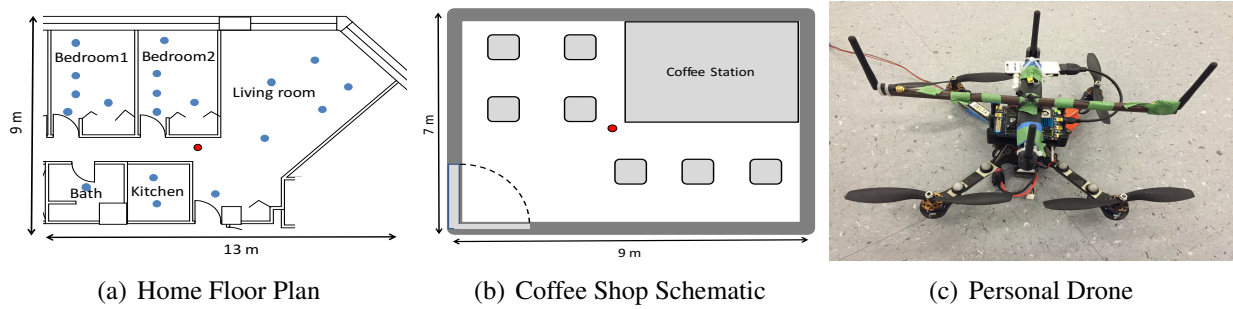


Figure 4-9: (a) Floor map of the apartment where Chronos is deployed. Red dot indicates the access point and the blue dots represent the client positions. (b) Coffee shop schematic. Red dot indicates the access point. (c) We implement Chronos on an AscTec Hummingbird quadrotor with an AscTec Atomboard.

Next, Fig. 4-8(b) plots a representative trace of the cumulative bytes of video received over time of a VLC video stream run by client-1 (solid blue line). The red line plots the cumulative number of bytes of video played by the client. Notice that at  $t = 6$  s, there is a brief time span when no new bytes are downloaded by the client (owing to the localization request). However, in this interval, the buffer has enough bytes of video to play, ensuring that the user does not perceive a video stall (i.e. the blue and red lines do not cross). In other words, buffers in today’s video streaming applications can largely cushion such short-lived outages [98, 97], minimizing impact on user experience. Similarly, Fig. 4-8(c) depicts a representative trace of the throughput over time of a TCP flow at client-1. The TCP throughput dips only slightly by 18.5% at  $t = 6$  s, when client-2 requests location. Thus, Chronos can support localization without much impact on data traffic. However, if more frequent localization is desired with large traffic demands, we recommend deploying a dedicated AP or Wi-Fi beacon for localization.

## 4.9 Applications

We evaluate Chronos in three application scenarios.

### 4.9.1 Room Occupancy Detection

Smart home technologies, such as personalized heating and lighting, can vastly benefit from information about the number and identity of people in individual rooms. Chronos is a natural solution for this problem as it can localize and track people using their smartphones and wearables, even if the home has a single

Wi-Fi access point.

**Method:** To demonstrate this capability, we deployed Chronos in a two-bedroom apartment that has four residents. The floor map of the apartment is shown in Fig. 4-9(a). The Chronos access point is centrally placed in the home and is indicated by the red dot. Each resident is given an ASUS netbook, equipped with Intel 5300 Wi-Fi cards, and running Chronos. The residents are then asked to move freely to locations within the apartment. Their locations are manually recorded and are marked by the blue dots in Fig. 4-9(a). Chronos measures the location of each resident and detects the room the person is in. In particular, Chronos distinguishes between the two bed rooms, living room, kitchen and bathroom.

**Results:** In our experiments, Chronos detects the user to be in the correct room in 94.3% of the experiments. Most of the errors occurred in Bedroom 1 in Fig. 4-9(a), and were due to the signal being too weak after traversing two walls and a closet. Overall, the results show that Chronos can enable applications based on room occupancy detection with a single home access point.

## 4.9.2 Wi-Fi Geo-Fencing

Chronos can be used to authorize Wi-Fi access in small businesses, which have only one access point. To demonstrate this capability, we deploy Chronos in a popular coffee shop with free Wi-Fi, and use the distance from the access point to measure whether an individual is inside or outside the coffee shop (Fig. 4-9(b)).

**Method:** We conducted 100 experiments in the coffee shop. The user used an ASUS netbook, equipped with the Intel 5300 Wi-Fi card to connect to the Chronos AP. In 50 of these experiments, the user was standing at a randomly chosen location inside the coffee shop, while in the others, the user was standing outside, while still being able to access the Wi-Fi connection.

**Results:** Chronos correctly inferred whether the user was inside or outside in 97% of experiments. However, if we simply authenticate users based on location without any buffer zone, the accuracy is 97%, but one legitimate customer cannot access Wi-Fi in his current location. In contrast, if we decide to accept users located within 30 cm of the premises, Chronos authenticates all legitimate customers but allows access incorrectly to people outside the premise in 5% of the experiments, decreasing the overall accuracy to 95%. Since it is more important to ensure customers can access Wi-Fi, we believe that one should use some buffer zone.

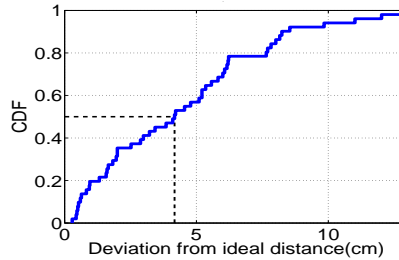


Figure 4-10: **Application to Personal Drones:** The drone uses Chronos to maintain a constant distance of 1.4 m to the user. The figure plots the CDF of errors in maintaining a distance of 1.4 m.

### 4.9.3 Personal Drones

We apply Chronos to indoor personal drones [33]. These drones can follow users around while maintaining a convenient distance relative to the control device in the user’s hand or pocket. Users can use these drones to take pictures or videos of them while performing an activity, even in indoor settings where GPS is unavailable.

**Method:** We use an AscTec Hummingbird quadrotor equipped with the AscTec Atomboard light-weight computing platform (with the Intel 5300 Wi-Fi card), a Go-pro camera and a Yei-Technology motion sensor. Fig. 4-9(c) depicts our setup. Note that the Intel 5300 Wi-Fi card supports 3-antennas; the fourth antenna on the quadrotor is placed only for balance and stability.

We perform our personal drone experiments in a 6 m × 5 m room augmented with the VICON motion capture system [3]. We use VICON to find the ground-truth trajectories of the personal drone and the user control device. In each experiment, the personal drone tracks an ASUS EEPc netbook with the Intel 5300 Wi-Fi card held by a user. The distance measurements from Chronos are integrated with drone navigation using a standard negative feedback-loop robotic controller [37]. The drone maintains a constant height and follows the user to maintain a constant distance of 1.4 m relative to the user’s device. The drone also captures photographs of the user along the way using the Go-Pro camera mounted on the Hummingbird quadrotor, keeping the user at 1.4 m in focus. The drone uses the compass on the user’s device and the quadrotor to ensure that its camera always faces the user.

**Results:** Fig. 4-10 measures the CDF of root mean squared deviation in distance of the drone relative to the desired value of 1.4 m — a median of 4.17 cm. Our results reveal that the drone tightly maintains its relative distance to the user’s device. Notice that our error in distance is significantly lower in this experiment relative to 4.8.2. This is because drones measure multiple distances as they navigate in the air,



which helps de-noise measurements and remove outliers. Chronos is the first system to achieve such a high accuracy in device to device positioning using no support from surrounding infrastructure.

## 4.10 Related Work

Chronos builds on vast literature on indoor Wi-Fi based localization [183, 43, 27, 251, 79, 244, 127, 124, 246, 9]. However, past work that delivers sub-meter location accuracy typically requires cooperation across multiple (four or five) AP's [79, 244, 127, 124, 246].

A few prior proposals have aimed to localize with a single Wi-Fi AP. They may be divided into two categories: some proposals [126, 254] require exhaustive fingerprinting of received signal power prior to deployment. Such proposals exhibit localization errors of several meters and incur a large overhead due to fingerprinting. The second class of proposals attempt to measure time-of-flight either directly [151], or indirectly using the phase [260]. However, since they cannot accurately measure the time-of-flight, they need the user to walk around, perform measurements in multiple locations, and intersect those measurements with the help of an accelerometer. In contrast, Chronos has tens of centimeter accuracy, and neither requires fingerprinting nor user motion.

A few past papers on Wi-Fi based localization leverage channel hopping [246, 245]. However, unlike Chronos which measures the absolute time-of-flight and localizes with a single AP, those systems measure differences in the time-of-flight and require the deployment of multiple AP's.

Prior theoretical ranging algorithms [228, 227] have used the Chinese Remainder theorem. However, Chronos differs from those algorithms in multiple ways. First, those algorithms ignore multipath and assume that wireless signals propagate in free space with a single time-of-flight value. In contrast, Chronos addresses the crucial problem of multipath, and hence its complete algorithm uses non-uniform Fourier transform as opposed to the Chinese Remainder theorem. Second, those algorithms ignore practical issues such as the frequency offset between the transmitter and the receiver and the inability of the receiver to separate the time of flight from the packet detection delay.

Finally, some past work has explored measuring the time-of-flight of Wi-Fi signals for reasons other than localization. There have been several studies that resolve time-of-flight to around ten nanoseconds using the clocks of Wi-Fi cards or other methods [239, 132, 78, 150, 181]. In contrast, Chronos can achieve sub-nanosecond resolution which is necessary for accurate localization.

## 4.11 Conclusion

This chapter describes Chronos, a system that measures sub-nanosecond time-of-flight on commercial Wi-Fi radios. Chronos uses these measurements to enable Wi-Fi device-to-device positioning at state-of-the-art accuracy, without support of additional Wi-Fi infrastructure or non-Wi-Fi sensors. By doing so, Chronos opens up Wi-Fi based positioning to new applications where additional infrastructure and sensors may be unavailable or inaccessible, e.g., geo-fencing, home occupancy measurements, finding lost devices, maintaining robotic formations, etc.

## Chapter 5

# Duet: Estimating User Position and Identity in Smart Homes Using Intermittent and Incomplete RF-Data

The vision of the smart home has captivated academia and industry since the 90's, when Bill Gates revealed the design of his new home where the lighting, music, and art collection change according to the person's taste and liking. Over the years, we have made advances towards that vision. Today, users can control heating, lighting, and appliances from their cell phones. However, current smart homes rely on explicit input from the user. We have not yet realized the complete vision of a smart home that tracks its occupants, monitors their habits, and adapts to their liking organically.

RF-based indoor localization can play a central role in realizing this vision of the smart home, by enabling the home to be aware of its occupants' locations and react to their presence. It can allow a smart home to track the user as she sits on the TV couch and tune to her favorite channel, alert the parents if, during their absence, the babysitter enters their bedroom instead of the baby's room, and turn the alarm clock off when the user actually leaves the bed. Further, there is much past work on RF-based localization [222, 124, 244, 127] and many of these systems achieve sub-meter accuracy.

The problem, however, is that RF-based localization systems typically rely on two assumptions neither of which is valid in the home scenario. First, a majority of RF-based localization systems assume that the user can be localized using signals emitted by their personal device, e.g., a cellphone. Yet, at home, it is typical for people to leave their phones behind, either on a counter or in their bags, or connected to a

charger, as they go about their activities. In particular, a recent study [55, 176] has shown that people spend 54% of the time away from their cellphones at home.

Second, RF-based localization systems assume that the RF signal along the direct line-of-sight to the user is not completely blocked. Although recent systems ([222, 124, 244]) can successfully localize in the presence of multipath, they fundamentally require the presence of the signal along the direct path. Yet, typical home structures may completely block the signal along the direct line of sight from the user to the receiver. TV screens, mirrors, HVAC are all metallic bodies that block RF signals. Thick structures such as pillars and in-wall plumbing can have the same effect. The blocking effect is further exacerbated by that, homes tend to have a single access point, unlike the enterprise where each location is covered by several access points. Once that single line-of-sight is blocked, there is no way to localize the user.

Some past work [14, 15, 229] has demonstrated device-free-localization, which can track a person using the RF signal reflected off her body. Such systems eliminate the need for having the user carry her phone, but they introduce their own challenges. First, they exacerbate the blocking problem. Since they rely purely on weak, low-power RF reflections off people's bodies, they are more likely to be blocked by home structures like TV, mirrors, HVAC, etc. Further, even when not blocked, they have a limited reach due to their lower power (100x lower power than Wi-Fi [15]). Second, device-free localization lacks the notion of identity. These systems are unable to identify who is sitting on the TV couch and who is cooking in the kitchen; they may end up tuning the TV to the wrong channel or making the temperature too high for the person in the kitchen.

This chapter introduces Duet, a location tracking system that is customized for the smart home. In contrast to past work, which has focused on new signal processing algorithms to infer physical location from radio signals, Duet focuses on addressing the above practical problems so that smart home applications can benefit from existing indoor localization algorithms.

Duet is a multi-modal system –i.e., a Duet AP has both a Wi-Fi radio for device-based localization and FMCW radio for device-free localization. Duet leverages device-free localization to track people's movements in the home even when they carry no personal devices on them. It then uses device-based localization to tag users' trajectories with their identities based on the users' intermittent interactions with their phones.

Duet's design has the following components which together deliver perpetual localization and identification:

- **Identification:** At first blush, it might seem that one can simply localize people using device-free localization, localize devices using Wi-Fi, and then identify each person by matching him with the nearest personal device. This however does not work. While the average accuracy of RF-localization is sub-meter, the 90th percentile error can be several meters. Further, localization error are likely higher in the home environment where, unlike the enterprise, one cannot cover each location by 3 to 5 APs. Such errors can easily position a device closer to the wrong person. Furthermore, being close to a cellphone does not mean that the person is using the device, and hence provides no information about the identity of the person. For identification, Duet considers only scenarios where the phone and user are moving and their movements are correlated. In this case, Duet uses the distance between the device and the person to infer a probability over them being matched. Whether this probability translates into the user being matched with the device is subject to other dependencies as explained below.
- **Blockage/Coverage:** Duet aims to locate a user even when she is outside the radio coverage area. To do so, Duet leverages that smart home applications care about the user’s symbolic location rather than her absolute location. For example, they want to learn whether the user in the bedroom or kitchen, whether she sat on the TV couch or at her desk, etc. Duet extracts the entry and exit boundaries to such symbolic spaces –e.g., the doors to each room, the bedside, the side of the couch for sitting. It then leverages this information to reason about entry and exit events to individual spaces. Even if parts of these spaces may be occluded or out of coverage areas, Duet can still identify the symbolic location of the user based on the entry and exit events. To encode this intuition, we leverage a Hidden Markov Model (HMM) for each symbolic space to reason about entry and exit events in a probabilistic manner. We note that entering and leaving at the specified boundaries is not a hard constraint –e.g. if a user enters the bed area at the bed foot, Duet corrects itself once it observes signals from the user in the bed.
- **Dependencies:** We recognize that there are dependencies between identity and location because the same person cannot be in two places at the same time. There are also dependencies between different symbolic locations. Say Duet sees two people as they enter their empty living room but later localizes two people on the couch and one person on the armchair, which are both in the living room. This would be a contradiction.

To deal with such dependencies, Duet introduces a framework which takes the probabilities of match-

ing identities with trajectories as well as the HMMs for all symbolic locations, and reasons holistically about the various dependencies. Duet’s reasoning framework combines probabilistic inference with first order logic. Specifically, at any point in time, Duet has an estimate of the most likely state of the home, where “state” refers to the identities and locations of people. Duet models each state as a set of variables, where each variable corresponds to a person in the environment. Each variable has three kinds of constraints: (a) possibility: a set of people that this variable can be assigned to, (b) impossibility: a set of people this variable cannot be assigned to and (c) area: the symbolic area of interest occupied by the person corresponding to this variable. This design allows the state to be very expressive. For example, it can accommodate constraints like: either Alice or Bob is in the kitchen, or, one person who is not Charlie is on the TV couch.

Duet can then reason about user location and identity by solving a satisfiability problem. While a general satisfiability problem in first-order logic is not decidable (i.e., it is impossible to determine if there exists a valid assignment to variables), Duet’s formulation of the satisfiability problem in this context is decidable.

While we have discussed Duet in the context of smart homes, its benefits extend to other smart environments. Even at work, users carry their phones only 56% of the time [55, 176]. Users often leave their phones on their desks, connected to the charger, and step out to chat with other employees or fetch some documents from the printer.

We have implemented Duet and evaluated it empirically in both, a home setting and an office environment. We ran our evaluation over an extended period (e.g., two weeks) with the actual occupants of the space. Further, we localize the occupants using their own devices (iPhones and Android phones) as they normally use them. To our knowledge, this is the first study that reports localization performance for real users using their own devices in their normal environment.

We compare Duet with a baseline that uses the-state-of-the-art Wi-Fi based localization [244]. We also evaluate the ability of Duet to deal with intermittent coverage and sparse interaction with devices and the resulting impact on accurate localization and identification. Our results reveal:

- Duet can accurately identify the person corresponding to a trajectory 95% of the times at home, and 93% of the times in the office. 36% and 56% of the correct identities (in home and office respectively) correspond to matching a device with the nearest user, while the rest were inferred using Duet’s first

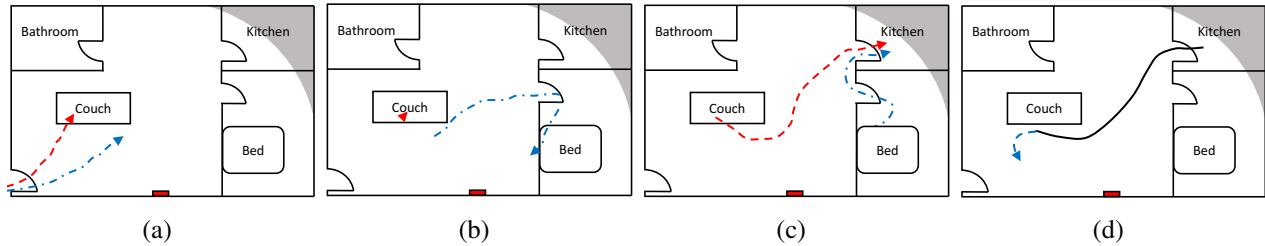


Figure 5-1: **Example:** Alice (red) and Bob (blue) are tracked using Duet. The corresponding events and states are described in table 5.1.

order logic model.

- In our experiments, over 45% of the time, the users were either blocked or outside the coverage area. Duet was able to correctly identify the symbolic space occupied by a user 96% of the time in the home and 94% of the time in the office environment.
- Overall, for the home and office environment, Duet’s accuracy in identifying the symbolic location of a person is 96% and 94% respectively. In contrast, the baseline’s accuracy is 17% and 42% respectively. The low accuracy of the Wi-Fi based localization baseline is mainly due to its assumption that users and their phones are co-located.

**Limitations and Scope:** As is common in pervasive computing [55, 176, 69], Duet assumes that the identity of the person is associated with her cell phone. In some cases, this may not be true. For example, a parent may give her kid her cellphone to play games. It may be easy to infer this usage through side channel information like the apps being used or the network traffic being generated [70, 137, 195]. We leave such an extension to future work.

## 5.1 Illustrative Example

Let us start with a simple example to show how Duet works. Consider the home in Fig. 5-1. Let the symbolic areas of interest be: kitchen, living room, bedroom, couch, and bed. For simplicity, assume that the only area out of radio coverage is the shaded area in the kitchen<sup>1</sup>. We will go through a sequence of

<sup>1</sup>The radio coverage of Wi-Fi based systems is typically larger than the radio coverage of device-free localization systems. Thus, for the rest of the chapter, the term ‘radio coverage’ has been used to refer to the range of the device-free localization system.

Event	Duet State
Alice (red) and Bob (blue) come in with their phones (Fig. 5-1(a)).	$v_1 = (\{Alice\}, LivingRoom), v_2 = (\{Bob\}, LivingRoom)$
Alice and Bob leave their phones behind; Alice goes to the couch to watch TV and Bob goes to the bed.	$v_1 = (\{Alice\}, Couch), v_2 = (\{Bob\}, Bed)$
Alice goes to the kitchen followed by Bob	$v_1 = (\{Alice\}, Kitchen), v_2 = (\{Bob\}, Kitchen)$
Bob comes out of the kitchen	$v_1 = (\{Alice, Bob\}, Kitchen), v_2 = (\{Alice, Bob\}, LivingRoom)$
Bob checks email on his phone	$v_1 = (\{Alice\}, Kitchen), v_2 = (\{Bob\}, LivingRoom)$

Table 5.1: **Example:** This table lists a sequence of home events (left column) and the corresponding Duet states (right column).

events and the corresponding sequence of states that Duet outputs. Recall, a state is a set of user variables, where each variable is defined by three constraints: a set of possible identities ( $P$ ), a set of impossible identities ( $I$ ), and a symbolic location ( $R$ ). In this simple example, there are no impossible identities. So, we simplify the notation for each variable to  $(P, R)$ .

Duet has four components: an identity-matching subsystem, one HMM that tracks access to each symbolic area, a probabilistic first order logic framework to leverage dependencies, and the device-free and device-based localization radios. Here, we illustrate how these components work together to contribute to the resulting state (the details of each component are left to later sections).

The home in our example has two occupants Alice and Bob. Table 5.1 shows a sequence of events in the first column and the corresponding Duet states in the second column. First, Alice and Bob come home from work and enter the living room carrying their cell phones. The corresponding Duet state is  $v_1 = (\{Alice\}, LivingRoom), v_2 = (\{Bob\}, LivingRoom)$  – i.e., Duet detects two people, identifies them as Alice and Bob, and locates them in the living room.

To produce this state, Duet has relied on its device-free localization which measured two user trajectories (the red and blue trajectories in Fig. 5-1(a)). These trajectories were fed to the HMM for the living room as observations. As a result, the HMM state changed and indicated two people in the living room. Duet’s identity matching component used Wi-Fi based localization to track the location of Alice’s and Bob’s phones and correlated it with the user trajectories. Recognizing that the two trajectories are highly correlated with the two phones, the identity matching component translated the high correlation into a



high probability that the red trajectory is Alice and the blue one is Bob. The logic took the outputs of the identity-matching component and the HMM, and decided that the best explanation of the evidence is that there are two people in the living room who are Alice and Bob.

Next, both Alice and Bob leave their phones on the coffee table to charge them. Alice (marked in red in Fig. 5-1(b)) goes to the couch to watch TV, while Bob goes to the bedroom to take a nap. As the bed is close to the living room, with just a wall between them, the errors in device-free localization lead Bob's final location to be inside the living room. However, since the blue trajectory in Fig. 5-1(b) shows an entry through the bedroom door but no corresponding exit through the door, Duet's HMM for the bedroom concludes that Bob is in the bedroom with high probability. Similarly, since the blue trajectory crossed the bedside but did not cross back, the HMM for the bed identifies Bob in the bed. Based on the evidence from both HMMs, the logic concludes that Bob is on his bed and could not have crossed to the living room through the wall. Thus, these events result in an update of Duet's state as follows:  $v_1 = (\{Alice\}, Couch)$ ,  $v_2 = (\{Bob\}, Bed)$ .

After some time, Alice goes to the Kitchen; later Bob also follows her to the Kitchen (Fig. 5-1(c)). Since only the kitchen's door is within radio coverage, the device loses track of the exact location of Alice and Bob inside the Kitchen. However, the device-free trajectories when fed as observations to the kitchen HMM, will cause the state to indicate that Alice and Bob are in the kitchen. In the absence of contradictory evidence, Duet's logic agrees and outputs the state:  $v_1 = (\{Alice\}, Kitchen)$ ,  $v_2 = (\{Bob\}, Kitchen)$ .

Next, Bob leaves the Kitchen and comes back to the living room. Device-free localization outputs a trajectory leaving the kitchen to the living room (Fig. 5-1(d)). The kitchen HMM and the living room HMM will show one person in each room. However since both Alice and Bob were outside the coverage area in the kitchen, Duet lost all the contextual information about them. Thus, the logic outputs:  $v_1 = (\{Alice, Bob\}, \{Kitchen\})$ ,  $v_2 = (\{Alice, Bob\}, LivingRoom)$ . This state implies that one of Alice and Bob is in the kitchen and the other is in the living room. At this moment, Duet presents a fuzzy state about the home, where two different outcomes are possible, unless further evidence is presented.

Finally, Bob picks up his phone to check his emails (Fig. 5-1(d)). This allows the identity matching component to tag the black trajectory in the living room as Bob with high probability. The respective HMMs still have the same states. Based on the interaction, the logic component can resolve the state and outputs:  $v_1 = (\{Alice\}, Kitchen)$ ,  $v_2 = (\{Bob\}, LivingRoom)$ . Duet would also go back to resolve the prior state recognizing that earlier Bob left the Kitchen and Alice stayed behind.

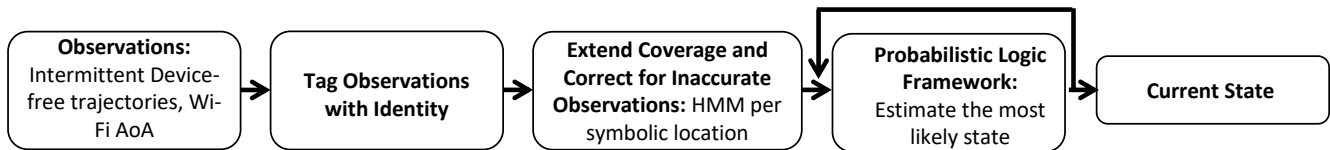


Figure 5-2: The different components of Duet.

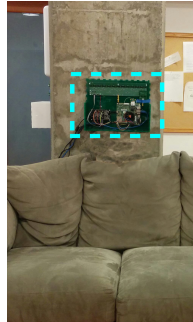


Figure 5-3: The Duet sensor (in blue rectangle) in a deployment.

This example illustrates the following:

- Duet can correct errors made by the underlying localization systems. This happened when the device-free localization system showed Bob in the living room, while he actually was on his bed.
- Duet can localize users even when they are outside the radio range – e.g., positioning Alice and Bob in the kitchen, even though the radio cannot sense them.
- Duet can successfully handle incomplete data and support late binding. This happened when Bob stepped out of the kitchen but Duet could not immediately tell the identity of the person leaving the kitchen. Duet then resolved this uncertainty once Bob interacted with his phone. This allowed Duet to go back to the earlier state and decide that Alice stayed in the Kitchen and Bob stepped out of it.

## 5.2 Duet Overview

Duet is a wireless sensor that hangs on the wall like a picture frame (Fig. 5-3). It takes as input the floor-map annotated with Duet’s own location, and symbolic areas of interest such as the bedroom, the bed, the TV couch, etc. It also takes the MAC address of the occupants’ phones, which it uses as their IDs. Duet tracks the location of the people in the home with respect to the symbolic areas.

Fig. 5-2 shows Duet’s four components and its workflow:

1. *Collecting Location Observations:* Duet tracks people using device-free localization and tracks their phones using Wi-Fi localization.
2. *Tagging Observations with Identity:* Duet has a subsystem that tags users' trajectories with identity based on the correlation between user movements and phone movements. The tagging is probabilistic and includes the option that the trajectory may be for a visitor, unknown a priori to Duet.
3. *Extending Coverage and Correcting for Inaccurate Observation:* Duet assigns an HMM to each symbolic area. The HMM takes the location measurements as observations and reasons about them to identify the number of people in its symbolic area.
4. *Reasoning with Probabilistic Logic:* Finally, the logic collects all evidence from the other components and estimates the most likely state of the home, where *state* refers to the identities and locations of people with respect to the symbolic areas.

The next few sections describe each component in detail. We note that while Duet can deliver its service with one sensor, Duet can also combine information across sensors to cover a large home.

## 5.3 Collecting Location Observations

Duet takes as input two types of location measurements:

### 5.3.1 Device-Free Observations

Duet is equipped with a multi-antenna FMCW sensor similar to that in [15], which uses RF-reflections to return the location of nearby individuals. The sensor returns a measurement every 20ms. Each measurement contains the location of the detected people. We assume that a person does not move much within 20ms, and hence nearby locations in consecutive measurements belong to the same person and identify her trajectory. We connect such consecutive nearby location estimates to create continuous stretches which we call tracklets. A tracklet looks similar to the blue or red lines in Fig. 5-1. A tracklet is discontinued whenever the signal from the moving user is too weak. This can happen when the user exits the coverage region or becomes blocked by a TV screen, a mirror, a thick closet, etc. It can also occur when the user

stops moving.<sup>2</sup> Tracklets have no identity. This is because the wireless signal reflects off all nearby humans. Thus, in contrast to a Wi-Fi transmission which is always tagged by its sender’s MAC address, in the absence of additional information, we cannot tell whose trajectory is captured by a particular tracklet.

### 5.3.2 Wi-Fi Based Observations

To add identity to tracklets, Duet uses the intuition that if a user is carrying her device, the motion of the device in the environment will match the motion of the user. Thus, Duet takes as input the MAC address of the devices associated with the home occupants. Duet can track multiple devices for the same user, e.g., her cellphone and laptop. However, for simplicity, we will assume that each occupant is recognized by one Wi-Fi device –her phone. Duet then measures how each device moves and try to correlate each tracklet with the motion of a phone.

One way to track the motion of phones is to continuously localize each of them using a Wi-Fi based positioning technique [124, 244]. Such systems typically require combining measurements across multiple Wi-Fi access points. However, we want Duet to be effective even if the user deploys only one Duet sensor in her home, i.e., one Wi-Fi access point. Thus, instead of measuring the location of the phone, we measure the angle of arrival (AoA) of its signal. Specifically, we use the Wi-Fi radio on our Duet sensor to compute the spatial direction along which the phone’s signal is received at our sensor. Due to multipath effects, the signal arrives along multiple directions. However, one of those directions should be the direct path from the phone to sensor.

There are many ways for measuring the power received along a spatial direction. In particular, for a multiple antenna radio, the power  $P$  received along a direction  $\theta$  at time  $t$  is:

$$P(\theta, t) = \left| \sum_i h_i(t) e^{\frac{j2\pi i l \cos(\theta)}{\lambda}} \right| \tag{5.1}$$

where  $h_i(t)$  is the channel measured at time  $t$  on the  $i^{th}$  antenna and  $l$  is the distance between two antennas. These measurements can be further refined by leveraging that Wi-Fi uses multiple frequencies called OFDM subcarriers ([124]).<sup>3</sup>

---

<sup>2</sup>When a user stops walking, his breathing signal can still be registered if he is close to the sensor. Yet since the inhale-exhale motion is very small, the ability to track a person based on those reflections becomes weaker –i.e., the coverage area of a static person is smaller than that of a moving person.

<sup>3</sup>We note that continuously pinging the phone to measure its AoA can significantly drain its battery. Duet pings phones only when it observes a person moving – i.e., observes a tracklet.

To summarize, Duet uses two types of location observations: user tracklets, and the power received from each phone along each spatial direction (output of Eqn. 5.1).

## 5.4 Tagging Observations with Identity

Next, for each tracklet, and for each phone, we want to estimate the likelihood that the person in the tracklet carried the phone. This provides us information about the identity of the person whose motion is captured by the tracklet. But how can we match a phone’s movement with a tracklet? As explained in the previous section, the phone’s movement is captured by the power received from it along each spatial direction –i.e., Eqn. 5.1. Thus, we need to correlate the phone’s AoA power profile in Eqn. 5.1 with how the AoA power profile would look like if the phone was located with the person in the tracklet.

Say the tracklet shows that the person is in location  $(x(t), y(t))$  at time  $t$ . Let us define  $d_i(t)$  as the distance between antenna  $i$  and the person’s location at time  $t$ . If the person has the phone on her at time  $t$ , then the amount of power that Duet would receive from the phone as a function of direction can be estimated by:

$$P_{est}(\theta, t) = \left| \sum_i \tilde{h}_i(t) e^{\frac{j2\pi i \cos(\theta)}{\lambda}} \right| \quad (5.2)$$

$$\text{where, } \tilde{h}_i(t) = \frac{1}{d_i(t)} e^{\frac{-j2\pi d_i(t)}{\lambda}} \quad (5.3)$$

In essence, Eqn. 5.3 estimates the wireless channel information as if the phone was at the same position at the users and then, Eqn. 5.2 uses these channel estimates to construct the power of the signal coming along each direction.

We can measure the similarity between the phone’s measured AoA profile, and the estimated one assuming the phone is carried by the moving person. In particular, let  $\vec{P}(t)$  be the vector of received power along all spatial directions and  $\vec{P}_{est}(t)$  the vector of estimated power along all spatial directions, had the phone been carried by the person. The similarity between them is captured by the normalized dot product of the two vectors:  $s(t) = \frac{\vec{P}(t) \circ \vec{P}_{est}(t)}{|\vec{P}(t)| * |\vec{P}_{est}(t)|}$ , where  $\circ$  represents the vector dot product and  $|\cdot|$  represents the L-2 norm. We can, then, estimate the similarity for the duration of the tracklet as  $\sum_t s(t)$ , where the sum is taken over all of the phone’s measurements that happen to occur during the tracklet. The larger the similarity, the more likely that the person in the tracklet was carrying that phone, and hence the

more likely that the person in the tracklet is the one who owns the phone.

We end with two notes:

- *Estimating Probabilities:* We would like to use the similarity function to compute a probability distribution over which phone may be carried by the person in the tracklet. We can do this by normalizing the similarities so that they sum up to 1. However, as we do that we have to account for the possibility of the user in the tracklet not carrying any of the phones known to Duet. To account for this case, we imagine a virtual phone whose  $\vec{P}(t)$  at any time is a random sample from past measurements of  $\vec{P}(t)$  values across all phones.
- *The Effect of Multipath:* In computing  $P_{est}$ , we have ignored the effect of multipath. While the multipath effect is easy to measure using an antenna array, it is very hard to predict multipath because such a prediction depends on the material and location of the walls and the furniture. However, two factors alleviate this problem. First, even in the case of multipath,  $P(\theta, t)$ , is high for  $\theta$  corresponding to the direct path. Second, as prior work ([124, 244]) has observed, multipath changes over time and only the direct path is consistently observed. Given that the multipath changes over the duration of the tracklet, it is unlikely to cause consistent bias.

## 5.5 Extending Coverage & Correcting Observations

Location observations can be erroneous or missing, as when the person leaves the coverage area. To deal with such issues, Duet extracts the entry and exit boundaries to the monitored symbolic areas –e.g., the doors to each room, the bedside, the side of the couch for setting. It then leverages this information to reason about entry and exit events to individual symbolic spaces. Even if parts of these spaces may be occluded or out of coverage, Duet can still identify the symbolic location of the user based on the entry and exit events. This approach works even when multiple people enter the out-of-coverage area.

So, how does one keep track of people entering and exiting a symbolic space? A naive solution is to mark the possible entrances and exits and increment or decrement a counter whenever a tracklet enters or leaves the space. However, this approach is not robust, primarily because location errors in device-free localization can lead to erroneous entrances and exits, leading to an incorrect estimate of the people count. To add robustness to the people count estimation, Duet uses a Hidden Markov Model (HMM). For

a specific area, the HMM takes as input a sequence of observations extracted out of the device-free system and outputs the number of people in the area. Below we give a short primer on HMM followed by our particular HMM design.

### 5.5.1 HMM Background

A hidden Markov model is a statistical model that tries to explain a sequence of observations with a sequence of hidden states. The key assumption underlying an HMM is that the probability of an observation given the current state is independent of any other observation or state. Mathematically, let us model the observation at time,  $t$ , as  $o_t$  and the hidden state as  $s_t$ . If  $P(X)$  denotes the probability of event  $X$ , then the assumption can be represented as:

$$P(o_t | s_0, s_1, s_2, \dots, s_{t-1}, s_t) = P(o_t | s_t) \quad (5.4)$$

Thus, the joint probability of having a sequence of states,  $\{s_t\}_{t=1}^{N_t}$  and observations,  $\{o_t\}_{t=1}^{N_t}$  is given by:

$$P(\{o_t, s_t\}_{t=1}^{N_t}) = P(s_0)P(o_0 | s_0) \prod_{t=1}^{N_t} P(o_t | s_t)P(s_t | s_{t-1}) \quad (5.5)$$

To model a problem as an HMM one has to define a set of possible states  $\mathcal{S} = \{\mathcal{S}_1, \dots, \mathcal{S}_M\}$ , a set of observations  $\mathcal{O} = \{\mathcal{O}_1, \dots, \mathcal{O}_N\}$ , a  $M \times M$  transition probability matrix  $\mathbf{T}$ , and a  $M \times N$  emission probability matrix  $\mathbf{E}$ . Transition probability is the probability of transitioning from state,  $\mathcal{S}_i$  to  $\mathcal{S}_j$  in consecutive time steps, i.e.,  $\mathbf{T}_{ij} = P(s_t = \mathcal{S}_i | s_{t-1} = \mathcal{S}_j)$ . The emission probability is the probability of an observation given a state, i.e.,  $\mathbf{E}_{ij} = P(o_t = \mathcal{O}_j | s_t = \mathcal{S}_i)$ . The set of observations and states is typically picked by the designer, and the transmission probabilities and emission probabilities are learnt from the data.

### 5.5.2 Design of the HMM

How should we pick the HMM states and observations for our application? One option is to directly use the location measurements as observations, and let the HMM determine when a tracklet enters an area, exits an area or stays within the area. This design, however, requires a lot of data to learn the HMM model. Further, the learned HMM will be dependent on the particular environment where the HMM was trained

and cannot be transferred to a new home. Instead, we use a more compact model that can be trained in one place and used in new homes.

**HMM Observations:** For each time instant, the observation can be: (a) an entry event ( $\mathcal{O}_{entry}$ ), (b) an exit event ( $\mathcal{O}_{exit}$ ), (c) number of tracklets ( $\mathcal{O}_k$ ) observed inside the area in this time step. This choice of observations does not require any information specific to the location and thus, can be trained only once. The entry and exit event signals are soft and can be overturned by the HMM based on the complete sequence of observations. In fact, this is exactly why we model the number of tracklets observed in an area as an observation in itself. For instance, if the system sees someone inside, even though it did not see an entry event due to a positioning error, the HMM will correct it to say that one person entered this area. Finally, we reiterate that the number of people inside an area is not necessarily equal to the number of tracklets observed in the area because the area is not completely covered by the device-free localization system.

**HMM States:** We use the HMM states to represent the number of people in the area at time  $t$ . Let us denote the state corresponding to  $k$  people as  $\mathcal{S}_k$ . Further, to allow the HMM to be expressive enough to detect exits and entrances, we introduce two new states for each value of  $k$ ,  $\mathcal{S}_{k \rightarrow k+1}$  and  $\mathcal{S}_{k \rightarrow k-1}$ . As the name implies, these states correspond to entry and exit events respectively.

**Learning and Inference:** Once the HMM model is defined, learning and inference can follow the standard algorithms in literature. The goal of the learning algorithm is to learn the transition probabilities matrix,  $T$ , and the emission probabilities matrix,  $E$ . In the absence of priors, the typical way to learn  $T$  and  $E$  is to compute their statistics using labeled data. So, we manually label a sequence of data with the correct states and use it to learn  $T$  and  $E$ . After the learning phase, the model can be used independent of the area, as described before. In the inference phase, the HMM is given a sequence of observations and is required to find a sequence of states that explain the observations. This is done using the Viterbi algorithm [71].

## 5.6 Reasoning with Probabilistic Logic

Duet uses the information it collects to infer the most likely state of the home. Note that the state of the home is different from the HMM state; the state of the home refers to the identities and locations of people



with respect to the symbolic areas of interest.

Duet has to reason about the information in a manner similar to how a human does it. For example, if Alice and Bob enter the bedroom and someone walks out, then Duet should be able to tell that the person walking out is either Alice or Bob. In addition, if the Wi-Fi data shows that the person inside is carrying Alice’s phone, then the system should be able to reason out that the person who had walked out was Bob.

To make such reasoning, we want to model three constraints. First, a human cannot be present in two different locations at the same time. Second, a human cannot walk into a location that he/she already occupies. Third, anyone walking out of a location has to be one of the people already present in the room.

These constraints are across different dimensions. The first constraint is spatial, the second one is temporal, and the third one is temporal and requires information about multiple people at the same time. This rich diversity in the constraints is great for aiding localization, but is not easy to model. For instance, it cannot be done via an HMM.

### 5.6.1 Context Model

To reason about people location and identity, Duet needs a model that captures the information it has and the constraints it needs to obey. We call this model “the context”, and describe it below.

**Hypothesis:** We model the context,  $\mathcal{C}$  as a set of  $N$  competing hypotheses, i.e.  $\mathcal{C} = \{\mathcal{H}_i | i = 1 \dots N\}$ , where  $\mathcal{H}_i$  denotes the  $i^{th}$  hypothesis. Each hypothesis is defined by a pair: a state,  $C_i$  and a belief score  $b_i$ . A higher belief score indicates a high probability of the system being in the corresponding state. While we used a single state in our illustrative example described in section 5.1, we use multiple hypotheses to enable the system to recover from errors made by the underlying localization systems. At any time, the system uses the hypothesis with the highest score to make decisions about symbolic locations of the users.

**State:** As discussed in section 5.1, each state is a set of variables, i.e.  $C_i = \{v_j | j = 1, \dots K\}$ , where  $v_j$  is the  $j^{th}$  variable and  $K$  is the number of variables. Here, each variable represents the constraints on the identity of a person. Each variable,  $v$ , is represented by three constraints: (a) a set of identities (say Alice, Bob, etc) that can be assigned to this variable,  $P$ ; (b) a set of identities that *cannot* be assigned to it,  $I$  and (c) the symbolic location of this person,  $R$  (say bathroom, bedroom, bed, couch). Exactly one identity can be assigned to one variable in a valid state. For completeness, if  $P$  and  $I$  have overlapping elements, the impossibility takes precedence, i.e., any person assigned to variable  $v$  must belong to the set difference,

---

## 2 Hypothesis Update Algorithm

---

▷ Given: Hypothesis  $\mathcal{H}_{in} = (C_{in}, b_{in})$ ; Event as 4-tuple of person, area, type and score,  $e = (p, r, type, score)$

▷ Output: A set of hypothesis,  $\mathcal{H}_{out}$

▷ Initialize  $\mathcal{H}_{out}$  to an empty set.

```

if  $type == entry$  then
   $v_{new}.P = p, v_{new}.I = [], v_{new}.R = r$ 
  for  $v \in C_{in}$  do
    if  $p \neq \mathcal{U}$  then  $v.I = v.I \cup p$  end if
  end for
   $\mathcal{H}_{out} = \{(C_{in} \cup v_{new}, b_{in} \times score)\}$ 
else if  $type == exit$  then
  for  $v \in C_{in}$  do
    if  $|(v.P - v.I) \cap \{p\}| > 0$  and  $v.R == r$  then
       $C_{new} = C_{in} \setminus v$ 
      for  $v' \in C_{new}$  do
        if  $p \neq \mathcal{U}$  then  $v'.I = v'.I \cup p$  end if
      end for
       $\mathcal{H}_{out} = \mathcal{H}_{out} \cup \{(C_{new}, b_{in} \times score)\}$ 
    end if
  end for
else
  for  $v \in C_{in}$  do
    if  $|(v.P - v.I) \cap \{p\}| > 0$  and  $v.R == r$  then
       $C_{new} = C_{in} \setminus v$ 
      for  $v' \in C_{new}$  do  $v'.I = v'.I \cup p$  end for
       $v.P = \{p\}, v.I = \{\}$ 
       $\mathcal{H}_{out} = \mathcal{H}_{out} \cup \{(C_{new} \cup v, b_{in} \times score)\}$ 
    end if
  end for
end if

```

---

$P - I$ . Finally, we use  $\mathcal{U}$  to represent the set of all identities in the universe, i.e., if  $\mathcal{U} \subset P$ , then anyone can be assigned to  $v$ , except people in  $I$ . Allowing the identity  $\mathcal{U}$  enables Duet to handle cases when there is no identity information available (for example, because their phone is powered off). Similarly, it enables Duet to deal with users who do not own/carry a device on them or whose device is not registered with the system.

**Events:** Duet uses three types of events: exit, entry and interaction. Recall, Duet uses one HMM per symbolic location to track the number of people in that location. Every change in the number of people in a symbolic location corresponds to an event. Specifically, the HMM state  $\mathcal{S}_{k \rightarrow k+1}$  corresponds to an entry event and the HMM state  $\mathcal{S}_{k \rightarrow k-1}$  corresponds to an exit event. These events are caused by a tracklet that entered or exited the space. In addition, Duet observes tracklets that are contained within one symbolic

location. Any interaction with a phone during these tracklets is considered as an event. Duet’s identity matching module assigns a probability score to each of the possible identities corresponding to all tracklets. Thus, each event is defined by a *set* of 4-tuples:  $\{(person, region, type, score)\}$ , where the *person* is the identity, *region* is the symbolic location corresponding to this event, *type* is entry or exit or interaction, and the *score* is the score assigned to this tracklet-identity pair by Duet’s identity matching module. Finally, when two regions are adjacent, exit from one region corresponds to an entry into another. In that case, the exit event is assumed to have occurred before the entry event.

### 5.6.2 Updating Context

Each hypothesis in a context gets updated after every event. Each hypothesis in the current context can fork into multiple hypotheses (since the events are probabilistic in nature) or collapse if the state corresponding to the hypothesis is invalid. We will define how we check for state validity later. Here, we discuss the algorithm for updating a hypothesis based on an event.

For simplicity, consider a single hypothesis ( $\mathcal{H}_{in} = (C_{in}, b_{in})$ ) and an event with only one possible identity. The exact algorithm to use the event to update the hypothesis is given in Algorithm 2. At a high level, if the event is of the type entry, then, a new variable  $v_{new}$  is created for the new person who has just entered the symbolic location,  $r$ , specified by the event. The possible identity set for this variable is set to the identity for this event,  $p$ . This new variable is now added to the state corresponding to this hypothesis. Further, the identity,  $p$ , is added to the impossibility set of every other variable in the state (since this identity cannot be assigned to any other person). Analogously, for an event of type exit, the algorithm removes a variable from the state and for an interaction event, it applies additional constraints to the current state. For exit events, the identity of the exiting variable is retained for future events caused by the same tracklet. As in our example in section 5.1, when the identity matching component failed to identify who exited the kitchen (thus labelling the identity as  $\mathcal{U}$ ), the system could leverage the fact that the kitchen has only two possible people, Alice and Bob. Thus, the exit tracklet (and subsequent entry to living room) could have matched to any one of them.

The above description is for the case when the identity matching module matches each event with just one identity. In the more general case, the event can be matched with multiple identities, i.e., either Bob entered the room for a score of  $s_1$  or Alice entered the room for a score of  $s_2$ . In this case, the event set has multiple elements with different identities and scores. This causes the input hypothesis to fork into

multiple hypotheses obtained by applying algorithm 2 on each element in the event set.

So far, we discussed the case when the context has a single hypothesis ( $\mathcal{H}_{in} = (C_{in}, b_{in})$ ). In general, Algorithm 2 is applied to each hypothesis in the current context. The updated context is the union of the hypotheses returned by Algorithm 2.

### 5.6.3 Checking State Validity

So far, we have considered how to model context as a set of hypotheses and how to update these hypotheses based on events. However, there is a very important aspect of this problem that remains to be discussed. How do you determine the validity of a state obtained after the update?

This notion of state validity is useful for two reasons. First, it prunes out invalid states and helps us identify the correct configuration out of the multiple possible states. Second, pruning out states with invalid configurations increases the constraints on the system that can be carried forward. To formalize this notion of states and validity checking, we leverage first-order logic theory. Similar logic tools have previously been used in the context of protocol verification, model-checking, etc, but Duet presents the first formulation of contextual states for estimating and tracking user location information.

**Definition:** A state is defined to be valid if there exists an assignment of identities to variables, wherein exactly one identity is assigned to each variable. To formalize this notion of validity, let us define the predicate  $In(x, X)$  to mean that an element  $x$  is in the set  $X$ . Further, let us define the predicate  $Ar(x, R)$  to denote that a person with identity  $x$  is physically present in area  $R$ . Recall, each variable  $v$  is defined by a triplet  $\{P, I, R\}$ , where  $P$  is the set of people that can be assigned to  $v$ ,  $I$  is the set of people that cannot be assigned to  $v$  and  $R$  is the area that this person is in. Thus, we can denote the constraint on the assignment of an identity to  $v$  as  $\exists x[In(x, v.P \setminus v.I) \wedge Ar(x, v.R)]$  ( $v.R$  is the value of  $R$  corresponding to variable  $v$  and  $v.P \setminus v.I$  represents the set difference  $v.P - v.I$ ). Thus, for a state which is a set of  $K$  variables, we need to model these constraints for each variable. The constraint becomes:  $\bigwedge_{i=1}^K \exists x_i[In(x_i, v_i.P \setminus v_i.I) \wedge Ar(x_i, v_i.R)]$ . Here,  $\bigwedge$  represents the logical *and* operator applied to multiple values.

We have modeled that every variable needs to be assigned an identity. Now, we add the constraint that no two variables can be assigned the same identity. So, we can define a state to be valid if and only if the

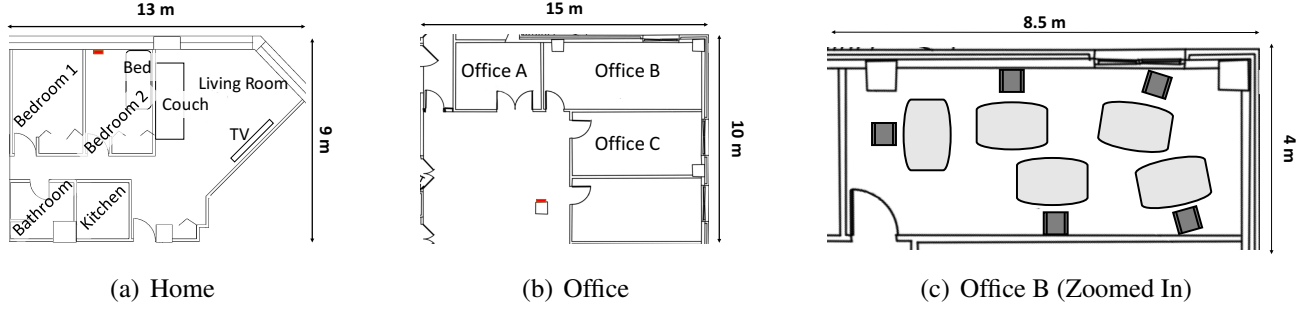


Figure 5-4: **Experimental Setup:** We deployed Duet in a 2 bedroom apartment as well as an office environment. The red box shows the location of our sensor. At home, we tracked access to the bed, two bedrooms, TV couch, kitchen and bathroom. In the office environment, we tracked access to the three offices. Within office B, we tracked access to each of the five desks.

following constraint (technically, a formula) is satisfiable:

$$\exists x_1, \dots, x_K \bigwedge_{i=1}^K [In(x_i, v_i.P \setminus v_i.I) \wedge Ar(x_i, v_i.R)] \bigwedge_{i=1}^K \bigwedge_{j=1}^{i-1} [x_i \neq x_j] \quad (5.6)$$

A formula in first order logic is satisfiable if there exists a valuation (an assignment of variables to constants) that makes the formula true. In our formulation, the set of constants are the possible identities that can be assigned. By incorporating the exclusive assignment constraints in Eqn. 5.6, we have modeled the state validity problem as a satisfiability problem in first-order logic. However, a general first-order satisfiability problem is not decidable, i.e., no algorithm today can ascertain if an assignment, that can make the formula true, exists.

**Checking Validity:** Duet leverages the insight that the formula in Eqn. 5.6 is not a generic first order formula, but has a specific structure that can be leveraged to decide satisfiability. Specifically, there exist two conditions that one can leverage. First, any variable  $v$  with  $\mathcal{U} \subset v.P$  is always satisfiable. Second, any variable  $v$  with  $\mathcal{U} \not\subset v.P$  and  $|v.P \setminus v.I| = 1$ , must be assigned the value in  $v.P \setminus v.I$ , as it has only one possibility. Duet builds on these insights to design the algorithm to check satisfiability. In appendix B, we present the detailed algorithm and a proof of correctness.

## 5.6.4 Discussion

To conclude, we discuss two design choices in our reasoning framework. First, the choice to model context as multiple hypotheses as opposed to one hypothesis allow us to correct for errors made by the reasoning framework itself. By tracking multiple hypotheses simultaneously, we allow future events to assign higher scores to currently less probable hypotheses and make them more likely in the future. For instance, in our example in section 5.1, when Bob exited the kitchen, imagine the system incorrectly estimated that the most likely outcome was that Alice exited the kitchen, and Bob was still inside. If we had just one deterministic hypothesis, even if the system later sees Bob entering the couch in the living room with very high probability, it would not be able to make the correct inference. It will simply reject the possibility of Bob being in the living room, since Bob is in the kitchen in the current state. But, in the current model, due to multiple hypotheses with different probabilities, Duet would continue to maintain both hypotheses, one where Bob exited the kitchen and the other where Alice exited the kitchen. When it makes a mistake, the incorrect hypothesis will win over temporarily but over time, as new events unfold, the correct hypothesis will become more likely.

Secondly, one potential problem with the approach above is that the number of hypotheses expands with each iteration. For  $N$  hypotheses in a context and  $T$  possible events, we get at least  $NT$  hypotheses in the next context. Duet controls the hypothesis space in two ways. First, hypotheses with invalid states get pruned out by using Algorithm 3. Second, Duet ranks all the hypotheses in the order of their belief scores. Then, it rejects all except the  $L$  most probable hypotheses<sup>4</sup>.

## 5.7 Implementation

We implemented Duet using off-the-shelf radios augmented with a custom PCB design. To implement a device-free positioning system, we use a multi-antenna FMCW radio similar to that used in [15]. The radio provides continuous real-time location measurements of individuals in the environment sampled every 20 ms. The algorithms developed in [15] are used to tease apart the signal reflections from multiple individuals when multiple individuals occupy the same space.

To get channel measurements for Wi-Fi packets transmitted by user devices, we use the two-antenna Intel 7265 Wi-Fi chipset, which provides channel state information (CSI) for each Wi-Fi packet. The 7265

---

<sup>4</sup>We set  $L = 10$  for our experiments.

chipset has just two antennas and hence cannot provide high-resolution angle of arrival (AoA). To improve the resolution, we built an 8-antenna switched array on a custom PCB. The array can be controlled by the Wi-Fi driver upon packet reception to switch the RF-chain from one antenna to the next. This allows Duet to create the equivalent of 8-antenna array using a 2-antenna off-the-shelf Wi-Fi chipset. As in past work, we correct for CFO by taking the ratio of the channels received on the two RF chains of the Wi-Fi card.

Duet’s algorithms described in sections 5.3, 5.4, 5.5 and 5.6 are implemented in Python and MATLAB. The HMM is trained on 9 days of manually labeled data before the evaluation deployment. Because of its design, the HMM can generalize and does not need to be trained separately for each area. The same model is used for all of our experiments, even though they are conducted in different spaces.

### 5.7.1 Evaluation Setup

We evaluate Duet in two different settings: a two-bedroom apartment measuring 9 m by 13 m (Fig. 5-4(a)), an office area measuring 10 m by 15 m (Fig. 5-4(b)). In the apartment, the symbolic locations are the bedrooms, the bed, the kitchen, the bathroom, and the TV couch, as shown in Fig. 5-4(a). In the office space, the symbolic locations are the offices marked as A, B and C in the figure. Further, within office B, we track access to 5 individual desks shown in Fig. 5-4(c), i.e., each desk-chair pair is marked as a symbolic area.

These environments vary in terms of size, occupancy, and symbolic locations. The home is inhabited by 2 people, with two other frequent visitors; the office space has 9 occupants. Specifically, Office A has one occupant, Office B has five and Office C has three occupants. In addition, this space has occasional visitors. The smallest area tracked in the home is the couch which is about 1.3 m<sup>2</sup>. In the open area, each desk (and chair) occupy about 1.5 m<sup>2</sup>.

Our experiments were conducted over two weeks. The location of our device is shown as a red rectangle in Fig. 5-4. The system operates in real-time. We ask users to register their devices in our system and consent to be continuously monitored. The users do not need to install any apps to be tracked by the system. As users enter the monitored environment, Duet automatically updates the current list of devices based on their Wi-Fi transmissions and starts tracking them.

All of our experiments are conducted in natural, dynamic environments with *no change to user devices or behavior*. To the best of our knowledge, our system is the first to present real-time continuous

localization with users' own devices.

**Ground Truth:** To obtain the ground truth, we placed cameras in the common areas and hand-labeled all data for ground truth by watching the camera videos. For areas of interest inside a bedroom (i.e., the bed), the occupant of that bedroom used a camera that he controls and a diary to keep track of bed use. All occupants consented for the camera monitoring.

**Baseline:** For baseline, we deploy an angle-of-arrival based Wi-Fi localization system in each of these spaces. Specifically, we deploy multiple access points and compute angle-of-arrival of the cellphones of all users using the MUSIC algorithm, which is the state-of-the-art algorithm for Wi-Fi localization. Then, we triangulate these measurements across the access points to get the actual location of the user devices. We re-iterate that these multiple Wi-Fi access points *do not* feed into Duet and are deployed just for a baseline comparison. The baseline uses two 8-antenna Wi-Fi radios positioned along the two axes in each of the spaces. The use of 8 antennas on the access points delivers accurate Angle-of-arrival based localization. To ensure that the baseline is accurate, we measure the localization accuracy of the baseline. The baseline achieves a median localization accuracy of 97 cm, which is comparable to the state-of-the-art work in [124].

## 5.8 Results

We present our evaluation of Duet below.

### 5.8.1 Overall Accuracy

We start by reporting the overall accuracy of Duet and comparing it with the Wi-Fi-based localization baseline. To calculate accuracy, we measure the percentage of time for which a user's symbolic location is correctly estimated. For example, if a user is in their bed from 5 PM to 6 PM, and the system estimates that the person is in the bed from 5:30 PM to 6:00 PM and not on the bed otherwise, the system would be 50% accurate.

We measure the accuracy for each day of deployment and plot the mean and standard deviation in Fig. 5-5. The wide bars in the figure show the average over days, whereas the small bars show the standard deviation. As seen in the figure, Duet can achieve a high accuracy of over 96% in home and 94% for the



office area. In comparison, the Wi-Fi localization baseline does much worse and is able to correctly identify the symbolic location only 17% of the times at home, 42% of the time in the office.

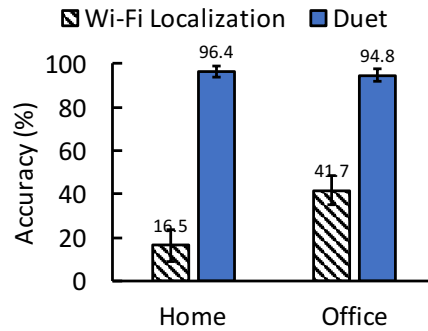


Figure 5-5: **Accuracy:** The figure compares localization and identification accuracy of Duet with Wi-Fi localization baseline.

The primary reason for the large errors in the baseline is that the users do not carry their cellphone with them all the time. Since the baseline system just monitors the cellphones of the users, it fails to notice when a user enters or exits a symbolic location without their cellphone. Also note that while past work shows that users tend to be within arm’s reach of their phones 46% percent of the time in the home and 56% percent of the time otherwise, these numbers put an upper bound on the accuracy of the baseline. The actual accuracy of the baseline is significantly lower because being close does not mean being in the same symbolic location. In particular, a significant percentage of the time that the users spend at home is in their beds. Their phones are typically close but not in the bed. In contrast, Duet addresses these issues and dramatically improves the location accuracy.

### 5.8.2 Error Breakdown in Duet

The above section shows that Duet’s errors are limited to a few percent. But where do those errors come from? We classify the errors into three types: (a) Insufficient Context: The user is not carrying her device and there is no prior contextual information about the user, (b) Incorrect Wi-Fi tagging: the user is carrying her device but her tracklet is incorrectly tagged with the id of another user. This could result from localization errors or a user interacting with another person’s device. (c) Error Propagation: the event is incorrectly marked because of an incorrect notion of the current state due to incorrect Wi-Fi tags from the past. As can be seen in Fig. 5-6, a majority of the errors are because of the lack of context when a person enters the environment (e.g., a user entering the coverage area and leaving it without interacting with a

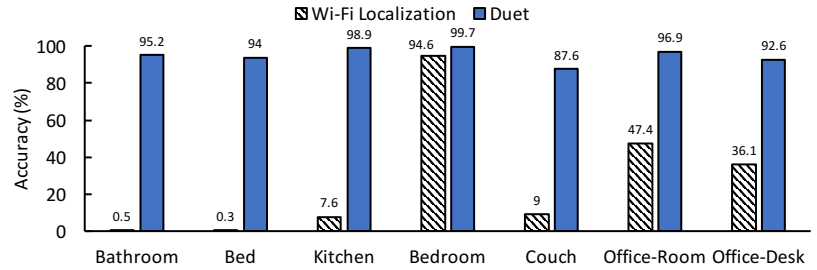
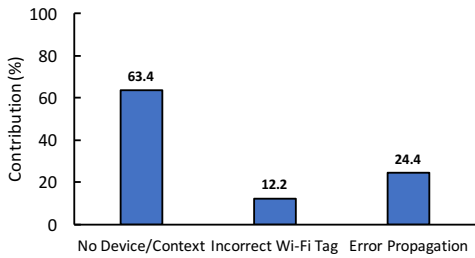


Figure 5-6: Breakdown of the errors experienced in Duet’s deployments

Figure 5-7: Accuracy results for different symbolic locations

phone and hence providing no context to infer her/his identity.) The overall low error rate and the fact that most errors are due to lack of contextual information imply that Duet’s probabilistic logic is reasoning accurately about various constraints and dependencies.

### 5.8.3 Accuracy in Sub-spaces

To further understand the working of Duet and user behavior, we measure the accuracy for different areas in our deployments. We plot the variation of accuracy over the different types of physical spaces in Fig. 5-7. The results indicate an interesting trend. For the Wi-Fi baseline, the accuracy goes up to 95% in the bedroom, but falls down to almost zero in the bathroom and the kitchen. This is explained by user behavior. When users are at home, they tend to leave their devices in one spot, such as the bedroom, but typically do not carry it to the kitchen or the bathroom. Thus, the Wi-Fi localization baseline correctly estimates user’s current area, only if she is in the space where she left her device (in this case, the bedroom).

In contrast, Duet can easily reason about these spaces by keeping track of contextual information. For instance, if a user comes into the bedroom, sets her alarm on the phone and sleeps on the bed, we can use the brief interaction with the phone to identify the user and continue to track her to her bed. As a result, for Duet, the accuracy is consistently over 90% for all the spaces.

Finally, observe that the performance of both Duet and the Wi-Fi baseline degrades when monitoring the desks in Office B as compared to monitoring the rooms themselves. This stems from two reasons. First, Office B represents a very cluttered environment. Each of the desks in this space has two large monitor screens and a couple of PC’s. Consequently, the presence of a large amount of metal in this area leads to the direct path of the signal being blocked more often, leading to more errors. Secondly, each of the desks is a smaller symbolic location and naturally, localization errors have larger impact on accuracy.

## 5.8.4 Identity Matching

The previous results show the overall accuracy for localization and identification. Here, we zoom in on the identification problem. We plot the percentage of trajectories that could be correctly tagged with the user identities if we just relied on the matching algorithm. Specifically, we pick the highest probability identity for each tracklet and compare the identity with the ground truth identity for the same tracklet. The percentage of tracklets correctly labeled by the matching algorithm is plotted in Fig. 5-8. For reference, we also plot the percentage of correctly labeled tracklets if we use the end-to-end Duet pipeline.

As seen in the plot, the percentage of tracklets correctly labeled by the matching algorithm is 36% in home, 56% in office area. These numbers go up to 95% and 93% once Duet applies its HMMs and probabilistic logic. This result shows that one cannot simply tag a tracklet with the identity of the nearest phone. This is because of two reasons: people do not carry their phones on them all the time, and both device-based and device-free localization make errors. Duet can correct these errors and fill in missing information based on its other components (the HMM's and the first order logic) and correctly identify the people in over 93% of the tracklets.

## 5.8.5 Expanding Coverage

An objective of Duet is to expand the coverage of the underlying localization systems in space and time. To assess if Duet could successfully achieve this in our deployment, we plot the percentage of time a person is located by the device-free localization system in Fig. 5-9.

As seen in the figure, the coverage provided by the device-free localization system is about 48% of the time in the home deployment and about 60% of the time in the office deployment. Note that there are two reasons for a person to be out of coverage. First, they are out of range of the device or they are blocked completely by metallic bodies like TV, HVAC, etc. Second, since device-free systems try to isolate humans from surrounding reflectors based on motion. Thus, if a human is still, they are unlikely to be tracked by the device. Duet can overcome this lack of coverage by leveraging the HMMs to track access to symbolic spaces. Thus, even if the user sleeps on the bed and disappears due to lack of motion, Duet uses the information that the user entered the bed and never exited it to continue to identify her symbolic location.

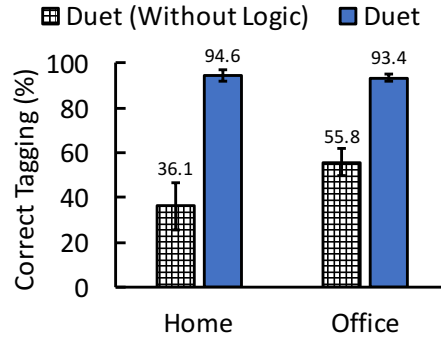


Figure 5-8: **Tagging Accuracy:** Duet’s reasoning framework improves the accuracy of identifying people corresponding to tracklets.

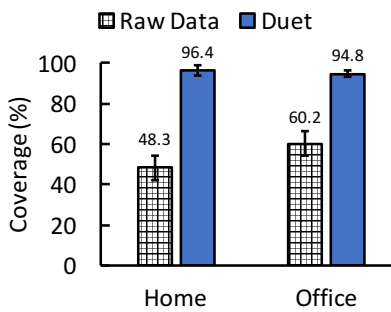


Figure 5-9: **Coverage:** Duet expands the coverage of the underlying device-free system in both home and office deployments.

### 5.8.6 Event Detection Accuracy

We note that some smart home applications are reactive, such as changing the TV channel, adapting the lights to the user’s liking, or turning the alarm off once the user steps out of her bed. These applications care about when the user steps in or out of the symbolic area. Thus instead of the percentage of time a user is localized accurately, which is the metric in section 5.8.1, their measure of accuracy focuses on detecting entrance and exit events and the identity of the user performing the event. To capture the interest of this class of applications, we also measure the event detection accuracy of Duet. Specifically, we consider each entry or exit from a symbolic area as an event. The accuracy is measured as the percent of such events for which the entry/exit is correctly detected *and* the identity of the user performing the action is correctly labeled. The event has to be detected and tagged with the user identity in less than 5 seconds to be considered correct.

We plot the event detection accuracy of Duet as well as the Wi-Fi localization baseline in Fig. 5-10. Duet achieves a high accuracy of around 94% in both the environments. On the other hand, the accuracy

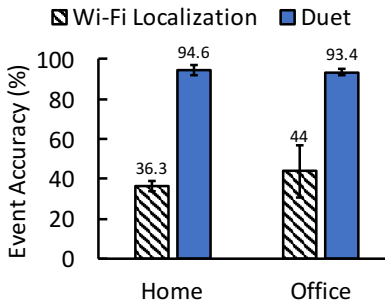


Figure 5-10: **Event Accuracy:** The figure plots the accuracy of detecting entry and exit events.

achieved by Wi-Fi localization is much lower due to the reason described before, namely, people are not carrying their phones on them as well as the localization errors. Duet can use past information to weed out incorrect choices and hence, more accurately identify the people performing the actions.

## 5.9 Related Work

**RF-based Localization:** Duet is inspired by past work on RF-based localization. Over the last two decades, researchers have used variety of localization techniques that exploit RSSI [27, 43, 183, 251], angle of arrival [244, 79, 112], time of flight [222, 246], or a combination of these concepts [124, 127, 230, 128, 193] to achieve sub-meter median location accuracy for RF devices. Some papers have even demonstrated methods that can localize a device using one access point [222, 236]. However, past literature focuses on improving localization accuracy assuming the user carries a wireless device on her. In contrast, Duet focuses on scenarios where users often leave their devices behind, at their desks, in their bags, or connected to a charger. Duet introduces a new design that can reason about intermittent and partial location data to deliver perpetual localization.

Duet also builds on past work on device-free localization [15, 14, 229, 252, 247]. These systems typically do not have a notion of identity. The few that identify different users do it only when the user walks on a specific predefined path, for which the system is calibrated [13, 243, 232]. In contrast, Duet focuses on identifying users independent of the path they walk. Further, it presents new techniques that allow localization systems to reason about intervals during which the user is out of coverage.

**Use of Contextual Information to Augment Localization:** Our work is also related to past papers that leverage contextual information to improve the performance of localization systems [211, 189, 16, 10].

These solutions, however, assume that location measurements are always available, and focus on improving the accuracy by leveraging the floor map or the fact that people do not walk across walls. In contrast, Duet focuses on scenarios where location fixes are intermittent and hence there is a need to reason both across space and time to understand how people move at home.

Some past work also focuses on leveraging sensors deployed in the environment to understand the interactions between people and surrounding objects [31, 177, 191, 116]. For example, they may tag all windows and doors to detect the opening/closing of a window/door and use this information to infer the location of a user [31]. When deployed, such sensors can play an important role in augmenting Duet to capture interactions with specific objects. However, a system that localizes people purely based on their interaction with a large number of tagged objects is hard to maintain over time, and furthermore, lacks identity information.

**Other Sensing Modalities:** Beyond RF-based localization systems, other sensing modalities like visible light [261, 129, 249, 255, 138, 139] or acoustic signals [142, 23, 168] have been used for indoor positioning. Much like RF positioning systems, they either require the user to carry a device to be localized [261, 129, 249, 255, 142, 23, 168] or lack identity information [138, 139]. We believe these systems are orthogonal to Duet and can complement the framework proposed in this chapter by presenting additional streams of information or replacing one of the two streams of data used in Duet.

## 5.10 Conclusion

We present Duet, a system for continuous localization and identification of users in a smart home. We base our design on a logical reasoning formulation that can enable a smart home to reason about the location and identity of different users in the environment, even when partial, intermittent location data is available. Our experience from two weeks of continuous deployment in users' natural environments suggests that Duet can act as a substrate for different kinds of smart home applications, ranging from playing music to long term analytics of user behavior.

## Chapter 6

# Eliminating Channel Feedback in Next-Generation Cellular Networks

The high cost of cellular spectrum has motivated network providers to seek advanced MIMO techniques to improve spectral efficiency [106, 6, 248]. Yet, only point-to-point MIMO multiplexing can be performed efficiently in current networks [111]. More advanced MIMO solutions such as massive MIMO [133], coordinated multi-point [135], distributed MIMO [182], and multi-user MIMO [17], all require the base station to know the downlink channels prior to transmission. In the absence of this information, the base station cannot beamform its signal to its users. Today, the only way to learn the downlink channels is to have the user perform the measurements and send the channels back to the basestation. Measuring the channels on the one thousand LTE subcarriers for every antenna on the base station, and feeding those measurements back to the base station generates much overhead [44, 248, 242, 202]. This feedback overhead is excessive even in today's networks which have a limited number of antennas on the base station – about 4.6 Mb/s of signaling per user in a 20 MHz  $4 \times 2$  network [106, 6]. The problem escalates in future 5G networks which rely on large MIMO systems with many antennas (massive MIMO, CoMP, etc.). In fact, the LTE standardization body that is investigating high-order MIMO systems with up to 64 antennas (Release 13), has declared this problem as a major challenge for future LTE networks [111]<sup>1</sup>.

Our goal is to enable cellular base stations to estimate the downlink channels *without* any user feedback. A natural approach that can help us achieve this goal is channel reciprocity [115]. Reciprocity

---

<sup>1</sup>For example, with 64-antenna base stations, the need to learn the downlink channels consumes 48% of the traffic generated by the base station, simply to send per-antenna reference signals [111].

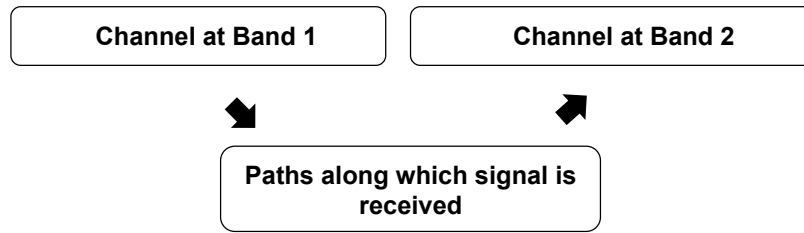


Figure 6-1: **R2-F2’s Approach:** R2-F2 extracts the paths of the signal from channels on band-1 to reconstruct the corresponding channels on band-2.

implies that uplink and downlink channels are the same,<sup>2</sup> so long as both the base station and the clients transmit on the same frequency band. Indeed, reciprocity has been proposed to minimize channel feedback in Wi-Fi networks [141, 81], where both the access point and its clients transmit on the same frequency. Unfortunately, the vast majority of today’s cellular connections (including every LTE network in the U.S. [188]) employ Frequency Division Duplexing (FDD) [102], i.e., they transmit data from the phone and base station at different dedicated frequency bands. Thus, extending reciprocity to LTE networks requires answering the following fundamental question: *How do we infer the wireless channels on one frequency band by observing those channels on a different band?*

We introduce R2-F2, a system that does exactly that – i.e., it can infer the RF channels on one band by observing them on a different band. Before we dive into R2-F2, let’s explain why wireless channels vary across frequency bands in the first place. RF signals are waves whose phase changes with time and frequency. The wireless channels are the result of those waves traversing multiple paths, reflecting off walls and obstacles, then combining at the receiver. Due to their frequency-dependent phases, RF waves that combine to reinforce each other on one frequency may cancel each other on another frequency. As a result, wireless channels could look quite different at different frequencies.

R2-F2 infers wireless channels across frequencies by leveraging a simple observation: while the channels change with frequencies, the underlying physical paths traversed by the signal stay the same. Hence, R2-F2 operates by identifying a transform that allows it to map the observed channels to the underlying paths, then map them back to the channels at a different frequency, as shown in Fig. 6-1.

But how do we identify a frequency-invariant transform for mapping channels to paths? It is natural to look into past work on RF-based localization systems since, like us, they need to relate RF channels to the underlying paths. Localization systems [244, 124, 9, 128, 127] exploit the MIMO antennas on a base

<sup>2</sup>Modulo a constant factor.



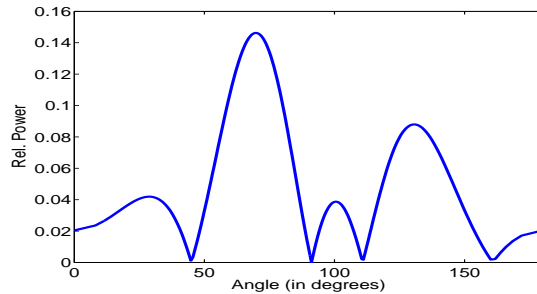


Figure 6-2: **Power Profile:** The power profile represents the relative power of the signal coming along different spatial directions.

station to create a power profile that shows the spatial directions of the incoming signal, as illustrated in Fig. 6-2. Each peak in the profile is, then, associated with the direction of an underlying path. Unfortunately, these localization power profiles are unsuitable for our purpose. While they reveal information about the direction of the signal, they lack information about the exact distance travelled by the signal and whether the path is direct or reflected off a wall. Such missing parameters introduce frequency-dependent phase variations in RF waves travelling along different spatial paths, and hence, change the channel values. Furthermore, in Sec. 6.3, we show that, due to windowing and superposition effects, the power profiles change with frequency and deviate from the spatial directions of the underlying paths. Our empirical results in Sec. 6.7 demonstrate that using the localization power profiles for recovering the underlying channels eliminates 60% of MIMO SNR gains.

R2-F2 builds on the insights learned from RF-localization, but it is the first to enable LTE base stations to infer the downlink channels without any feedback, and at an accuracy suitable for MIMO techniques. In Sec. 6.4, we explain how we design a channel-to-path transform that incorporates the information needed to predict channels across frequencies. We further embed this transform in a full system that overcomes additional practical challenges, including accounting for: (1) frequency offset between the user and the base station; (2) hardware differences in transmit and receive chains; and (3) packet detection delay — all of which affect wireless channels differently at different frequency bands.

We built R2-F2 in USRP radios and integrated it with LTE OFDM. Our testbed emulates a small cell setting with a 5-antenna LTE base station. We deploy our base station within a few meters from one of the LTE base stations on our campus. Since we cannot transmit in the cellular spectrum, we operate our testbed on the 640-690 MHz white space frequency band, which is in the vicinity of the Verizon LTE band (only 30 MHz away). Our results reveal the following:

- For an uplink-downlink frequency separation equal to that in AT&T and Verizon networks, the channels computed by R2-F2 deliver accurate MIMO beamforming within 0.7 dB of the beamforming obtained with the ground-truth channels. The resulting SNR increase has improved the average data rates in our testbed by  $1.7\times$ . This result shows that R2-F2 can be used by MIMO solutions to deliver LTE throughput gain while eliminating channel feedback overhead.
- R2-F2 can also be used to eliminate interference at cell edges and improve spatial reuse. In our testbed, R2-F2 reduced the SNR of the interfering signal from 9 dB to only 0.9 dB.
- The quality of R2-F2's inferred channels remains high across frequencies separated by up to 40 MHz, which is larger than the LTE uplink-downlink separation in most US LTE deployments. Further, the degradation of SNR with uplink-downlink separation is less than 0.2 dB per 10MHz.

To our knowledge, R2-F2 is the first system that demonstrates the practicality of inferring LTE downlink channels from uplink channels using reciprocity and without channel feedback. This result contributes a better understanding of reciprocity in FDD systems, and a solution to one of the important challenges facing future 5G MIMO networks.

## 6.1 Related Work

Related work falls under two broad categories.

**(a) Channel Estimation in Cellular Networks:** Much prior work has reported the excessive overhead associated with channel estimation and feedback in cellular networks [44, 248, 106, 242, 202]. Even in today's networks, which have a relatively small number of antennas, the feedback overhead can be prohibitive – as much as 4.6 Mb/s of signalling traffic per user in a  $4\times 2$  system [106, 6]. All recent LTE releases recognize this challenge [6, 7, 5]. To mitigate the problem, the standard allows for either sending full channel information, or compressing the information using a codebook of limited values. Unfortunately, neither option is satisfactory since the former causes excessive overhead, whereas the latter leads to poor channel resolution that impedes the gains of MIMO techniques [164, 77, 114]. As a result, only point-to-point MIMO is common in today's LTE networks (in the US), and more advanced techniques, such as MU-MIMO have yet to gain deployment traction [65]. This problem is increasingly critical with

the advent of 5G networks which rely on large MIMO systems (e.g., massive MIMO) to increase spectral efficiency [131, 204].

Past work on addressing this problem has focused on various techniques for compressing channel feedback [44, 248, 184, 204]. R2-F2 is motivated by the same desire of learning downlink channels with minimal overhead, but it aims to eliminate channel feedback altogether, and replace it with passive inference of channel values.

A few papers study reciprocity in the context of FDD systems. In particular, Hugel et. al [99] observe that the channels at two cellular FDD bands are correlated and hence postulate that one can infer downlink channels from uplink channels. Some papers [88, 101, 174, 175] propose theoretical models that use large antenna arrays to infer channels on the downlink from those on the uplink. Their models are either based on long-term channel statistics and do not account for fast variations, or are based on the angle of arrival power profile (used in RF localization), which we show in Sec. 6.7 to yield poor performance in practice. Further, they do not account for practical challenges in system design such as the limited LTE bandwidth (typically 10MHz), carrier frequency offset (CFO) and detection delay. In contrast, R2-F2 does not need long-term statistics and is empirically demonstrated in a testbed deployment. R2-F2 achieves this through a new design that relates the channels to frequency-invariant parameters (e.g., path lengths), compensates for frequency dependent parameters (e.g., path phases), and accounts for distortion factors (e.g., window effect).

**(b) Related Work Outside Cellular Networks:** R2-F2 is related to the problem of channel quality estimation. Some applications aim to infer channel quality on a particular frequency band, but do not need the exact channel values. For example, two Wi-Fi nodes may want to select the best quality Wi-Fi channel for their connection without actively running measurements on all Wi-Fi channels [50, 194]. The same applies to cognitive radios in the White Spaces [180]. These systems observe the channel on one or more bands and use that information to infer the SNR of the channel on a different band –i.e., the channel quality. In contrast, R2-F2 needs to infer the full channel values–i.e., it needs both the phase and the magnitude of the channel for every OFDM sub-carrier and every antenna.

R2-F2 is also related to past work that focuses on estimating the channels across a large band of spectrum by sub-sampling the frequencies in that band. For example, the work in [30] subsamples the spectrum and uses compressive sensing to recover the channel values at the missed bands. This approach does not apply to LTE networks since the observed uplink channels do not satisfy the sampling requirements of

compressive sensing (i.e. the uplink channel is only available on one contiguous band).

There is also a large body of work that aims to predict wireless channels in the future based on their values in the past [241, 39, 61]. This work does not predict channels across frequency bands. R2-F2 is complementary to this work in that it estimates wireless channels at different values of frequency as opposed to different points in time.

Finally, we note that R2-F2 is related to a wide range of systems for the TV whitespaces that aim to predict occupancy [38, 201] or interference [259] by hopping between a minimal number of frequency bands. R2-F2 complements these systems by estimating the wireless channel at any target frequency band based on sampling the channel at one other band.

## 6.2 Background

In this section, we list a few known results in modeling wireless channels, which are important for the rest of the exposition. Note that the mathematical expressions refer to the transmission frequency by the corresponding wavelength  $\lambda$ .

Wireless channels describe how the signal changes as it propagates from transmitter to receiver. They are a direct function of the paths along which the signal propagates as well as the transmission frequency. In particular, the channel of a narrowband signal traversing a single path is given by [216]:

$$h = ae^{-j2\pi\frac{d}{\lambda}+j\phi} \quad (6.1)$$

where  $\lambda$  is the wavelength,  $a$  is the path attenuation,  $d$  is the distance the path traverses, and  $\phi$  is a frequency-independent phase that captures whether the path is direct or reflected. Since the signal travels along multiple paths, say  $N$ , the channel at a receive antenna can be written as:

$$h = \sum_n^N a_n e^{-j2\pi\frac{d_n}{\lambda}+j\phi_n}, \quad (6.2)$$

which is the sum of the channel components over all paths that the signal takes between transmitter and receiver.

Finally, we note that base-stations have multiple antennas, so they obtain one channel per antenna. For

a  $K$  antenna base station, the set of channels,  $h_i$  on antenna  $i$  is:

$$h_i = \sum_n^N \left( a_n e^{-j2\pi \frac{d_n}{\lambda} + j\phi_n} \right) e^{-j2\pi \frac{il \cos \theta_n}{\lambda}}, \quad (6.3)$$

where  $\theta_n$  is the angle-of-arrival of the signal along path  $n$ ,  $d_n$  is the distance travelled by the signal along path  $n$  to the first antenna and  $l$  is the pairwise separation between antennas on the base station. Note that the above equation depends both on frequency and all underlying signal propagation paths.

### 6.3 Intuition Underlying R2-F2

R2-F2's primary objective is to infer wireless channels on a target frequency band, given the wireless channels on a different frequency band. In order to achieve this objective, R2-F2 relies on the observation that the channels are the direct result of the signal paths. While the channels change across frequencies, the underlying paths stay the same. Thus, if one could obtain a frequency-invariant representation of signal paths from wireless channels on any given frequency, one can recreate an estimate of the channels at any other frequency of interest.

But what is a frequency-independent representation of signal paths that can be mapped to (and from) wireless channels? The answer to this question lies in Eqn. 6.3, which defines wireless channels based on underlying propagation paths. Specifically, wireless channels  $h_i$  depend on four distinct attributes of signal paths: (1) Their attenuation  $a_n$ ; (2) Their frequency-independent phase  $\phi_n$ , that distinguishes the direct path from reflected paths; (3) Their angle of arrival  $\theta_n$ ; (4) The distance they traverse  $d_n$ . These four quantities, when listed for each path, fully define the wireless channels on any given frequency. More importantly, none of these parameters depend on the frequency at which the channel is obtained. In other words set of four-tuples of the form  $(a_n, \phi_n, \theta_n, d_n)$  is a natural representation of signal paths that is frequency-invariant.

Now that we have a representation of signal paths, we need to understand how to extract it given wireless channels on any frequency. To do so, observe that wireless channels in Eqn. 6.3 take the form of the familiar discrete Fourier transform (parameterized by spatial angle-of-arrival  $\cos \theta$ ). In particular, this Fourier transform takes as input quantities that depend directly on our signal path four-tuples. Since the discrete Fourier transform is invertible, one might wonder if we can simply apply the inverse Fourier

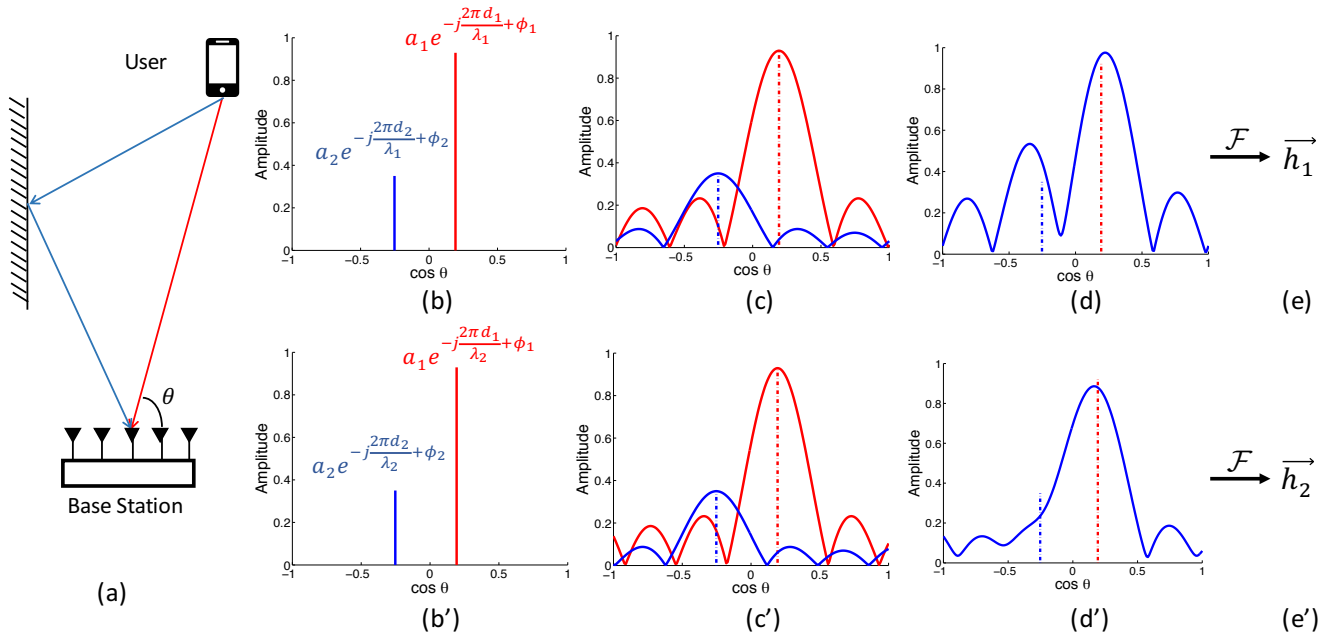


Figure 6-3: **Transforming Signal Paths to Channels on Two Frequency Bands:** (a) Consider two signal paths emerging from  $80^\circ$  and  $105^\circ$  as shown. Their corresponding attenuations are:  $a_1, a_2$ , distances traversed are  $d_1, d_2$ , and phase offsets due to reflectors (or lack there-of)  $\phi_1, \phi_2$ . (b)-(b') Depicts the signal components of individual paths across angle-of-arrival. We observe two spikes at  $80^\circ$  and  $105^\circ$  as expected scaled by the respective path amplitudes. The peaks differ only in phase. (c)-(c') Incorporates the windowing effect that causes the peaks to be convolved with sinc functions. The red and blue sinc correspond to the red and blue path. Further, the width of the sincs changes with frequency. (d)-(d') Depicts the superposition of the sinc functions in both frequency bands. The two plots look very different – both due to the difference in shape of the sincs as well as the difference in their phases. (e)-(e') Denotes the wireless channels obtained after applying the Fourier Transform on the two bands – two different sets of values.

transform to retrieve the signal paths given wireless channels. Unfortunately, our task is not this simple. This is because, upon inverting the Fourier transform, we get quantities that depend not just on our signal path four-tuples, but also the frequency. As a result, teasing apart signal four-tuples from wireless channels requires removing this dependency on frequency.

To understand how to achieve this, it is instructive to study how the same signal 4-tuples manifest as different wireless channels on two different frequencies, say 600MHz and 650MHz. We do so in the context of a specific example. Consider Fig. 6-3(a) which depicts signals from the phone to the base station traversing two paths. Let the corresponding signal path 4-tuples be:  $(a_1, \phi_1, \theta_1 = 80^\circ, d_1 = 19.5m)$  and  $(a_2, \phi_2, \theta_2 = 105^\circ, d_2 = 23m)$ . These undergo four distinct transformations, inclusive of the Fourier transform, before they become the overall wireless channels on the two frequencies (from Fig. 6-3(a) to

(e)-(e')) as described below:

- **Phase Variation (Fig. 6-3(a) to (b)-(b')):** We first begin by mapping the signal path 4-tuples to inputs of the Fourier transform. Recall from Eqn. 6.3 that these inputs are simply the wireless channel components along individual paths at the two frequencies. Fig. 6-3(b)-(b') visualizes the amplitude and phase of the signal components from the two paths across angle-of-arrival. As expected, both these plots have two spikes that correspond to the two paths, scaled by their respective attenuations. In fact, the two plots differ only in the phase of the spikes, which scales inversely with the wavelength of the two bands.
- **Windowing Effect (Fig. 6-3(b)-(b') to (c)-(c')):** Before we can apply the Fourier transform, we need to account for an effect that occurs since the cellular base station has a limited number of antennas (5 in our example). Specifically, this means that the base station samples the signal from the two paths within a window of space (the space between the first and last antenna). Since the channels are observed only within a window of space, the signal's angles of arrival are convolved with a sinc function. This is a standard property of the Fourier transform: multiplying by a window in one domain translates into a convolution with a sinc in the other domain. Thus each impulse from the corresponding angle as in Fig. 6-3(b) and (b') is transformed as a sinc function as shown in Fig. 6-3(c) and (c'). The convolution with a sinc makes the signal look different across frequencies. Specifically, the precise shape of the sinc changes with the transmission frequency. This is because the distances between antennas are measured relevant to the wavelength of the transmission signal. Hence, at higher frequency the distance between antennas seem larger and the sinc narrower.
- **Superposition (Fig. 6-3(c)-(c') to (d)-(d')):** At this point, the signals components from different paths super-impose at the receiver. Thus, the base station gets a super-position of the blue and red sincs in Fig. 6-3(b) and (b'), scaled by their respective phases that (as described earlier) are different. This results in Fig. 6-3(d) and (d') that now look significantly different.
- **Fourier Transform (Fig. 6-3(d)-(d') to (e)-(e')):** Finally, we apply the Fourier transform to take our super-imposed sincs in Fig. 6-3(d) and (d') to the wireless channels sampled at the five antenna locations, and shown in Fig. 6-3(e) and (e').

Thus, if we want to move from wireless channels to their underlying signal paths, we must invert this

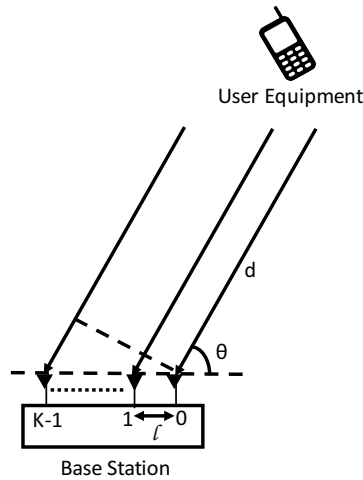


Figure 6-4: **Antenna Setup** The base station is equipped with multiple antennas and receives signals on a fixed bandwidth.

whole process. We need to: (1) Invert the Fourier transform; (2) Separate the super-imposed sines; and (3) Undo the windowing effect; (4) Correct for the difference in phase between the two frequencies. In the next section, we describe this process mathematically and formulate it as an L-2 minimization problem.

## 6.4 Algorithm

In this section, we formalize the discussion in Sec. 6.3. We begin by formalizing mathematically the transform between wireless channels and signal paths. We then invert the effects of this transform by formulating the problem as an L-2 optimization whose solution results in the frequency-invariant signal path characteristics. These paths are then used to infer the channels on a desired band.

### 6.4.1 Transforming Physical Paths to Wireless Channels

Let us assume that the signal from the user to the  $K$ -antenna base station arrives along  $N$  distinct paths. The antennas are indexed by  $0, 1, \dots, K - 1$  and  $l$  is the inter-antenna separation.<sup>3</sup> Further, let us denote the signal path 4-tuple of the  $n^{\text{th}}$  path to be  $(a_n, \phi_n, \theta_n, d_n)$ . The setup (with just one path) is illustrated in Fig. 6-4.

<sup>3</sup>In typical antenna arrays, the inter-antenna spacing is set to  $\frac{\lambda}{2}$ , where  $\lambda$  is the signal wavelength.



We now present the mathematical formulations of the individual transforms described in Sec. 6.3, between our signal path representation and wireless channels (see Fig. 6-3).

- **Phase Variation:** Since the Fourier transform described in equation 6.3 operates on the  $\cos \theta$  domain, we discuss the algorithm in terms of  $\cos \theta$ . Let us denote  $\psi = \cos \theta$ . Then, the contribution of the  $n^{\text{th}}$  path to the directional representation of the signal at wavelength,  $\lambda_1$  can be given by:

$$P_1^n(\psi) = a_n e^{-j \frac{2\pi d_n}{\lambda_1} + j\phi_n} \delta(\psi - \psi_n), \quad (6.4)$$

where  $P_1^n(\psi)$  represents the signal component along direction  $\psi$  and  $\delta(\cdot)$  is the impulse function.  $\phi_n$  denotes the phase accumulated by the  $n^{\text{th}}$  path by virtue of undergoing reflections,  $a_n$  is the amplitude of the path as received on antenna 0 and  $\psi_n = \cos \theta_n$ . This representation corresponds to the representation in Figs. 6-3(b) and (b').

- **Windowing Effect:** As described before, the signal sensed by the antenna array along different spatial directions is the inverse Fourier transform of the channel measurements on the different antennas of the antenna array. Since the antennas on the base station sample a finite space, it is equivalent to applying a window on the antenna domain of width  $\frac{L}{\lambda_1}$ , where  $L (= Kl)$  is the width of the antenna array. This creates a *sinc* in the spatial direction domain, i.e. the path directions get convolved with  $\frac{L}{\lambda_1} \text{sinc} \left( \frac{L\psi}{\lambda_1} \right)$ . Thus, if we represent the spatial profile after convolution of the sincs with  $P_1^n$  as  $P_2^n$ , then  $P_2^n$  is given by

$$P_2^n(\psi) = \{a_n e^{-j \frac{2\pi d_n}{\lambda_1} + j\phi_n} \delta(\psi - \psi_n)\} * \frac{L}{\lambda_1} \text{sinc} \left( \frac{L\psi}{\lambda_1} \right) \quad (6.5)$$

where  $*$  denotes convolution operation. Thus,  $P_2^n(\psi)$  refers to the graphs in Figs. 6-3(c) and (c').

- **Superposition:** In case of multiple paths, the perceived path profile is simply the sum of individual path profiles. Thus, the overall profile  $P_3(\psi)$ , can be computed as:

$$P_3(\psi) = \sum_{n=0}^{N-1} P_2^n(\psi). \quad (6.6)$$

This equation mathematically represents Fig. 6-3(d)-(d').

- **Discrete Fourier Transform:** Finally, the channel measurements at the antennas are just the Fourier transform of the signal arriving along spatial directions. In order to represent this mathematically, observe that equation 6.6 can be simplified as follows:

$$P_3(\psi) = \sum_{n=0}^{N-1} \{a_n e^{-j\frac{2\pi d_n}{\lambda_1} + j\phi_n} \delta(\psi - \psi_n)\} * \frac{L}{\lambda_1} \text{sinc}\left(\frac{L\psi}{\lambda_1}\right) \quad (6.7)$$

$$= \sum_{n=0}^{N-1} a_n e^{-j\frac{2\pi d_n}{\lambda_1} + j\phi_n} \times \frac{L}{\lambda_1} \text{sinc}\left(\frac{L(\psi - \psi_n)}{\lambda_1}\right) \quad (6.8)$$

Equation 6.7 follows from equation 6.8 by using the convolution property of the delta function.

The above four transformations can be summarized succinctly as a sequence of matrix operations. Specifically, given that the antennas are positioned at  $K$  discrete locations in space, we can now represent the Fourier transform by a matrix multiplication. Let us define  $\mathbf{F}$  to be the  $K \times K$  Fourier matrix, such that  $\mathbf{F}_{ij} = e^{-j\frac{2\pi i l j'}{\lambda_1}}$ , where  $\psi'$  defines the discretization on the variable  $\psi$  ( $\psi' = \frac{2}{K}$ ).<sup>4</sup> Further, define  $\mathbf{S}$  to be the  $K \times N$  matrix where  $\mathbf{S}_{ij}$  denote the value of the sinc function corresponding to the  $j^{\text{th}}$  path at  $\psi = i\psi'$ . Specifically,  $\mathbf{S}_{ij} = \frac{L}{\lambda_1} \text{sinc}\left(\frac{L(i\psi' - \psi_j)}{\lambda_1}\right)$ . Finally, define  $\vec{a}'_1$  to be the  $N$  dimensional vector such that the  $i^{\text{th}}$  component is  $a_i e^{-j\frac{2\pi d_i}{\lambda_1} + j\phi_i}$ . Then, the channel measurements at the antennas, represented by  $\vec{h}_1$  can be given by:

$$\vec{h}_1 = \mathbf{F}\mathbf{S}\vec{a}'_1 \quad (6.9)$$

Note that,  $\vec{h}_1$  is the  $K$  dimensional vector such that the  $k^{\text{th}}$  element represents the channel measurement at antenna  $k$ . Observe that, in the vector notation, the  $i^{\text{th}}$  component of  $\mathbf{S}\vec{a}'_1$  is nothing but  $P_3(i\psi')$ . In summary, we now have a transform that maps signal paths to channels.

## 6.4.2 From Wireless Channels to Paths

Now that we understand, how the channels are derived from the underlying physical paths, the goal is to find a way to invert this mechanism. In other words, given channel measurements,  $\vec{h}_1$  on wavelength  $\lambda_1$ ,

---

<sup>4</sup>When the antenna separation,  $l$ , is not equal to  $\frac{\lambda_1}{2}$ , the Fourier matrix is replaced by the non-uniform Fourier matrix and  $\psi' = \frac{\lambda}{L}$ , where  $L = Kl$  is the total antenna array aperture.

we need to identify the underlying physical paths. We do so by inverting the individual components of the transform – the Fourier Transform, windowing and super-position and phase variations.

**Inverting the Fourier Transform:** The first step is to invert the effect of the Fourier transform, which is simply the inverse Fourier transform on the channel measurements,  $\vec{h}_1$ . This can be achieved by multiplying  $\vec{h}_1$  by  $\mathbf{F}^{-1}$ .

**Inverting Windowing and Superposition:** Next, we need to invert the superposition effect, stated in equation 6.6 and the windowing effect from equation 6.5. These two effects are jointly represented by the matrix multiplication,  $\mathbf{S}\vec{a}'_1$  in equation 6.9. The goal is to infer  $\mathbf{S}$  and  $\vec{a}'_1$ , given the perceived signal paths,  $\mathbf{F}^{-1}\vec{h}_1$ . Observe that,  $\mathbf{S}$  depends solely on the directions of the underlying paths ( $\psi_n$ ). Thus, in order to compute  $\mathbf{S}$ , we need to find  $\{\psi_n\}_{n=0}^{N-1}$  for each of the  $N$  sinc functions that sum up to yield this profile. We pose this problem as an L-2 norm minimization problem. We optimize for  $\{a'_{1,n}\}_{n=0}^{N-1}$  and  $\{\psi_n\}_{n=0}^{N-1}$  such that  $\|\mathbf{F}^{-1}\vec{h}_1 - \mathbf{S}\vec{a}'_1\|^2$  is minimized. Let us write this objective function as:

$$O(\{a'_{1,n}, \psi_n\}_{n=0}^{N-1}) = \|\mathbf{F}^{-1}\vec{h}_1 - \mathbf{S}\vec{a}'_1\|^2 \quad (6.10)$$

where  $a'_{1,n}$  denotes the  $n^{\text{th}}$  element of  $\vec{a}'_1$

In order to simplify the problem, observe that, if we know  $\mathbf{S}$ , the optimization problem becomes a linear optimization problem and can be solved for  $\vec{a}'_1$  in the closed form. In particular, the minimum value can be attained by setting  $\vec{a}'_1 = \mathbf{S}^\dagger \mathbf{F}^{-1}\vec{h}_1$ , where  $\mathbf{S}^\dagger$  denotes the pseudo-inverse of  $\mathbf{S}$ .

Thus, the objective function in equation 6.10 can be re-framed as:

$$O(\{\psi_n\}_{n=0}^{N-1}) = \|\mathbf{F}^{-1}\vec{h}_1 - \mathbf{S}\mathbf{S}^\dagger \mathbf{F}^{-1}\vec{h}_1\|^2 \quad (6.11)$$

We have, now, reduced the problem to identifying the directions of the signal paths that contribute to the directional signal profile. This objective function, however, is non-linear and non-convex. We discuss in Sec. 6.4.3 how we find a solution to this optimization problem.

**Accounting for Phase Variation:** Finally, in order to infer channels at a different wavelength,  $\lambda'$ , we need to fit in another missing piece. Recall that the phase of  $a'_{1,n}$  inferred at wavelength,  $\lambda_1$  for each of the paths, is dependent on the wavelength (since  $a'_{1,n} = a_n e^{-j\frac{2\pi d_n}{\lambda_1} + j\phi_n}$ ). In order to infer the frequency-dependent component of  $a'_{1,n}$ , we leverage the fact that for cellular systems, the wireless signal is transmitted at

multiple frequencies, called the OFDM subcarriers. This gives us access to channel measurements on multiple frequencies. Thus, we add the distance  $d_n$  for each of the paths as a parameter of the optimization problem given in equation 6.10. This allows us to solve the optimization problem jointly for multiple subcarriers and adds constraints to the solutions returned by the optimization at different frequencies.

In particular, let us denote the channel measurements at wavelength,  $\lambda_i$ , by  $\vec{h}_i$ ,  $i = 0, 1, \dots, I - 1$ . We define  $\vec{P}_i = \mathbf{F}^{-1}\vec{h}_i$ , and  $\mathbf{S}_i$  to be the matrix  $\mathbf{S}$  corresponding to wavelength  $\lambda_i$ . Let,  $\mathbf{D}_i$  be the  $N \times N$  diagonal matrix such that  $\mathbf{D}_i(k, k) = e^{-j\frac{2\pi d_k}{\lambda_i}}$  and  $\vec{a}$  be the  $N$  dimensional vector such that  $i^{th}$  element is  $a_i e^{j\phi_i}$ . Let  $\vec{\mathcal{P}}$  denote the  $IK$  dimensional vector formed by the concatenation of the vectors  $\vec{P}_i$  and  $\mathbf{S}$  be the  $IK \times N$  matrix formed by the concatenation of the matrices  $\mathbf{S}_i \mathbf{D}_i$ . Specifically:

$$\vec{\mathcal{P}} = \begin{pmatrix} \vec{P}_1 \\ \vec{P}_2 \\ \cdot \\ \cdot \\ \vec{P}_K \end{pmatrix} \quad \mathcal{S} = \begin{pmatrix} \mathbf{S}_1 \mathbf{D}_1 \\ \mathbf{S}_2 \mathbf{D}_2 \\ \cdot \\ \cdot \\ \mathbf{S}_K \mathbf{D}_K \end{pmatrix} \quad (6.12)$$

Thus, the modified objective function can be written as:

$$O(\{\psi_n, d_n, a_n\}_{n=0}^{N-1}) = \|\vec{\mathcal{P}} - \mathcal{S}\vec{a}\|^2 \quad (6.13)$$

This objective function is similar to equation 6.10. Like before, we can replace  $\vec{a} = \mathcal{S}^\dagger \vec{\mathcal{P}}$ . Thus, the objective function reduces to:

$$O(\{\psi_n, d_n\}_{n=0}^{N-1}) = \|\vec{\mathcal{P}} - \mathcal{S}\mathcal{S}^\dagger \vec{\mathcal{P}}\|^2 \quad (6.14)$$

### 6.4.3 Solving the Optimization

In this section, we describe how we solve the optimization problem that transforms channels to paths. Our goal is to find the values of  $\{\psi_n, d_n\}_{n=0}^{N-1}$ , such that:

$$\begin{aligned} \{\psi_n, d_n\}_{n=0}^{N-1} &= \arg \min_{\psi_n, d_n} O(\{\psi_n, d_n\}_{n=0}^{N-1}) \\ \text{s.t.} \quad &-1 \leq \psi_n \leq 1 \quad \forall n \in \{0, 1, \dots, N-1\} \end{aligned} \quad (6.15)$$

This optimization problem is non-convex and constrained. In order to solve this optimization problem, we use the well-known interior-point method. However, since the function is non-convex, the optimization is prone to convergence to a local minimum, which is not the global minimum. Thus, a good initialization is important to ensure that the correct solution is determined.

- **Initialization:** R2-F2 computes an approximate solution in order to initialize the minimization of the objective function described in equation 6.14. We compute an approximate probability distribution,  $P$  such that  $P(d, \psi)$  indicates the probability of the existence of a path from direction  $\psi$  and distance  $d$ . A natural candidate to do so is the power of the inverse Fourier transform of the channel itself (akin to Fig. 6-3(c)-(c')), which while prone to the windowing and superposition effects provides an approximate understanding of where signal paths emerge from. Generalizing the inverse Fourier transform to operate across both distance and angle-of-arrival, we define  $P$  to be:

$$P(d, \psi) = \left\| \sum_{i=1, \dots, I; k=1, \dots, K} h_{i,k} e^{j \frac{2\pi(d+kl\psi)}{\lambda_i}} \right\|^2$$

where  $h_{i,k}$  denotes the channel measured at antenna  $k$  and wavelength  $\lambda_i$  and  $l$  is the inter-antenna separation on the antenna array. Once,  $P$  has been computed for different values of  $d$  and  $\theta$ , we pick the  $N$  largest peaks to initialize the optimization problem with  $N$  paths.

- **Stopping Criterion:** So far, we have assumed that we know the number of paths,  $N$ , a priori. However, that is not the case in practice. Notice that, as we increase the number of paths,  $N$ , in our objective function, the minimum value attained on the objective function decreases. In other words, the algorithm keeps finding a better fit. However, after certain number of paths, we start to overfit, i.e., the additional paths being found do not correspond to physical paths, but to signal noise. This could

lead to decrease in the accuracy of our channel estimation algorithm. In order to avoid overfitting and yet achieve a good fit, we incrementally add paths to the solution till one of the two conditions is met. Either, the value of the objective function drops below a threshold,<sup>5</sup> or decrease in the value of the objective function is small. When that happens, we select that value of  $N$  as the number of paths.

- **Conditioning:** When the number of paths,  $N$ , is greater than 1, the optimization can find solutions, such that  $(\psi_i, d_i)$  is very close to  $(\psi_j, d_j)$  for  $i \neq j$ , i.e. two paths come from nearly the same angle and distance. In that case, the matrix  $\mathcal{S}$  becomes ill-conditioned and can lead to poor solutions. In such cases, R2-F2 rejects one of these paths and reduces the number of paths by 1. This improves the condition number of the matrix and avoids overfitting.

## 6.5 Integrating R2-F2 with the LTE Architecture

This section describes R2-F2’s end-to-end system design, and how it interacts with the LTE protocol. R2-F2 takes as inputs wireless channels measured on the uplink at the base station for a particular user. It outputs the estimated wireless channels at the downlink frequency band for that user. These channels can then be used to perform beam-steering for advanced MIMO techniques (coherent beamforming, interference nulling, etc.).

The following steps summarize R2-F2’s approach: (1) R2-F2 runs an iterative algorithm to find a representation of signal paths that fit the observed uplink channels. This is done by solving the optimization in Eqn. 6.14 as described in Sec. 6.4.3. (2) R2-F2 use the recovered 4-tuple signal paths to map the uplink channels to the frequency used on the downlink channel (Eqn. 6.9). (3) Now that it has the values of the uplink channels for the downlink frequency, R2-F2 applies standard reciprocity [85] to infer the downlink channels.<sup>6</sup> Fig. 6-5 presents an overview of R2-F2’s architecture.

We next discuss a couple of issues that arise when integrating the above steps with LTE cellular systems.

---

<sup>5</sup>We set  $\epsilon$  to  $0.01 \times IK$ , where  $IK$  is the number of elements in  $h$ .

<sup>6</sup>Standard reciprocity infers the forward channels from the reverse channels by multiplying by calibrated reciprocity constants, which are computed once for the lifetime of the device as described in [85].

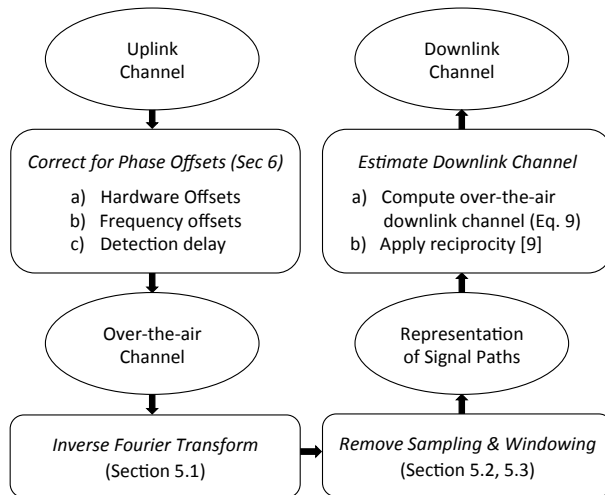


Figure 6-5: **R2-F2's System Design:** The above diagram summarizes the steps in R2-F2's system design in translating channels at the base station measured on the uplink to channels at the user on the downlink frequency bands.

### 6.5.1 Measuring the Uplink Channels

In R2-F2, the base station measures the uplink channels in order to infer the downlink channels. However, since LTE uses SC-FDMA on the uplink, the client transmits its data only on a subset of all OFDM subcarriers. Thus, while LTE uses one thousand subcarriers, only a subset of those sub-carriers can be used to measure the uplink channels for a particular client at any point in time. However, in LTE, a client does not only send data. It also sends signaling information. In particular, the LTE standard uses Sounding Reference Signals (SRS) [6] which the client sends periodically across OFDM subcarriers. The period of the SRS signal is configurable and takes values between 2ms and 320ms. Thus, R2-F2 measures the uplink channels using the SRS transmissions. It further refines these channels with measurements from uplink data and acknowledgments, which can provide new measurements every 1 ms, but span only a subset of the OFDM sub-carriers. Combining all these measurements allows R2-F2 to obtain a better estimate of the uplink channels, which naturally improves its inference of the downlink channels.

One however has to be careful when combining channel measurements that did not occur at exactly the same time. Measurements taken at different times can be affected by the carrier frequency offset (CFO) between the transmitter and receiver, frame detection delay as well as inherent delays in hardware. We discuss these effects and how we compensate for them before combining the channel measurements.

- **Frequency Offset:** Between any client and the base station, there always exists an offset in frequency

(CFO),  $\delta f$ . The CFO causes a phase rotation over time. Thus, two measurements of the same channel that are taken  $\tau$  seconds apart, exhibit a phase difference of  $2\pi\tau\delta f$ . This phase difference is a measurement artifact; the channels over the air have not changed. To overcome this issue, we leverage the fact that for all MIMO techniques (beamforming, nulling, interference alignment, etc.) the parameter of importance is not the exact value of the wireless channels, but the relative change in the channels across the antennas. As a result, a constant multiplied to the channel measurements on all antennas does not affect our ability to perform all MIMO techniques. Thus, since all antennas on the base station experience the same CFO, we can eliminate the phase rotation caused by CFO by dividing the channel of each base station antenna by the channel of the first antenna, measured at the same point in time. This division scales all channels by the same value and hence does not affect MIMO techniques.

- Frame Detection Delay:** There is a time delay between the moment the signal reaches the radio and the moment when it actually gets detected. This delay causes an additional phase rotation in the measured channels. If we denote this detection delay by  $t_d$ , the additional phase rotation is given by  $2\pi f t_d$ . The channel measurements on all the antennas get effected by this quantity. This detection delay varies per measurement and thus, makes it infeasible to combine information across different measurements without eliminating the phase rotation due to the detection delay. To overcome this challenge, observe that across OFDM subcarriers any delay in time manifests itself as rotation in phase across subcarrier frequency whose slope is exactly  $2\pi t_d$ . As a result, R2-F2 can enforce consistency between measurements of the wireless channel at any given frequency by ensuring they have zero relative slope in phase.
- Hardware Delay:** The hardware across different antennas introduces a fixed delay to the different receive chains. Thus, the wireless channel measured at each antenna suffers a phase rotation. In particular, the channel measured at an antenna with hardware delay  $t$  experiences an additional offset  $e^{-j\frac{2\pi ct}{\lambda}}$ , where  $\lambda$  is the wavelength of the signal and  $c$  is the speed of light. Further, this offset is different for different receive chains. Fortunately, hardware-induced phase offset is fixed for the lifetime of the base station and can be calibrated once, apriori and applied to all future channel measurements.

## 6.5.2 Inter-cell Interference

R2-F2's channel estimates can be used to improve overall throughput by reducing inter-cell interference



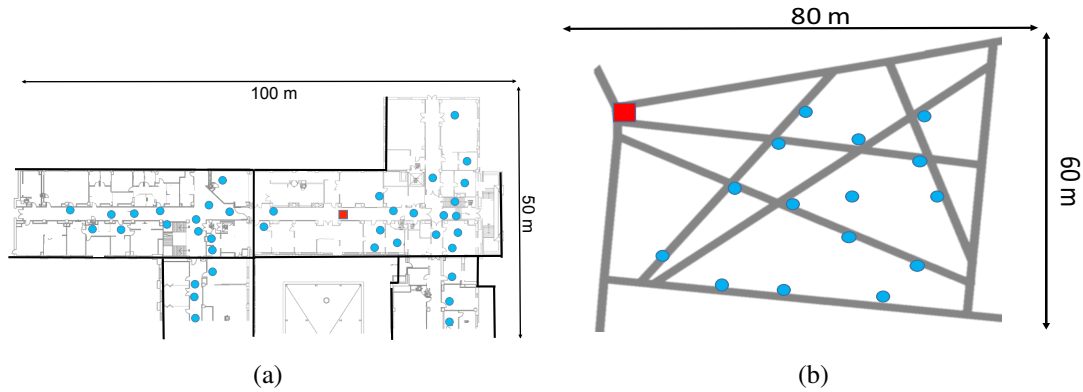


Figure 6-6: **Experiment Testbed:** (a) plots the indoor testbed for our experiments. Thick black lines indicate building boundaries. The floor map of individual buildings is marked inside each building. Experiments were conducted in the publicly accessible spaces across four buildings. (b) is the outdoor testbed with roads and pathways shown; buildings have been removed for anonymity. In (a) and (b), the red square marks the presence of the base station and the blue dots represent the clients.

at edge clients – devices that are close to cell boundaries and hence, receive signal from multiple base stations. R2-F2 adopts two strategies to limit the interference caused at edge clients by neighboring base stations: (1) It encourages sharing of information between adjacent base stations to predict uplink channels to users across base stations. This allows the base station to estimate the client channels and learn any potential interference from the channel estimates of its neighboring base stations. (2) Base stations can use the channel estimates they infer from R2-F2 to transmit to their clients while nulling interference to clients of other base stations that happen to be at cell edges. Indeed, the same wireless channels used for beamforming can be employed for interference nulling as well. Our results in Sec. 6.7 show that interference nulling at edge clients using R2-F2 leads to improved throughput in LTE networks.

## 6.6 Implementation

We implemented R2-F2 on a five-antenna Ettus USRP N210 software radio platform emulating a five-antenna LTE base station. The USRPs are synchronized using an external clock and act as a five-antenna MIMO node. They transmit using an OFDMA architecture that transmits 1024 subcarriers over 10 MHz, which are parameters identical to common LTE deployments of major US operators [6]. We implement LTE’s OFDMA by modifying the Ettus UHD driver for USRP software radios. Our software radios operate over the whitespace spectrum in frequency bands up to 10 MHz wide at center frequencies between 640-

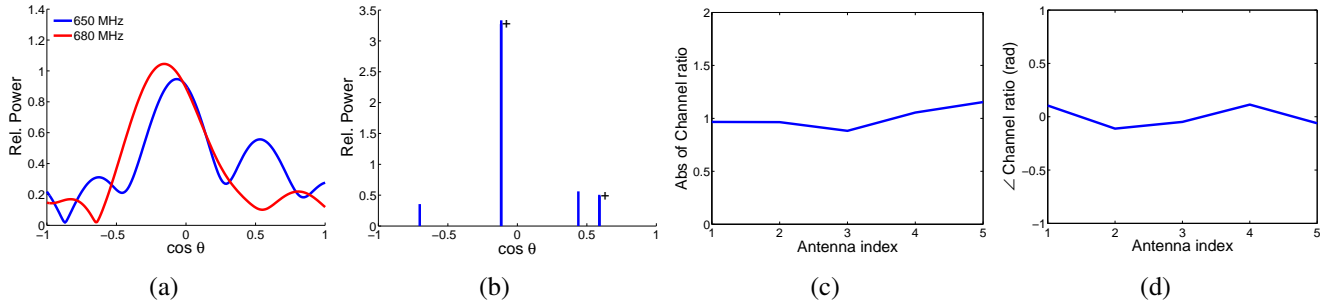


Figure 6-7: **Microbenchmark:** R2-F2 measures wireless channels on the uplink at 650 MHz and predicts the downlink channels on 680 MHz. The directional power profile for the uplink channel in a particular measurement is shown in (a). We also plot the downlink profile, obtained using ground truth measurements for reference. As explained in Sec. 6.3, these profiles appear very different. The paths inferred by R2-F2 are plotted in (b). A ‘+’ sign next to a path indicates presence of two paths being plotted as one due to the plotting resolution. R2-F2 uses these paths to predict channels on 680 MHz. The absolute value of the ratio of the estimated channels to the ground truth channels is plotted in (c), while (d) plots the phase of this ratio.

690 MHz. We note that these bands are just 20-30 MHz away from the frequency bands of commercial LTE deployments of the two major US cellular operators: AT&T and Verizon [54].

We use a single-antenna USRP software radio to emulate an LTE cellular phone. The USRP transmits using SC-FDMA on bandwidth up to 10 MHz on up to 1024 subcarriers, as per the LTE PHY standard [6]. Unless specified otherwise, the uplink and downlink bands are at center frequencies of 650 MHz and 680 MHz, separated by 30 MHz. We note that major US carriers separate uplink and downlink frequency bands by 30 MHz in commercial deployments.

We ensure that the characteristics of our software-radio based LTE testbed in the whitespaces closely mirror those of commercial LTE deployments. The clients transmit data only on 8% of the OFDM subcarriers, and transmit sounding reference signals once every two subframes (at most once every 2 ms, as in Verizon’s LTE network configuration). In all tests we co-locate our base station with one of the LTE base stations deployed on our campus. This ensures that the multipath effects experienced by R2F2 base station closely match the multipath observed by an actual LTE deployment.

**Evaluation:** We perform our experiments in both indoor and outdoor settings. We co-locate our base station next to an operational LTE base stations in both settings. The client is moved to various locations across multiple buildings indoors as well as in outdoor locations along the streets and between buildings. Fig. 6-6 plots our testbed with the location of the base station marked in red and client locations marked as blue dots. Our experiments are performed across multiple randomly chosen client locations and the

average and standard deviation of results are presented. The experiments are evenly split between outdoor and indoor locations. The experiments were conducted over three days and the weather was mostly clear, with a snow cover present throughout. Our results report both the measured signal-to-noise ratio (SNR) measured at the client as well as throughput (in Mb/s). We note that we measure throughput from SNR across LTE subcarriers using the effective SNR (ESNR) metric [86] with bit-rates adjusted based on the LTE standard [6, 238].

Note that, the delay introduced by USRP software radios (about 10 seconds) in switching frequencies limits our evaluation to static clients. While we leave an evaluation of R2-F2 for mobile clients to future work, we note that FPGA implementations of interior-point methods (like the one discussed in section 6.4) are fairly standard and converge in hundreds of microseconds [110]. This time is much lower than the few milli-seconds of channel coherence time for typical scenarios [216]. Finally, since R2-F2’s estimates rely on uplink channels from the client to the base station, rather than the last downlink channel from the base station to the client, R2-F2’s channel estimates are more recent than explicit channel feedback, thus aiding mobility.

## 6.7 Results

In this section, we present the results of an experimental evaluation of R2-F2.

### 6.7.1 Micro Benchmarks

We aim to check whether the model in Sec. 6.3 matches the empirical measurements. We conduct our experiments in the testbed described in Sec. 6.6. In each run, the base station transmits to the client, and the client computes the ground truth channels on the downlink. The client then transmits to the base station, and the base station computes the uplink channels. We run R2-F2 on the uplink channels to infer the downlink channels.

Fig. 6-7 plots the results from a representative run. Fig. 6-7(a) plots the Fourier transform of the channel measurements on uplink and downlink channels. The Fourier transform is plotted with a super resolution factor of 20, (i.e., the Fourier matrix has 5 columns that correspond to the 5 channels and 100 rows). The figure shows that the Fourier Transforms, and hence, the corresponding channels differ significantly from the downlink channels and their Fourier Transform, despite that the uplink and downlink

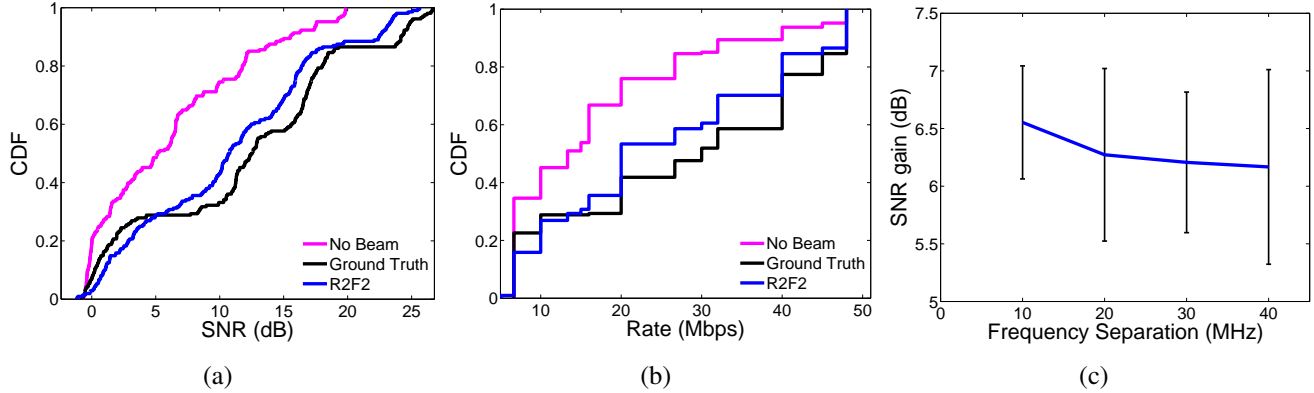


Figure 6-8: **Beamforming:** We use the channels estimated by R2-F2 to achieve beamforming towards the client. Figure (a) depicts the CDF of the SNR at the client without beamforming, using beamforming with the channels predicted by R2-F2 and using beamforming with the true channels measured at the client. R2-F2 achieves ~6 dB SNR gain over no beamforming, which is just 0.7 dB less than beamforming with ideal channels. Figure (b) depicts the datarates achieved by the different schemes. R2-F2 enables a median gain of 1.7x in datarate for clients in our testbed. Figure (c) depicts the median gain in SNR due to beamforming using channels estimated by R2-F2 as a function of frequency separation.

are separated by only 30MHz. Note that, the figure shows the uplink and downlinks for the same OFDM subcarrier on each frequency band.

R2-F2 uses the measured channels on the uplink to infer the underlying physical paths. The inferred paths are shown in Fig. 6-7(b). The R2-F2 algorithm infers 6 different paths (two sets of two paths are clustered together due to the plotting resolution and are marked by a '+' sign in the figure). The downlink channels inferred from these paths strongly match the ground truth channels measured at the client. The ratio of the downlink channels estimated by R2-F2 and the channels measured by the client is shown in Fig. 6-7(c) and Fig. 6-7(c). Notice that the absolute value of the ratio (Fig. 6-7(c)) is very close to 1. Moreover, the phase error in the channel ratio (Fig. 6-7(d)) is close to zero. Thus, this example shows that the model in Sec. 6.3 captures the RF propagation in the testbed.

## 6.7.2 Effectiveness of Beamforming

Beamforming is the key function underlying all MIMO solutions such as MU-MIMO, massive MIMO, etc. Thus, we would like to examine whether R2-F2 can deliver the same beamforming gain as ground truth channels.

As before, we run our experiments in the testbed in Fig. 6-6. We repeat the experiment for different

client locations, and for each client location, we collect 10 measurements. The clients were placed at distances of up to 75 meters from the base station. We measure the ground truth channels as before. We also measure the signal-to-noise ratio at the client for signals received from the base station across these experiments.

We compare the results for three different schemes: (1) Beamforming using the channels inferred by R2-F2; (2) Beamforming using the ground truth channels; and (3) Transmission in the absence of beamforming.

Fig. 6-8(a) depicts the CDF of the signal-to-noise ratio of these three schemes across experiments. This figure shows multiple interesting results. First, beamforming using R2-F2 provides almost the same SNR gains as beamforming using the ground truth channels. In fact, the average difference in the SNR of these two schemes is only 0.7dB. This demonstrates that R2-F2 can deliver accurate beamforming without any channel feedback, and using a completely passive channel estimation process.

Second, transmitting without beamforming reduces the SNR by an average of 6.5 dB. This result matches expectation since the theoretical gain of 5-antenna MIMO beamforming is  $10 \log_{10} 5 = 6.98dB$ . The gains are lower at low SNR –i.e., SNR less than 3 dB. This is because channel estimation at such low SNR does not work well. This is true for both the ground truth measurement at the client and the uplink measurements at the base station.

In order to evaluate the throughput improvement, we plot the data rates associated with the SNRs for all three schemes in Fig. 6-8(b). The figure shows that R2-F2 can double or tripple throughput in our testbed. The average throughput increase is 1.7x. The throughput gains are large at low to moderate SNRs but are less at higher SNR. This is expected since the rate is the log of the SNR. Also, at SNR more than 20dB, the highest data rate is achieved and beamforming doesn't help in increasing the rate. Similar to Fig. 6-8(a), the beamforming gains are low at SNR less than 3 dB. This is because at such low SNR, channel measurements become noisy, giving R2-F2 a noisy input.

### 6.7.3 Performance as a Function of Channel Separation

We study R2-F2's performance as a function of the separation between the uplink and downlink channels. We repeat our experiments by changing the separation between uplink and downlink frequency bands between 10 MHz and 40 MHz within the whitespace band of frequencies. Limitations of our white space

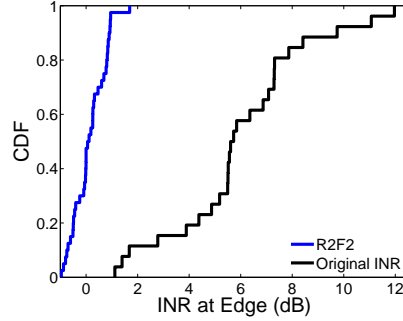


Figure 6-9: **Nulling interference at Edge Clients:** R2-F2 can reduce inter-cell interference by enabling the base station to null its signal to the clients at the cell edge. R2-F2 reduces the interference at the edge from a median of 5.5 dB to 0.2 dB and the 90th percentile from 9 dB to 0.9 dB.

license do not let us go beyond the 40 MHz separation. We measure R2-F2's SNR gain due to beamforming, for users at different randomly chosen locations in the testbed. Fig. 6-8(c) plots the mean and standard deviation of gain in SNR using R2-F2's beamforming, across different separations of uplink and downlink frequency bands. As expected, R2-F2's gain improves with a lower separation, with the highest gain achieved for a 10 MHz separation (6.55 dB). However, we observe that the SNR reduced very slowly with increase in downlink-uplink separation. Since the separation between the LTE downlink and uplink for most of the Verizon and AT&T deployments are 20MHz and 30MHz respectively, we believe that R2-F2 can be used to eliminate channel feedback in these networks. A potential cause of the degradation of the performance of R2-F2 with larger frequency separations is the variation in reflection properties of materials across frequencies, as observed in [166] in the context of GPS signals.

#### 6.7.4 Interference Nulling at Edge Clients

Clients at cell edge can suffer a significant amount of interference from neighboring cells which could amount to 10 to 12dB [210]. R2-F2 can be used to reduce interference at edge-clients located at cell boundaries using interference nulling. To evaluate this function, we set up our base station as in Fig. 6-6, but we move the client to the edge of the cell to emulate a client from a neighboring cell. We repeat the experiment from the previous section. However instead of using the inferred downlink channels to beamform, the base station uses the channels to perform interference nulling.

Fig. 6-9 plots a CDF of the interference power before and after nulling. The figure shows that R2-F2 dramatically reduces the interference at edge clients. In particular, the average INR (interference to noise ratio) is reduced from 5.5 dB to 0.2 dB, and the 90th percentile from 9 dB to 0.9 dB. This shows that

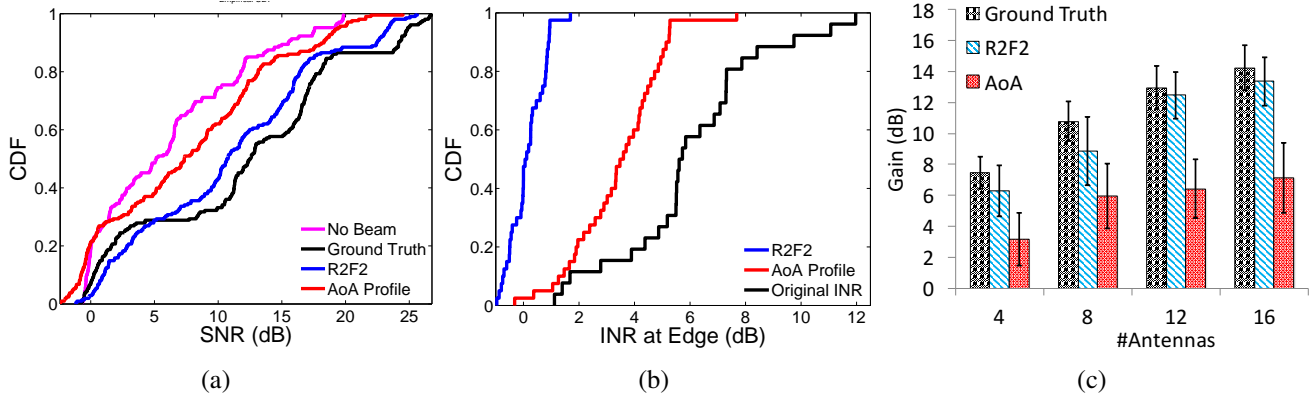


Figure 6-10: **Comparison with AoA Power Profile:** (a) AoA profile based channel estimation increases the median SNR of the testbed by 2.8 dB (as opposed to 6.3 dB for R2-F2). (b) Interference at the edge clients can be brought down from a median of 5.5 dB to 3.5 dB. However, R2-F2 outperforms this approach by nulling to 0.2 dB (median). (c) Simulation results show that with increase in number of antennas, the gain achieved by R2-F2 closely follows the ideal beamforming gain.

R2-F2 can be used beyond coherent beamforming, to counter inter-cell interference.

### 6.7.5 Comparison with Angle-of-Arrival Power Profile

At this stage, one might wonder if it is possible to achieve gains similar to R2-F2 by using the angle-of-arrival (AoA) power profiles similar to the ones shown in Fig. 6-2 and Fig. 6-3(d). In principle, one could use the measured wireless channels on one frequency to compute the AoA power profile using standard AoA equations. Then, this angle-of-arrival profile can be treated as a signature of the underlying physical propagation and can be used to compute the channels at the target frequency band. We conduct experiments on our testbed to evaluate this approach and compare the gains achieved by R2-F2 with the gains achieved by the AoA profile.

**Beamforming:** We compare the beamforming gain achieved by R2-F2 with the gains achieved with the AoA-based approach. The CDF of the signal to noise ratios achieved with the two approaches is compared in Fig. 6-10(a). While the AoA approach increases the median SNR of the testbed by 2.8 dB, the gain is much lower than R2-F2 which increases the SNR of the testbed by 6.3 dB. This is understandable, given the intuition developed in section 6.3 and 6.4. While the AoA power profiles of the signal have the same underlying paths, they are inherently dependent on frequency. Thus, using these profiles directly to

estimate channels across frequencies leads to errors in the estimation.

**Nulling at Edge Clients:** Similar to Sec. 6.7.4, we aim to null the interference caused by the base station at the edge clients. We use the channels estimated using the AoA approach to null the interfering signal at the client. The CDF of the INR (interference to noise ratio) after nulling is shown in Fig. 6-10(b). The median interference SNR is reduced from 5.5 dB to 3.5 dB. However, as expected, the errors in channel estimates prevent it from going down to the median SNR achieved by R2-F2 (0.2 dB).

**Variation with Number of Antennas:** Finally, it is natural to ask if, with the increase in the number of antennas, the sinc functions plotted in Fig. 6-3 (c), start to become narrower and thus, two sincs for different paths do not impact each other. This would lead to the AoA power profiles in Fig. 6-3 (d) to look identical at different frequencies and hence, improve the performance of this approach.

Since we have a 5 antenna base station, we are unable to test this hypothesis empirically. However, we test this hypothesis using a simulation. We simulate a wireless testbed of size  $200\ m \times 200\ m$ . Channels are modeled using equation the standard wireless propagation model (Eqn. 6.2 [216]). The signals from the base station to the client and vice-versa travel through the direct line of sight path and also after reflecting off up to 10 different obstacles along the way. The positions of the reflectors are randomly selected and so is the attenuation faced by each path. The number of antennas on the base station is varied from 4 to 16 in steps of 4. The frequency of operation is 650 MHz (uplink) and 680 MHz (downlink), as used in the rest of the results. Random additive white Gaussian noise is added in order to vary the SNR.

We plot the variation of the SNR gains achieved by different systems in Fig. 6-10(c). As the number of antennas increases, the ideal beamforming gain with the true channels increases as  $20\log(N)$  on average, where  $N$  is the number of antennas. R2-F2 keeps up with this increase in gain, as the number of antennas increases and stays within 1 dB of the ideal gain. With more antennas, R2-F2 has more information to estimate the path parameters underlying the wireless channels and hence, improves its estimates. On the other hand, the AoA profile based approach can only achieve half of the gain achieved by the beamforming with the true channel information, even as the number of antennas is increased to 16. This is because of two main reasons: a) as the number of antennas increases, the directionality of the antenna array increases and hence, a better estimation of channels is required to maintain the ideal gain and b) even if the sincs become narrower, as is expected, with increasing number of antennas, the phase of each of the sincs is still frequency dependent. When these sincs combine to give the channels, these phase add errors in the



channel estimates.

To conclude, this simulation provides two interesting insights. First, R2-F2 maintains its performance even as the number of antennas is increased. Second, it does not suffice to use AoA power profiles to infer wireless channels at the target frequency. One needs to go further and separate the individual paths and their phases to achieve accurate channel inference, as done by R2-F2.

## 6.8 Concluding Remarks

We present, R2-F2 a system that uses wireless channel measurements on one frequency band to infer channels on a different frequency band. By doing so, R2-F2 enables multi-antenna techniques to be used in LTE systems with *zero* feedback from the client. R2-F2's performance was demonstrated empirically with uplink and downlink channels separated by 20-30 MHz, as in the majority of LTE deployments in United States [225]. Beyond LTE, our work has implications on the general concept of reciprocity across frequency bands, for other wireless technologies such as wireless LANs and whitespace networks. However, we note that the relatively small separation of uplink and downlink frequencies in LTE ensures that reflection properties of objects in the environment and the divergence between physical propagation characteristics is limited. We believe extending the results presented by us to technologies that require significantly larger frequency separations is non-trivial and falls in the realm of future work.

# Chapter 7

## Conclusion

This thesis takes an end-to-end approach to Internet-of-things system design. As opposed to the traditional approach where new primitives are developed within each layer of the computing stack, our approach uses insights across the computing stack. Such cross-layer design has enabled us to develop new algorithms and system designs across a diverse range of settings: in-body, homes, and farms. Specifically, we developed the following systems as parts of this dissertation:

- **In-body Backscatter Communication and Localization:** We present ReMix, the first system to enable (far-field) deep-tissue communication and localization for in-body devices. In designing ReMix, we make the observation that circuit non-linearities, a hardware property that has been traditionally undesirable, can be leveraged to remove unwanted signal reflections caused by the human body and focus on tiny signals being reflected off in-body devices. We also build a new algorithm to localize in-body devices to centimeter-scale accuracy using the signals reflected off the in-body devices.
- **IoT Platform for Data-driven Agriculture:** Data-driven agriculture has been long-known to reduce agricultural costs, improve yields, and minimize environmental impact. Yet, the adoption of data-driven agriculture remains limited due to the challenges associated with data collection in remote rural farms. Specifically, farms have scarce sources of power and connectivity, and can only do limited sensor deployments due to physical and economical constraints. We build FarmBeats to overcome these challenges. Underlying FarmBeats is a new system design that combines information across multiple sensing modalities to reduce sensor deployment and corresponding data communication requirements. We validated these methods using long-term deployments in multiple farms with different size and

weather conditions.

- **Context-aware Smart Homes:** IoT deployments in smart homes require contextual information to react organically to the habits, needs, and behaviors of the occupants. We build two systems, Chronos and Duet, that provide such contextual information (specifically, the location and identity of occupants). In Chronos, we build the first system to enable accurate positioning of Wi-Fi devices using just one Wi-Fi access point. Chronos achieves this by enabling decimeter-level distance measurements using off-the-shelf Wi-Fi cards. Furthermore, we build a logical reasoning framework, Duet, that can augment Chronos and infer continuous location and identity of individuals, even when they are not carrying their Wi-Fi devices all the time. Duet formalizes the constraints of the physical world like people's inability to be present at two locations simultaneously. It uses this formal specification to deliver accurate location and identity in spite of error-prone and intermittent underlying data.
- **Next Generation Cellular Communications:** We demonstrate that the techniques built in this thesis have wider applicability beyond IoT system design. Specifically, we build R2-F2, a system to infer wireless channels on one frequency bands from channels on a different frequency band. This system eliminates the requirement of channel feedback in multi-antenna communications, thereby improving spectrum utilization as well as reducing the power burden on tiny IoT devices that would otherwise have to spend energy on transmitting wireless channels as feedback.

## 7.1 Lessons Learnt

In addition to the research contributions outlined above, the work done for this thesis has revealed interesting lessons for IoT system design. We summarize them below:

- **Multi-modality enables New Design Choices:** Traditional systems focus on making individual modes of sensing more robust and more capable, often at the cost of additional power and compute utilization. This thesis takes a different approach. It combines multiple error-prone sensing modes to develop robust end-to-end systems. This not only allows IoT systems to be reliable, but also exposes new tradeoffs in the system design. For instance, in Duet, we use two modes of indoor positioning: a system that uses radio signals transmitted by Wi-Fi devices and another system that uses radio signal reflections from human bodies. Both systems have different failure modes. Device-based systems fail

to identify a user's location when they are not carrying a device. In contrast, while reflection-based systems can locate users without needing them to carry a device, they cannot identify users. Duet fuses together the best of these two systems and is able to deliver both location and identity even when the users react with their device intermittently. Similarly, in FarmBeats, visual data from drones and ground sensor data are used together to enable farmers to collect dense data about their fields from sparse sensor deployments.

- **Exposing the Physical World to the Computing Stack:** In traditional IoT system design, information about the physical world (like the temperature, weather, environmental variables, etc.) is treated as yet another byte of data that needs to be passed up to the application layer to be processed, displayed, and stored. We advocate that this information should be exposed to and integrated into each layer of the computing stack, thereby making it possible for devices, protocols, and algorithms to react to change in surroundings. For instance, in FarmBeats, we use weather forecasts to estimate the amount of power that will be generated by a solar panel. This power budget is then apportioned to the sensing task in a way that maximizes data freshness. This strategy of incorporating weather forecasts in power cycling as well as the communication strategy ensures that farmers maximize their access to sensor data in the farm, even in harsh weather conditions. Similarly, R2-F2 uses information about the paths travelled by a wireless signal in the physical world to infer wireless channels across frequency bands, thereby enabling low-overhead communication for IoT devices.

## 7.2 Future Directions

Our research has focussed on building IoT systems to solve practical real-world problems. Going forward, IoT systems will form the core of three emerging areas in computing. First, questions of sensing, communication, and inference are integral to the promise of digital healthcare. These digital healthcare systems will glean information from wearables, in-body devices, radio signal reflections, and medical records to reduce risk of health emergencies, ensure better treatment procedures, decrease long-term healthcare costs, and generally produce better healthcare outcomes. Second, multi-modal inference mechanisms will form a core part of the next generation wireless and IoT systems, as diverse sensors continue to get deployed in our physical environments. These inference mechanisms will build on sensing using different modes and combine them with inference tools like computer logic, and machine learning. Finally, underlining all of

this will be the questions of security, privacy, and integrity of the data being collected using these systems. This becomes even more relevant with sensitive information like health data, sensor data from our homes, etc. We expand on these research directions below:

- **Digital Healthcare:** Digital healthcare entails continuous monitoring of an individual's health, and ensuring timely interventions. IoT systems can enable the push in this space along two directions. First, our work in Duet has shown how we can track people in-home without requiring changes to user behavior. Extracting more insights from this data opens up several possibilities for monitoring the daily wellbeing of individuals, disease progression, hospitalization risk, and ensuring compliance with drug routines. For example, inferring activities of daily living could allow us to prevent hospitalization by enabling early interventions in patients with chronic conditions like CHF (congestive heart failures). Second, we envision that future work will design in-body sensor networks that last for years and provide constant health monitoring. Our work on in-body communication and localization has already taken the first step in this direction. Building on this work to design medium access protocols for multiple sensors, zero-power sensing mechanisms, and exploring the interaction of radio waves with human bodies can have widespread impact in healthcare and medicine. For instance, zero-power sensing of pH can provide long-term continuous monitoring for patients with acid reflux and micro-sensors embedded in pills could ensure drug adherence, a cause of 125,000 preventable deaths every year.
- **Multi-modal Sensing on Smartphones:** Over the next few years, commercial devices like our smartphones will become equipped with radio devices capable of transmitting at vastly different frequency ranges. Near field communication (NFC) operates at sub-100 MHz, Bluetooth at 2.4 GHz, Wi-Fi at 5 GHz, and millimeter wave (mmWave) networks at 24 and 80 GHz. These RF frequencies, combined with pre-existing sensors like cameras and microphones, will enable multiple modes of sensing the environment. However, all these modes have very different forms of interacting with the environment, and extracting meaningful insights from these complex interactions is a challenging task. Fortunately, tools in machine learning, particularly deep learning, have recently had success in analyzing complex data. We believe exploring this confluence between multiple sensing modes and machine learning can open up several interesting capabilities for our smartphones, ranging from scanning the environment for Virtual Reality or Augmented Reality applications, to looking inside closed boxes for security, to

in-home medical imaging, to material identification, to evaluating food quality.

- **Security and Privacy in Smart Environments:** The advent of IoT and cyber-physical systems have forced us to change our existing notions of security and privacy. Security and privacy risks are not limited to digital information that is stored online but encompass our physical identities and environments. IoT systems have the capability to sense information about us and our surroundings and can change our physical environments by controlling devices and objects. As a result, there is an urgent need to rethink our security and privacy solutions. We believe the capability of IoT systems to sense our physical environment presents us an opportunity to rethink our security solutions by incorporating the physical information itself into our security approach. Specifically, the physical information being sensed and the communication channel will form signatures for verifying next-generation IoT devices. Future work will build on this intuition to integrate low-level sensing mechanisms with edge architectures, inference mechanisms, and security protocols to design end-to-end security and privacy mechanisms for IoT systems.

# Appendix A

## ReMix: Lemma for Localization Algorithm

**Lemma:** If an EM wave travels through  $L$  parallel layers, each with thickness,  $l_i$ ,  $i = 1, \dots, L$ ; then the phase accumulated by the EM wave is independent of the order of the layers.

**Proof:** Consider a 2-D system where a wave propagates in the  $XY$  plane (similar to Fig. 2-5) from  $X_T = (x_T, y_T)$  to  $x_R = (x_R, y_R)$ . Furthermore, without loss of generality, assume that the signal travels through  $L$  layers, stacked on top of each other along the  $Y$  axis. For this proof, we leverage the concept of wave vectors. Wave vector,  $\vec{k} = (k_x, k_y)$ , is a vector that points in the direction of wave propagation and its magnitude is given by,  $k_x^2 + k_y^2 = (2\pi \frac{f}{v})^2$ , where  $v = c/\sqrt{\epsilon_r \mu_r}$  is the speed of light in the material. Further, if the wave propagation path is denoted as another vector,  $\vec{r} = (r_x, r_y)$ , then the phase accumulated by the wave in travelling through  $\vec{r}$  is:  $Re(\vec{k} \cdot \vec{r})$ .

For layer  $i$ , let the wave vector be  $\vec{k}_i = (k_{xi}, k_{yi})$ , the propagation path be  $\vec{r}_i = (r_{xi}, r_{yi})$ , and the EM parameters be  $(\epsilon_{ri}, \mu_{ri})$ . Then, the phase accumulated by the wave is,  $\phi = \sum_{i=1}^L Re(\vec{k}_i \cdot \vec{r}_i)$ . Now, observe  $r_{yi} = l_i$ , since the total path along the  $Y$ -axis in each layer is the depth of the layer. Also, phase continuity at interfaces ensures that,  $k_{xi} = k_{xj} = k_x$  for all  $(i, j)$  [172]. Finally,  $\sum_{i=1}^L r_{xi} = x_R - x_T$ , since we are

considering the phase for propagation from transmitter to receiver. Combining, we have:

$$\phi = \sum_{i=1}^L Re(k_x)r_{xi} + Re(k_y)l_i \quad (\text{A.1})$$

$$= Re(k_x) \sum_{i=1}^L r_{xi} + \sum_{i=1}^L (2\pi \frac{f}{c})^2 \epsilon_{ri} \mu_{ri} l_i \quad (\text{A.2})$$

$$= Re(k_x)(x_R - x_T) + \sum_{i=1}^L (2\pi \frac{f}{c})^2 \epsilon_{ri} \mu_{ri} l_i \quad (\text{A.3})$$

Notice that Eq. A.3 just depends on the depth of each layer and is independent of the order. This implies that the phase accumulated by an EM wave in traversing parallel stacked layers is independent of the order of stacked layers.



# Appendix B

## Duet's Algorithm to Check State Validity

---

### 3 Algorithm to Check State Validity

---

▷ Given: State,  $C = \{v_i\}_{i=1}^K$

▷ Preprocessing Step: Remove variables (and corresponding constraints) that have  $\mathcal{U} \subset v.P$  as they are always satisfiable

▷ Output: True if the state is valid, false otherwise

**function** CHECK\_VALIDITY( $C$ )

**for**  $i = 1, \dots, K$  **do**

**if**  $|v_i.P \setminus v_i.I| == 0$  **then**

      return false

**else**

**for**  $j = 1, \dots, |v_i.P \setminus v_i.I|$  **do**

$v'.P = (v_i.P \setminus v_i.I)_j, v'.I = [], v'.R = v_i.R$ , where  $(X)_j$  is the  $j^{th}$  element in set  $X$

$x_i = (v_i.P \setminus v_i.I)_j$

**for**  $k = i + 1, \dots, K$  **do**

$v_k.I = v_k.I \cup \{(v_i.P \setminus v_i.I)_j\}$

**end for**

$C_1 = \bigcup_{l=i+1}^K v_l$

**if** check\_validity( $C_1$ ) **then** return true **end if**

**end for**

      return false

**end if**

**end for**

**end function**

---

The algorithm to check the validity of states in Duet's probabilistic reasoning model is described in algorithm 3. We prove the correctness of Algorithm 3 below. Specifically, we prove that the algorithm returns "true" if and only if the formula in Eqn. 5.6 is satisfiable.

We will prove this by the principle of mathematical induction. For the base case, consider the case of

a state that has a single variable,  $v$ . In this case, if the algorithm returns true, there exist at least one value in  $v.P \setminus v.I$  that can be assigned to  $v$ . This implies that the state is valid or the corresponding first-order formula is satisfiable. Similarly, if the state is valid, there exists at least one value that can be assigned to the variable,  $v$ . This implies  $v.P \setminus v.I$  is non-empty. It is trivial to see that the algorithm will return true in this case.

Now, let us assume for all  $k < K$  variables in a state, the algorithm returns true if and only if the corresponding first-order formula is satisfiable. If we prove that this assumption implies that the statement holds for  $k = K$ , then by the principle of mathematical induction, the statement must be true for all integer values of  $k$ , which is what we aim to prove. First, let us assume that the formula is satisfiable, i.e. there exist constants  $p_1, p_2, \dots, p_K$ , such that setting  $x_i = p_i$  makes the formula true. Thus, for all values of  $i$ ,  $p_i \in v_i.P \setminus v_i.I$ . Since,  $p_1 \in v_1.P \setminus v_1.I$ , the algorithm must try  $x_1 = p_1$ . When the algorithm tries  $x_1 = p_1$ , it makes a recursive call to the algorithm with an updated state,  $C_1$ , where each variable has ruled out  $p_1$ . Clearly, the assignment  $x_i = p_i$  satisfies the updated formula for  $C_1$ , since each  $p_i$  is unique. Since the formula for  $C_1$  is satisfiable and has  $k - 1$  variables, the algorithm must return true on this recursive call (using our assumption). Thus, if the formula is satisfiable, the algorithm returns true.

To prove the converse, let us assume that the algorithm returned true for a state,  $C$ , with  $K$  variables. We need to prove that the formula corresponding to state  $C$  is satisfiable. To return true for a state of size  $K > 1$ , for at least one value of  $x_1 \in v_1.P \setminus v_1.I$  (say  $p_1$ ), the recursive call to the algorithm returned true for the corresponding updated state  $C_1$ . Since,  $C_1$  has less than  $K$  variables, this implies that there exists an assignment of variables which satisfies the formula corresponding to  $C_1$ . Finally, none of those variables can be assigned  $p_1$  because it was removed as a possibility by the algorithm before the recursive call was performed. Thus, there exist a valid assignment to variables for the formula corresponding to  $C$ . Using principles of mathematical induction, this proves that the algorithm returns true if and only if the state is valid.

# Bibliography

- [1] Bosch Laser Distance Measurer GLM50. <http://www.boschtools.com/Products/Tools/Pages/BoschProductDetail.aspx?pid=GLM%2050>.
- [2] IEEE 802.11af: <https://standards.ieee.org/findstds/standard/802.11af-2013.html>.
- [3] VICON T-Series. <http://www.vicon.com/products/documents/Tseries.pdf>.
- [4] IEEE 802.11n-2009 Standard. 2009. <http://standards.ieee.org/findstds/standard/802.11n-2009.html>.
- [5] 3rd Generation Partnership Project. Evolved Universal Terrestrial Radio Access (E-UTRA), Physical Layer Procedures (Release 8), 3GPP TS 36.213, v8.8.0. Oct 2009.
- [6] 3rd Generation Partnership Project. Evolved Universal Terrestrial Radio Access (E-UTRA), Physical Channels and Modulation (Release 8), 3GPP TS 36.211, v8.9.0. Jan 2010.
- [7] 3rd Generation Partnership Project. Evolved Universal Terrestrial Radio Access (E-UTRA), Physical Channels and Modulation (Release 8), 3GPP TS 36.211, v8.9.0. Jan 2010.
- [8] A. M. Abbosh A. T. Mobashsher. Artificial human phantoms: Human proxy in testing microwave apparatus that have electromagnetic interaction with the human body. *ArXiv*, 2015.
- [9] Omid Abari, Deepak Vasisht, and Dina Katabi. Caraoke: An E-Toll Transponder Network for Smart Cities. ACM SIGCOMM, 2015.
- [10] Heba Abdelnasser, Reham Mohamed, Ahmed Elgohary, Moustafa Farid Alzantot, He Wang, Souvik Sen, Romit Roy Choudhury, and Moustafa Youssef. SemanticSLAM: Using Environment Landmarks for Unsupervised Indoor Localization. Transactions on Mobile Computing, 2016.
- [11] Abubakar Abid, Jonathan M. O'Brien, Taylor Bensef, Cody Cleveland, Lucas Booth, Brian R. Smith, Robert Langer, and Giovanni Traverso. Wireless power transfer to millimeter-sized gastrointestinal electronics validated in a swine model. *Nature Scientific Reports*, 2017.
- [12] Adaptrum. <http://www.adaptrum.com>.
- [13] Fadel Adib, Chen-Yu Hsu, Hongzi Mao, Dina Katabi, and Fredo Durand. RF-Capture: Capturing the Human Figure Through a Wall . ACM SIGGRAPH Asia, 2015.
- [14] Fadel Adib, Zach Kabelac, Dina Katabi, and Robert C. Miller. 3D Tracking via Body Radio Reflections. USENIX NSDI, 2014.

- [15] Fadel Adib, Zachary Kabelac, and Dina Katabi. Multi-person Localization via RF Body Reflections. *USENIX NSDI*, 2015.
- [16] Imad Afyouni, Cyril Ray, and Christophe Claramunt. Spatial Models for Context-Aware Indoor Navigation Systems: A Survey. *JOSIS*, 2012.
- [17] Ian F Akyildiz, David M Gutierrez-Estevez, and Elias Chavarria Reyes. The evolution to 4g cellular systems: Lte-advanced. *Elsevier Physical Communication*, 2010.
- [18] Mohammed H. Almarshadi and Saleh M. Ismail. Effects of Precision Irrigation on Productivity and Water Use Efficiency of Alfalfa under Different Irrigation Methods in Arid Climates. *Journal of Applied Sciences Research*, 2011.
- [19] American Society for Gastrointestinal Endoscopy. Wireless capsule endoscopy, 2013. <https://www.asge.org/docs/default-source/importfiles/assets/0/73730/c4d44578-c3d0-4583-9949-b15f3e8537e0.pdf?sfvrsn=4>.
- [20] Automated Home. Apple iBeacons Explained: Smart Home Occupancy Sensing Solved?, 2013.
- [21] AutoPano. kolor.com.
- [22] Syed Mahfuzul Aziz, Milos Grcic, and Tharshan Vaithianathan. *A Real-Time Tracking System for an Endoscopic Capsule using Multiple Magnetic Sensors*. Springer Berlin Heidelberg, 2008.
- [23] Martin Azizyan, Ionut Constandache, and Romit Roy Choudhury. Surroundsense: Mobile phone localization via ambience fingerprinting. *ACM MobiCom*, 2009.
- [24] Francis Bach, Rodolphe Jenatton, Julien Mairal, and Guillaume Obozinski. *Convex Optimization with Sparsity-Inducing Norms*, 2011.
- [25] Sonali Bagchi and Sanjit K. Mitra. *The Nonuniform Discrete Fourier Transform and Its Applications in Signal Processing*. 1999.
- [26] A. Baggio. Wireless sensor networks in precision agriculture. *ACM Workshop on Real-World Wireless Sensor Networks*, 2005.
- [27] P. Bahl and V.N. Padmanabhan. RADAR: An in-building RF-based User Location and Tracking System . *IEEE INFOCOM*, 2000.
- [28] Paramvir Bahl, Ranveer Chandra, Thomas Moscibroda, Rohan Murty, and Matt Welsh. White Space Networking with Wi-Fi like Connectivity. *ACM SIGCOMM Computer Communication Review*, 2009.
- [29] Waheed U. Bajwa, Jarvis Haupt, Akbar Sayeed, and Robert Nowak. Compressed Channel Sensing: A New approach to Estimating Sparse Multipath Channels. *Proceedings of the IEEE*, 2010.
- [30] Waheed U Bajwa, Jarvis Haupt, Akbar M Sayeed, and Robert Nowak. Compressed channel sensing: A new approach to estimating sparse multipath channels. *Proceedings of the IEEE*, 2010.

- [31] Paolo Barsocchi, Stefano Chessa, Erina Ferro, Francesco Furfari, and Francesco Potorti. Context Driven Enhancement of RSS-based Localization Systems. ISCC, 2011.
- [32] M. R. Basar, F. Malek, Khairudi M. Juni, M. Shaharom Idris, and M. Iskandar M. Saleh. Ingestible wireless capsule technology: A review of development and future indication. *International Journal of Antennas and Propagation*, 2012.
- [33] Ben Popper. The Drone You Should Buy Right Now, 2014. <http://www.theverge.com/2014/7/31/5954891/best-drone-you-can-buy>.
- [34] Dinesh Bharadia, Kiran Raj Joshi, Manikanta Kotaru, and Sachin Katti. BackFi: High Throughput WiFi Backscatter. ACM SIGCOMM, 2015.
- [35] James Brooks. Swedish workers implanted with microchips to replace cash cards and id passes. Independent UK, 2017.
- [36] Matthew Brown and David G. Lowe. Automatic Panoramic Image Stitching Using Invariant Features. *International Journal of Computer Vision*, 2007.
- [37] Yuriy Brun, Giovanna Marzo Serugendo, Cristina Gacek, Holger Giese, Holger Kienle, Marin Litoiu, Hausi Müller, Mauro Pezzè, and Mary Shaw. *Software Engineering for Self-Adaptive Systems*. 2009.
- [38] A. Canavitsas, L.A.R.S. Mello, and M. Grivet. White space prediction technique for cognitive radio applications. In *Microwave Optoelectronics Conference (IMOC), 2013 SBMO/IEEE MTT-S International*, 2013.
- [39] Weifeng Cao and Wenbo Wang. A frequency-domain channel prediction algorithm in wideband wireless communication systems. In *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2004.
- [40] Kenneth G. Cassman. Ecological Intensification of Cereal Production Systems: Yield Potential, Soil Quality, and Precision Agriculture. *Proceedings of the National Academy of Sciences (PNAS)*, 1999.
- [41] Rohit Chandra, Anders J. Johansson, and Fredrik Tufvesson. Localization of an rf source inside the human body for wireless capsule endoscopy. *BodyNets*, 2013.
- [42] X. Chen, X. Zhang, L. Zhang, X. Li, N. Qi, H. Jiang, and Z. Wang. A wireless capsule endoscope system with low-power controlling and processing asic. *IEEE Transactions on Biomedical Circuits and Systems*, 2009.
- [43] Krishna Chintalapudi, Anand Padmanabha Iyer, and Venkata N. Padmanabhan. Indoor Localization without the Pain. ACM MobiCom, 2010.
- [44] Junil Choi, David J Love, and Patrick Bidigare. Downlink training techniques for fdd massive mimo systems: Open-loop and closed-loop training with memory. *IEEE Journal of Selected Topics in Signal Processing*, 2014.

- [45] Ji chun Zhao, J. Zhang, Y. Feng, and J. Guo. The Study and Application of the IOT Technology in Agriculture. *IEEE International Conference on Computer Science and Information Technology*, 2010.
- [46] B. G. Colpitts and G. Boiteau. Harmonic radar transceiver design: miniature tags for insect tracking. *IEEE Transactions on Antennas and Propagation*, 2004.
- [47] Wikipedia contributors. Eb/N0 — Wikipedia, The Free Encyclopedia, 2017. <https://en.wikipedia.org/w/index.php?title=Eb/N0&oldid=809750730>.
- [48] Wikipedia contributors. Magnetic dipole — wikipedia, the free encyclopedia, 2017. [https://en.wikipedia.org/w/index.php?title=Magnetic\\_dipole&oldid=811519977](https://en.wikipedia.org/w/index.php?title=Magnetic_dipole&oldid=811519977).
- [49] Jason R. Cook, Richard R. Bouchard, and Stanislav Y. Emelianov. Tissue-mimicking phantoms for photoacoustic and ultrasonic imaging. *Biomedical Optics Express*, 2011.
- [50] R. Crepaldi, Jeongkeun Lee, R. Etkin, Sung-Ju Lee, and R. Kravets. CSI-SF: Estimating wireless channel state using CSI sampling & fusion. *IEEE INFOCOM*, 2012.
- [51] DataMapper. <http://www.precisionhawk.com/datamapperinflight>.
- [52] Amy Berrington de González and Sarah Darby. Risk of cancer from diagnostic x-rays: estimates for the uk and 14 other countries. *The Lancet*, 2004.
- [53] Decagon Devices. Decagon Devices Cellular Logger. <https://www.decagon.com/en/data-loggers-main/data-loggers/em50g-wireless-cellular-data-logger>.
- [54] Paul Denisowski. Recognizing and resolving lte/catv interference issues. *White Paper, Rohde and Schwarz*, 2011.
- [55] Anind K. Dey, Katarzyna Wac, Denzil Ferreira, Kevin Tassini, Jin-Hyuk Hong, and Julian Ramos. Getting Closer: An Empirical Investigation of the Proximity of User to Their Smart Phones. *Ubi-Comp*, 2011.
- [56] I. Dietlicher, M. Casiraghi, C. Ares, A. Bolsi, D. Weber, A. Lomax, and F. Albertini. Experimental measurement with an anthropomorphic phantom of the proton dose distribution in the presence of metal implants. *PTCOG*, 2014.
- [57] C. Ding, D. Pei, and A. Salomaa. *Chinese Remainder Theorem: Applications in Computing, Coding, Cryptography*. 1996.
- [58] DJI. <http://developer.dji.com>.
- [59] J. Doerflinger and T. Gross. Sustainable ICT in Agricultural Value Chains. *IT Professional*, 2012.
- [60] Thomas A. Doerge. In *International Plant Nutritional Institute*, 1999.
- [61] Liang Dong, Guanghan Xu, and Hao Ling. Prediction of fast fading mobile radio channels in wideband communication systems. In *IEEE Global Telecommunications Conference*, 2001.

- [62] Ilka Dove. Analysis of radio propagation inside the human body for in-body localization purposes. Master's thesis, University of Twente, 2014.
- [63] DroneDeploy. [dronedeploy.com](http://dronedeploy.com).
- [64] A. Dutt and V. Rokhlin. Fast Fourier Transforms for Nonequispaced Data. *SIAM J. Sci. Comput.*, 1993.
- [65] Ayman ElNashar, Mohamed El-saidny, and Mahmoud Sherif. *Design, Deployment and Performance of 4G-LTE Networks: A Practical Approach*. John Wiley & Sons, 2014.
- [66] Ettus Research. USRP X310. <https://www.ettus.com/product/details/X310-KIT>.
- [67] FCC. FCC Publication 703867, 2017. <https://apps.fcc.gov/oetcf/kdb/forms/FTSSearchResultPage.cfm?id=27023&switch=P>.
- [68] Federal Communications Commission. <https://www.fcc.gov/general/white-space-database-administration>.
- [69] Denzil Ferreira, Jorge Goncalves, Vassilis Kostakos, Louise Barkhuus, and Anind K. Dey. Contextual Experience Sampling of Mobile Application Micro-usage. *MobileHCI*, 2014.
- [70] Andrey Finkelstein, Ron Biton, Rami Puzis, and Asaf Shabtai. Classification of Smartphone Users Using Internet Traffic. *CoRR Arxiv*, 2017.
- [71] G. D. Forney. The Viterbi Algorithm. *Proceedings of the IEEE*, 1973.
- [72] K. R. Foster and J. Jaeger. Rfid inside. *IEEE Spectrum*, 2007.
- [73] C. D. Franco and G. Buttazzo. Energy-aware coverage path planning of uavs. In *International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, , 2015.
- [74] FreeWave. <http://www.freewave.com>.
- [75] Antonio-Javier Garcia-Sanchez, Felipe Garcia-Sanchez, and Joan Garcia-Haro. Wireless Sensor Network Deployment for Integrating Video-surveillance and Data-monitoring in Precision Agriculture over Distributed Crops. *Computers and Electronics in Agriculture*, 2011.
- [76] Sinan Gezici, Zhi Tian, Georgios B. Giannakis, Hisashi Kobayashi, Andreas F. Molisch, Vincent Poor, Zafer Sahinoglu, Sinan Gezici, Zhi Tian, Georgios B. Giannakis, Hisashi Kobayashi, Andreas F. Molisch, H. Vincent Poor, and Zafer Sahinoglu. Localization via Ultra-wideband Radios. In *IEEE Signal Processing Magazine*, 2005.
- [77] Rizwan Ghaffar and Raymond Knopp. Interference-aware receiver structure for multi-user mimo and lte. *EURASIP Journal on Wireless Communications and Networking*, 2011.
- [78] Domenico Giustiniano and Stefan Mangold. CAESAR: Carrier Sense-based Ranging in Off-the-shelf 802.11 Wireless LAN. *CoNEXT*, 2011.

- [79] Jon Gjengset, Jie Xiong, Graeme McPhillips, and Kyle Jamieson. Phaser: Enabling Phased Array Signal Processing on Commodity WiFi Access Points. *ACM MobiCom*, 2014.
- [80] H. Charles J. Godfray, John R. Beddington, Ian R. Crute, Lawrence Haddad, David Lawrence, James F. Muir, Jules Pretty, Sherman Robinson, Sandy M. Thomas, and Camilla Toulmin. Food Security: The Challenge of Feeding 9 Billion People. *Science*, 2010.
- [81] S. Gollakota, S.D. Perli, and D. Katabi. Interference alignment and cancellation. *ACM SIGCOMM*, 2009.
- [82] H. C. Gomes and N. B. Carvalho. The use of intermodulation distortion for the design of passive rfid. In *2007 European Radar Conference*, 2007.
- [83] Hugo Gomes and Nuno B. Carvalho. Rfid for location proposes based on the intermodulation distortion. *Sensors & Transducers*, 2009.
- [84] Leslie Greengard and June-yub Lee. Accelerating the Nonuniform Fast Fourier Transform. *SIAM REVIEW*, 2004.
- [85] M. Guillaud, D.T.M. Slock, and R. Knopp. A Practical Method for Wireless Channel Reciprocity Exploitation through Relative Calibration. 2005.
- [86] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. Predictable 802.11 packet delivery from wireless channel measurements. *ACM SIGCOMM Computer Communication Review*, 2011.
- [87] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. Tool Release: Gathering 802.11n Traces with Channel State Information. *ACM SIGCOMM CCR*, 2011.
- [88] Yantao Han, Jiqing Ni, and GaoKe Du. The potential approaches to achieve channel reciprocity in fdd system with frequency correction algorithms. In *International Conference on Communications and Networking in China (CHINACOM)*, 2010.
- [89] B. R. Hanson and S. Orloff. Monitoring soil moisture for irrigation water management. Technical report, UC Davis, 2007.
- [90] J. Heiskala and J. Terry. *OFDM Wireless LANs: A Theoretical and Practical Guide*. Sams Publishing, 2001.
- [91] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and James Collins. *Global Positioning System: Theory and Practice*. Springer Science & Business Media, 2013.
- [92] J. Hou, Y. Zhu, L. Zhang, Y. Fu, F. Zhao, L. Yang, and G. Rong. Design and implementation of a high resolution localization system for in-vivo capsule endoscopy. In *2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing*, 2009.
- [93] Ke Hou, Zirui Zhou, Anthony Man-Cho So, and Zhi-Quan Luo. On the Linear Convergence of the Proximal Gradient Method for Trace Norm Regularization. *NIPS*, 2013.



- [94] Jason Hsu, Sadaf Zahedi, Aman Kansal, Mani Srivastava, and Vijay Raghunathan. Adaptive Duty Cycling for Energy Harvesting Systems. In *International Symposium on Low Power Electronics and Design*, 2006.
- [95] Chao Hu, Max Qinghu Meng, and M. Mandal. Efficient magnetic localization and orientation technique for capsule endoscopy. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005.
- [96] Pan Hu, Pengyu Zhang, Mohammad Rostami, and Deepak Ganesan. Braidio: An Integrated Active-Passive Radio for Mobile Devices with Asymmetric Energy Budgets. *ACM SIGCOMM*, 2016.
- [97] Te-Yuan Huang, Ramesh Johari, and Nick McKeown. Downton Abbey Without the Hiccups: Buffer-based Rate Adaptation for HTTP Video Streaming. *FhMN*, 2013.
- [98] Te-Yuan Huang, Ramesh Johari, Nick McKeown, Matthew Trunnell, and Mark Watson. A Buffer-based Approach to Rate Adaptation: Evidence from a Large Video Streaming Service. *ACM SIGCOMM*, 2014.
- [99] Klaus Hugel, Kimmo Kalliola, and Juha Laurila. Spatial reciprocity of uplink and downlink radio channels in fdd systems. 2002.
- [100] Henkjan J. Huisman, Jurgen J. Fütterer, Emile N. J. T. van Lin, Arjan Welmers, Tom W. J. Scheenen, Jorn A. van Dalen, Andries G. Visser, J. A. Witjes, and Jelle O. Barentsz. Prostate cancer: Precision of integrating functional mr imaging with radiation therapy treatment by using fiducial gold markers. *Radiology*, 2005.
- [101] S. Imtiaz, G. S. Dahman, F. Rusek, and F. Tufvesson. On the directional reciprocity of uplink and downlink channels in frequency division duplex systems. In *IEEE Symposium on Personal Indoor and Mobile Radio Communications*, 2014.
- [102] Telesystem Innovations. Lte in a nutshell. *White paper*, 2010. <https://home.zhaw.ch/kunr/NTM1/literatur/LTE%20in%20a%20Nutshell%20-%20Physical%20Layer.pdf>.
- [103] Institute of Applied Physics. Dielectric Properties of Body Tissues. <http://niremf.ifac.cnr.it/tissprop/htmlclie/htmlclie.php>.
- [104] Texas Instruments. ISM-Band and Short Range Device Regulatory Compliance Overview, 2005. <http://www.ti.com/lit/an/swra048/swra048.pdf>.
- [105] International Data Corporation. Steady commercial and consumer adoption will drive worldwide spending on the internet of things to usd1.1 trillion in 2023, according to a new idc spending guide. <https://www.idc.com/getdoc.jsp?containerId=prUS45197719>.
- [106] Ralf Irmer, Heinz Droste, Patrick Marsch, Michael Grieger, Gerhard Fettweis, Stefan Brueck, Hans-Peter Mayer, Lars Thiele, and Volker Jungnickel. Coordinated multipoint: Concepts, performance, and field trial results. *IEEE Communications Magazine*, 2011.

- [107] A.Z.M. Touhidul Islam and Indraneel Misra. Performance of Wireless OFDM System with LS-Interpolation-Based Channel Estimation in Multi-path Fading Channel. IJCSA, 2012.
- [108] Koichi Ito, Katsumi Furuya, Yoshinobu Okano, and Lira Hamada. Development and characteristics of a biological tissue-equivalent phantom for microwaves. *Electronics and Communications in Japan (Part I: Communications)*, 2001.
- [109] Aditya Jain, Zerina Kapetanovic, Akshit Kumar, Vasuki Narasimha Swamy, Rohit Patil, Deepak Vasisht, Rahul Sharma, Manohar Swaminathan, Ranveer Chandra, Anirudh Badam, Gireeja Ranade, Sudipta Sinha, and Akshay Uttama Nambi S N. Low-cost aerial imaging for small holder farmers. ACM COMPASS, 2019.
- [110] J. L. Jerez, G. A. Constantinides, and E. C. Kerrigan. Fpga implementation of an interior point solver for linear model predictive control. In *International Conference on Field-Programmable Technology (FPT)*, 2010.
- [111] Hyoungju Ji, Younsun Kim, Juho Lee, Eko Onggosanusi, Younghan Nam, Jianzhong Zhang, Byungju Lee, and Byonghyo Shim. Overview of full-dimension mimo in lte-advanced pro. *arXiv preprint arXiv:1601.00019*, 2015.
- [112] Kiran Joshi, Steven Hong, and Sachin Katti. PinPoint: Localizing Interfering Radios. USENIX NSDI, 2013.
- [113] O. Visser K. Langendoen, A. Baggio. Murphy loves Potatoes: Experience from a Pilot Sensor Network Deployment in Precision Agriculture. In *International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, 2006.
- [114] Florian Kaltenberger, David Gesbert, Raymond Knopp, and Marios Kountouris. Performance of multi-user mimo precoding with limited feedback over measured channels. In *IEEE Global Telecommunications Conference*, 2008.
- [115] Florian Kaltenberger, Haiyong Jiang, Maxime Guillaud, and Raymond Knopp. Relative channel reciprocity calibration in mimo/tdd systems. In *IEEE Future Network and Mobile Summit*, 2010.
- [116] Avinash Kalyanaraman, Dezhi Hong, Elahe Soltanaghahi, and Kamin Whitehouse. Forma track: Tracking people based on body shape. *ACM IMWUT*, 2017.
- [117] Emanuel Kanal, A. James Barkovich, Charlotte Bell, James P. Borgstede, William G. Bradley Jr, Jerry W. Froelich, J. Rod Gimbel, John W. Gosbee, Ellisa Kuhni-Kaminski, Paul A. Larson, James W. Lester Jr, John Nyenhuis, Daniel Joe Schaefer, Elizabeth A. Sebek, Jeffrey Weinreb, Bruce L. Wilkoff, Terry O. Woods, Leonard Lucey, and Dina Hernandez. Acr guidance document on mr safe practices: 2013. *Journal Of Magnetic Resonance Imaging*, 2013.
- [118] Srikanth Kandula, Kate Ching-Ju Lin, Tural Badirkhanli, and Dina Katabi. FatVAP: Aggregating AP Backhaul Capacity to Maximize Throughput. USENIX NSDI, 2008.
- [119] Kathleen McLaughlin . Gaps in 4G Network Hinder High-tech Agriculture: FCC Prepares to Release 500 Million to Improve Coverage. <http://www.bendbulletin.com/newsroomstafflist/4535283-151/gaps-in-4g-network-hinder-high-tech-agriculture>.

- [120] Bryce Kellogg, Vamsi Talla, Shyamnath Gollakota, and Joshua R. Smith. Passive wi-fi: Bringing low power to wi-fi transmissions. *USENIX NSDI*, 2016.
- [121] Hak-Jin Kim, Kenneth A. Sudduth, and John W. Hummel. Soil Micronutrient Sensing for Precision Agriculture. *Journal of Environmental Monitoring, The Royal Society of Chemistry*, 2009.
- [122] Jaehoon Kim and Y. Rahmat-Samii. Implanted antennas inside a human body: simulations, designs, and characterizations. *IEEE Transactions on Microwave Theory and Techniques*, 2004.
- [123] Ronold Wyeth Percival King, Glenn S Smith, Margaret Owens, and Tai Tsun Wu. Antennas in matter: Fundamentals, theory, and applications. *NASA STI/Recon Technical Report A*, 81, 1981.
- [124] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. SpotFi: Decimeter Level Localization Using WiFi. *ACM SIGCOMM*, 2015.
- [125] Hideo D Kubo and Bruce C Hill. Respiration gated radiotherapy treatment: a technical study. *Physics in Medicine and Biology*, 1996.
- [126] C.P. Kumar, R. Poovaiah, A. Sen, and P. Ganadas. Single Access Point-based Indoor Localization Technique for Augmented Reality Gaming for Children. *Students' Technology Symposium (TechSym)*, 2014 IEEE, 2014.
- [127] Swarun Kumar, Stephanie Gil, Dina Katabi, and Daniela Rus. Accurate Indoor Localization with Zero Start-up Cost. *ACM MobiCom*, 2014.
- [128] Swarun Kumar, Ezzeldin Hamed, Dina Katabi, and Li Erran Li. LTE Radio Analytics Made Easy and Accessible. *ACM SIGCOMM*, 2014.
- [129] Ye-Sheng Kuo, Pat Pannuto, Ko-Jen Hsiao, and Prabal Dutta. Luxapose: Indoor positioning with mobile phones and visible light. *ACM MobiCom*, 2014.
- [130] Divya Kurup, Gunter Vermeeren, Emmeric Tanghe, Wout Joseph, and Luc Martens. In-to-out body antenna-independent path loss model for multilayered tissues and heterogeneous medium. *IEEE Sensors*, 2014.
- [131] Taehoon Kwon, Yeon-Geun Lim, and Chan-Byoung Chae. Limited channel feedback for rf lens antenna based massive mimo systems. In *IEEE International Conference on Computing, Networking and Communications (ICNC)*, 2015.
- [132] S. Lanzisera, D. Zats, and K.S.J. Pister. Radio Frequency Time-of-Flight Distance Measurement for Low-Cost Wireless Sensor Localization. *Sensors Journal, IEEE*, 2011.
- [133] Erik Larsson, Ove Edfors, Fredrik Tufvesson, and Thomas Marzetta. Massive mimo for next generation wireless systems. *IEEE Communications Magazine*, 2014.
- [134] Mariya Lazebnik, Ernest L Madsen, Gary R Frank, and Susan C Hagness. Tissue-mimicking phantom materials for narrowband and ultrawideband microwave applications. *Physics in Medicine and Biology*, 2005.

- [135] Daewon Lee, Hanbyul Seo, Bruno Clerckx, Eric Hardouin, David Mazzaresse, Satoshi Nagata, and Krishna Sayana. Coordinated multipoint transmission and reception in LTE-advanced: deployment scenarios and operational challenges. *IEEE Communications Magazine*, 2012.
- [136] "W.S. Lee, V. Alchanatis, C. Yang, M. Hirafuji, D. Moshou, and C. Li. Sensing Technologies for Precision Specialty Crop Production. *Computers and Electronics in Agriculture*, 2010.
- [137] Huaxin Li, Zheyu Xu, Haojin Zhu, Di Ma, Shuai Li, and Kai Xing. Demographics inference through Wi-Fi network traffic analysis. INFOCOM, 2016.
- [138] Tianxing Li, Chuankai An, Zhao Tian, Andrew T. Campbell, and Xia Zhou. Human sensing using visible light communication. ACM MobiCom, 2015.
- [139] Tianxing Li, Qiang Liu, and Xia Zhou. Practical human sensing in the light. ACM MobiSys, 2016.
- [140] Z. Li and V. Isler. Large Scale Image Mosaic Construction for Agricultural Applications. *IEEE Robotics and Automation Letters*, 2016.
- [141] Kate Ching-Ju Lin, Shyamnath Gollakota, and Dina Katabi. Random access heterogeneous mimo networks. *ACM SIGCOMM*, 2011.
- [142] Hongbo Liu, Yu Gan, Jie Yang, Simon Sidhom, Yan Wang, Yingying Chen, and Fan Ye. Push the limit of wifi based localization for smartphones. ACM Mobicom, 2012.
- [143] Vincent Liu, Aaron Parks, Vamsi Talla, Shyamnath Gollakota, David Wetherall, and Joshua R. Smith. Ambient Backscatter: Wireless Communication out of Thin Air. ACM SIGCOMM, 2013.
- [144] R. Lodato, V. Lopresto, R. Pinto, and G. Marrocco. Numerical and experimental characterization of through-the-body uhf-rfid links for passive tags implanted into human limbs. *IEEE Transactions on Antennas and Propagation*, 2014.
- [145] Digital Logger. PoE Switch. <http://www.digital-loggers.com/poe48.html>.
- [146] LoRa Technology. <https://www.lora-alliance.org/what-is-lora/technology>.
- [147] Jess Lowenberg-DeBoer. The Precision Agriculture Revolution: Making the Modern Farmer. <https://www.foreignaffairs.com/articles/united-states/2015-04-20/precision-agriculture-revolution>.
- [148] A. Ma and A. S. Y. Poon. Midfield wireless power transfer for bioelectronics. *IEEE Circuits and Systems Magazine*, 2015.
- [149] D. Manteuffel and M. Grimm. Localization of a functional capsule for wireless neuro-endoscopy. In *2012 IEEE Topical Conference on Biomedical Wireless Technologies, Networks, and Sensing Systems (BioWireless)*, 2012.
- [150] Andreas Marcaletti, Maurizio Rea, Domenico Giustiniano, Vincent Lenders, and Aymen Fakhredine. Filtering Noisy 802.11 Time-of-Flight Ranging Measurements. CoNEXT, 2014.
- [151] Alex T. Mariakakis, Souvik Sen, Jeongkeun Lee, and Kyu-Han Kim. SAIL: Single Access Point-based Indoor Localization. MobiSys, 2014.

- [152] Amelia Masters and Katina Michael. Lend me your arms: The use and implications of humancentric rfid. *Electronic Commerce Research and Applications*, 2007.
- [153] A. McBratney and MJ Pringle. Estimating average and proportional variograms of soil properties and their potential use in precision agriculture. *Precision Agriculture*, 1999.
- [154] Harald J. Meyer, Nantarika Chansue, and Fabio Monticelli. Implantation of radio frequency identification device (rfid) microchip in disaster victim identification (dvi). *Forensic Science International*, 2006.
- [155] K. Michael. Rfid/nfc implants for bitcoin transactions. *IEEE Consumer Electronics Magazine*, 2016.
- [156] Microseven. Ip cameras. <http://www.microseven.com/product/IP-Cameras.html>.
- [157] Microsoft. Azure IoT Hub. <https://azure.microsoft.com/en-us/services/iot-hub>.
- [158] Microsoft. Azure IoT Suite. <https://www.microsoft.com/en-us/cloud-platform/internet-of-things-azure-iot-suite>.
- [159] Microsoft Research. Image Composite Editor. <http://research.microsoft.com/en-us/um/redmond/projects/ice>.
- [160] B. J. Mohammed, A. M. Abbosh, S. Mustafa, and D. Ireland. Microwave system for head imaging. *IEEE Transactions on Instrumentation and Measurement*, 2014.
- [161] Nathaniel D. Mueller, James S. Gerber, Matt Johnston, Deepak K. Ray, and Jonathan A. Foley Navin Ramankutty. Closing Yield Gaps through Nutrient and Water management. *Nature*, 2012.
- [162] A. Nag and S.C. Mukhopadhyay. Occupancy Detection at Smart Home Using Real-Time Dynamic Thresholding of Flexiforce Sensor. *Sensors Journal, IEEE*, 2015.
- [163] L. Nagl, R. Schmitz, S. Warren, T. S. Hildreth, H. Erickson, and D. Andresen. Wearable Sensor System for Wireless State-of-health Determination in Cattle. In *Engineering in Medicine and Biology Society*, 2003.
- [164] Young-Han Nam, Md Saifur Rahman, Yang Li, Gang Xu, Eko Onggosanusi, Jianzhong Zhang, and Ji-Yun Seol. Full dimension mimo for lte-advanced and 5g. *IEEE Communications Magazine*, 2013.
- [165] New York Times. Homes Try to Reach Smart Switch, 2015. <http://www.nytimes.com/2015/04/23/business/energy-environment/homes-try-to-reach-smart-switch.html>.
- [166] Shahriar Nirjon, Jie Liu, Gerald DeJean, Bodhi Priyantha, Yuzhe Jin, and Ted Hart. Coin-gps: Indoor localization from direct gps receiving. *ACM MobiSys*, 2014.
- [167] C. Oancea, K. Shipulin, G. Mytsin, A. Molokanov, D. Niculae, I. Ambrožová, and M. Davídková. Effect of titanium dental implants on proton therapy delivered for head tumors: experimental validation using an anthropomorphic head phantom. *Journal of Instrumentation*, 2017.

- [168] J. Oh and J. Um. Acoustic signal-based indoor global coordinates system for smartphones. *IEEE Sensors Journal*, 2018.
- [169] Tamoghna Ojha, Sudip Misra, and Narendra Singh Raghuwanshi. Wireless sensor networks for agriculture. *Computers and Electronics in Agriculture*, 2015.
- [170] Teruo Onishi and Shinji Uebayashi. Biological Tissue-equivalent Phantoms Usable in Broadband Frequency Range. *NTT DoCoMo Technical Journal*, 2006.
- [171] OpenWeatherMap, Inc. OpenWeather API. <http://openweathermap.org>.
- [172] Sophocles J Orfanidis. *Electromagnetic waves and antennas*. Rutgers University New Brunswick, NJ, 2002.
- [173] George Ou, Neal Shahidi, Cherry Galorport, Oliver Takach, Terry Lee, and Robert Enns. Effect of longer battery life on small bowel capsule endoscopy. *World Journal of Gastroenterology*, 2015.
- [174] Nico Palleit and Tobias Weber. Obtaining transmitter side channel state information in mimo fdd systems. In *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2009.
- [175] Nico Palleit and Tobias Weber. Channel prediction in point-to-point mimo-systems. In *IEEE International Symposium on Wireless Communication Systems (ISWCS)*, 2010.
- [176] Shwetak N. Patel, Julie A. Kientz, Gillian R. Hayes, Sooraj Bhat, and Gregory D. Abowd. Farther Than You May Think: An Empirical Investigation of the Proximity of Users to Their Mobile Phones. *UbiComp*, 2006.
- [177] D. J. Patterson, D. Fox, H. Kautz, and M. Philipose. Fine-grained Activity Recognition by Aggregating Abstract Object Usage. *ISWC*, 2005.
- [178] Duc Minh Pham and Syed Mahfuzul Aziz. A real-time localization system for an endoscopic capsule using magnetic sensors. *IEEE Sensors*, 2014.
- [179] PIX4D. Pix4d. <https://pix4d.com>.
- [180] Bozidar Radunovic, Alexandre Proutiere, Dinan Gunawardena, and Peter Key. Dynamic channel, rate selection and scheduling for white spaces. *ACM CoNEXT*, 2011.
- [181] Hariharan Rahul, Haitham Hassanieh, and Dina Katabi. SourceSync: A Distributed Wireless Architecture for Exploiting Sender Diversity. *ACM SIGCOMM*, 2010.
- [182] Hariharan Rahul, Swarun Kumar, and Dina Katabi. MegaMIMO: Scaling Wireless Capacity with User Demands. *ACM SIGCOMM*, 2012.
- [183] Anshul Rai, Krishna Kant Chintalapudi, Venkata N. Padmanabhan, and Rijurekha Sen. Zee: Zero-effort Crowdsourcing for Indoor Localization. *ACM MobiCom*, 2012.
- [184] Xiongbin Rao and Vincent K. N. Lau. Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems. *CoRR*, abs/1405.2786, 2014.

- [185] K. Rasilainen, J. Ilvonen, A. Lehtovuori, J. M. Hannula, and V. Viikari. On design and evaluation of harmonic transponders. *IEEE Transactions on Antennas and Propagation*, 2015.
- [186] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.
- [187] S. Roberts, P. Garnett, and R. Chandra. Connecting Africa Using the TV White Spaces: From Research to Real World Deployments. In *IEEE LANMAN*, 2015.
- [188] Martyn Roetter. Spectrum for mobile broadband in the americas: Policy issues for growth and competition. 2011.
- [189] Montserrat Ros, Joshua Boom, Gavin de Hosson, and Matthew D’Souza. Indoor Localisation Using a Context-Aware Dynamic Position Tracking Model. *International Journal of Navigation and Observation*, 2012.
- [190] L. Ruiz-Garcia, L. Lunadei, P. Barreiro, and J. Ignacio Robla. A Review of Wireless Sensor Technologies and Applications in Agriculture and Food Industry: State of the Art and Current Trends. *Sensors*, 2009.
- [191] Christoph Scholz, Martin Atzmueller, and Gerd Stumme. *Unsupervised and Hybrid Approaches for On-line RFID Localization with Mixed Context Knowledge*. International Symposium on Foundations of Intelligent Systems. 2014.
- [192] S. Y. Semenov, A. E. Bulyshev, A. Abubakar, V. G. Posukh, Y. E. Sizov, A. E. Souvorov, P. M. van den Berg, and T. C. Williams. Microwave-tomographic imaging of the high dielectric-contrast objects using different image-reconstruction approaches. *IEEE Transactions on Microwave Theory and Techniques*, 2005.
- [193] Souvik Sen, Jeongkeun Lee, Kyu-Han Kim, and Paul Congdon. Avoiding Multipath to Revive Inbuilding Wi-Fi Localization. *ACM MobiSys*, 2013.
- [194] Souvik Sen, Bozidar Radunovic, Jeongkeun Lee, and Kyu-Han Kim. Cspy: Finding the best quality channel without probing. *ACM MobiCom*. ACM, 2013.
- [195] Suranga Seneviratne, Aruna Seneviratne, Prasant Mohapatra, and Anirban Mahanti. Your Installed Apps Reveal Your Gender and More! *SIGMOBILE Mob. CCR*, July 2014.
- [196] SenseFly. Sensefly. [sensefly.com](http://sensefly.com).
- [197] Sentera. <https://sentera.com>.
- [198] P. Setlur, G. Alli, and L. Nuzzo. Multipath Exploitation in Through-Wall Radar Imaging Via Point Spread Functions. *IEEE Transactions on Image Processing*, 2013.
- [199] Michael Shamos. *Computational Geometry*. Yale University, 1978.
- [200] Navin Sharma, Jeremy Gummeson, David E. Irwin, and Prashant J. Shenoy. Cloudy Computing: Leveraging Weather Forecasts in Energy Harvesting Sensor Systems. In *IEEE SECON*, 2010.

- [201] Lixin Shi, Paramvir Bahl, and Dina Katabi. Beyond sensing: Multi-ghz realtime spectrum analytics. In *USENIX NSDI*, 2015.
- [202] Tan Shuang, T. Koivisto, H.-L. Maattanen, K. Pietikainen, T. Roman, and M. Enescu. Design and evaluation of lte-advanced double codebook. In *IEEE Vehicular Technology Conference (VTC Spring)*, 2011.
- [203] SIGFOX. <http://www.sigfox.com>.
- [204] Min Soo Sim and Chan-Byoung Chae. Compressed channel feedback for correlated massive mimo systems. In *IEEE Globecom Workshops (GC Wkshps)*, 2014.
- [205] Sudipta N Sinha and Marc Pollefeys. Pan-tilt-zoom Camera Calibration and High-resolution Mosaic Generation. *Computer Vision and Image Understanding*, 2006.
- [206] Skyworks. SMS7630 Series. [http://www.skyworksinc.com/Product/511/SMS7630\\_Series?IsProduct=true](http://www.skyworksinc.com/Product/511/SMS7630_Series?IsProduct=true).
- [207] M. Soderstrom, Gustav Sohlenius, Lars Rodhe, and Kristin Piikki. Adaptation of Regional digital soil mapping for precision agriculture. *Precision Agriculture*, 2016.
- [208] John V. Stafford. *Precision Agriculture Book*. 2005.
- [209] P. R. Stauffer, F. Rossetto, M. Prakash, D. G. Neuman, and T. Lee. Phantom and animal tissues for modelling the electrical properties of human liver. *International Journal of Hyperthermia*, 2003.
- [210] Shaohui Sun, Qiubin Gao, Ying Peng, Yingmin Wang, and Lingyang Song. Interference management through comp in 3gpp lte-advanced networks. *IEEE Wireless Communications*, 2013.
- [211] Zheng Sun, Rick Farley, Telis Kaleas, Judy Ellis, and Kiran Chikkappa. Cortina: Collaborative Context-aware Indoor Positioning Employing RSS and RToF Techniques. PERCOM, 2011.
- [212] A Surowiec, S S Stuchly, L Eidus, and A Swarup. In vitro dielectric properties of human tissues at radiofrequencies. *Physics in Medicine and Biology*, 1987.
- [213] Richard Szeliski. Image Alignment and Stitching: A Tutorial. Technical report, Microsoft Research, 2004.
- [214] Qinghui Tang, S. K. S. Gupta, and L. Schwiebert. BER performance analysis of an on-off keying based minimum energy coding for energy constrained wireless sensor applications. In *IEEE International Conference on Communications*, 2005.
- [215] Taoglas. PC 30 Antenna. <http://www.taoglas.com/product/pc30-2g3g-cellular-fr4-pcb-antenna-mmcmra-2>.
- [216] D. Tse and P. Vishwanath. *Fundamentals of Wireless Communications*. Cambridge University Press, 2005.
- [217] I. Umay, B. Fidan, and M. R. Yüce. Endoscopic capsule localization with unknown signal propagation coefficients. In *2015 International Conference on Advanced Robotics (ICAR)*, 2015.



- [218] Ilknur Umay, Barış Fidan, and Billur Barshan. Localization and tracking of implantable biomedical sensors. *IEEE Sensors*, 2017.
- [219] United Nations General Assembly. Food Production Must Double by 2050 to Meet Demand from World’s Growing Population, Innovative Strategies Needed to Combat Hunger, Experts Tell Second Committee, 2009. <http://www.un.org/press/en/2009/gaef3242.doc.htm>.
- [220] Deepak Vasisht, Anubhav Jain, Chen-Yu Hsu, Zachary Kabelac, and Dina Katabi. Duet: Estimating user position and identity in smart homes using intermittent and incomplete rf-data. *ACM IMWUT/UbiComp*, 2018.
- [221] Deepak Vasisht, Zerina Kapetanovic, Jongho Won, Xinxin Jin, Ranveer Chandra, Sudipta Sinha, Ashish Kapoor, Madhusudhan Sudarshan, and Sean Stratman. Farmbeats: An iot platform for data-driven agriculture. *USENIX NSDI*, 2017.
- [222] Deepak Vasisht, Swarun Kumar, and Dina Katabi. Decimeter-Level Localization with a Single WiFi Access Point. *USENIX NSDI*, 2016.
- [223] Deepak Vasisht, Swarun Kumar, Hariharan Rahul, and Dina Katabi. Eliminating channel feedback in next-generation cellular networks. *ACM SIGCOMM*, 2016.
- [224] Deepak Vasisht, Guo Zhang, Omid Abari, Hsiao-Ming Lu, Jacob Flanz, and Dina Katabi. In-body backscatter communication and localization. *ACM SIGCOMM*, 2018.
- [225] H. Victor. Which 4G LTE bands do ATT, Verizon, T-Mobile and Sprint use in the USA? [http://www.phonearena.com/news/Cheat-sheet-which-4G-LTE-bands-do-AT-T-Verizon-T-Mobile-and-Sprint-use-in-the-USA\\_id77933](http://www.phonearena.com/news/Cheat-sheet-which-4G-LTE-bands-do-AT-T-Verizon-T-Mobile-and-Sprint-use-in-the-USA_id77933).
- [226] Christopher M. Vigorito, Deepak Ganesan, and Andrew G. Barto. Adaptive Control of Duty Cycling in Energy-harvesting Wireless Sensor Networks. In *IEEE SECON*, 2007.
- [227] Chen Wang, Qinye Yin, and Hongyang Chen. Robust Chinese Remainder Theorem Ranging-method based on Dual-Frequency Measurements. *IEEE Transactions on Vehicular Technology*, 2011.
- [228] Chen Wang, Qinye Yin, and Wenjie Wang. An Efficient Ranging Method for Wireless Sensor Networks. *ICASSP*, 2010.
- [229] Ju Wang, Hongbo Jiang, Jie Xiong, Kyle Jamieson, Xiaojiang Chen, Dingyi Fang, and Binbin Xie. LiFS: Low Human-effort, Device-free Localization with Fine-grained Subcarrier Information. *ACM MobiCom*, 2016.
- [230] Jue Wang and Dina Katabi. Dude, Where’s My Card?: RFID Positioning That Works with Multipath and Non-line of Sight. *ACM SIGCOMM*, 2013.
- [231] Jue Wang, Deepak Vasisht, and Dina Katabi. Rf-idraw: Virtual touch screen in the air using rf signals. *ACM SIGCOMM*, 2014.
- [232] Wei Wang, Alex X. Liu, and Muhammad Shahzad. Gait Recognition Using Wi-Fi Signals. *Ubi-Comp*, 2016.

- [233] Y. Wang, R. Fu, Y. Ye, U. Khan, and K. Pahlavan. Performance bounds for rf positioning of endoscopy camera capsules. In *2011 IEEE Topical Conference on Biomedical Wireless Technologies, Networks, and Sensing Systems*, 2011.
- [234] T. Wark, P. Csiro, P. Corke, Sikka, L. Klingbeil, Y. Guo, C. Crossman, P. Valencia, D. Swain, and G. Bishop-Hurley. Transforming Agriculture through Pervasive Wireless Sensor Networks. *IEEE Pervasive Computing*, 2007.
- [235] Eric Weisstein. Chinese Remainder Theorem. <http://mathworld.wolfram.com/ChineseRemainderTheorem.html>.
- [236] F. Wen and C. Liang. Fine-Grained Indoor Localization Using Single Access Point With Multiple Antennas. *IEEE Sensors Journal*, 2015.
- [237] Geoff Werner-Allen, Konrad Lorincz, Jeff Johnson, Jonathan Lees, and Matt Welsh. Fidelity and Yield in a Volcano Monitoring Sensor Network. In *Operating Systems Design and Implementation*, 2006.
- [238] Oliver Werther and Roland Minihold. Lte system specifications and their impact on rf & base band circuits. *Application Note, Rohde and Schwarz*, 2013.
- [239] Sigit Basuki Wibowo, Martin Klepal, and Dirk Pesch. Time of Flight Ranging using Off-the-self IEEE802.11 WiFi Tags. POCA, 2009.
- [240] WiFi Alliance. Make Security a Priority in 2011: Protect Your Personal Data on Wi-Fi Networks, 2011. <http://www.wi-fi.org/news-events/newsroom/make-security-a-priority-in-2011-protect-your-personal-data-on-wi-fi-networks>.
- [241] Ian C Wong and Brian L Eva. Joint channel estimation and prediction for ofdm systems. In *IEEE Global Telecommunications Conference*, 2005.
- [242] Lu Wu, Jinhui Chen, Hongwei Yang, and Di Lu. Codebook design for lte-a downlink system. In *IEEE Vehicular Technology Conference (VTC Fall)*, 2011.
- [243] Tong Xin, Bin Guo, Zhu Wang, Mingyang Li, and Zhiwen Yu. FreeSense: Indoor Human Identification with Wi-Fi Signals. *CoRR Arxiv*, 2016.
- [244] Jie Xiong and Kyle Jamieson. ArrayTrack: A Fine-Grained Indoor Location System. USENIX NSDI, 2013.
- [245] Jie Xiong, Kyle Jamieson, and Karthikeyan Sundaresan. Synchronicity: Pushing the Envelope of Fine-grained Localization with Distributed MIMO. HotWireless, 2014.
- [246] Jie Xiong, Karthikeyan Sundaresan, and Kyle Jamieson. ToneTrack: Leveraging Frequency-Agile Radios for Time-Based Indoor Wireless Localization. ACM MobiCom, 2015.
- [247] Chenren Xu, Bernhard Firner, Yanyong Zhang, Richard Howard, Jun Li, and Xiaodong Lin. Improving RF-based Device-free Passive Localization in Cluttered Indoor Environments Through Probabilistic Classification Methods. IPSN, 2012.

- [248] Yi Xu, Guosen Yue, and Shiwen Mao. User grouping for massive mimo in fdd systems: New design methods and analysis. *Access, IEEE*, 2014.
- [249] Zhice Yang, Zeyu Wang, Jiansong Zhang, Chenyu Huang, and Qian Zhang. Wearables can afford: Light-weight indoor positioning with visible light. *ACM MobiSys*, 2015.
- [250] Yunxing Ye and Kaveh Pahlavan. Accuracy bounds for and rss and toa based rf localization in capsule endoscopy. 2011.
- [251] Moustafa Youssef and Ashok Agrawala. The Horus WLAN Location Determination System. *MobiSys*, 2005.
- [252] Moustafa Youssef, Matthew Mah, and Ashok Agrawala. Challenges: Device-free Passive Localization for Wireless Environments. *MobiCom*, 2007.
- [253] M. R. Yuce and T. Dissanayake. Easy-to-swallow wireless telemetry. *IEEE Microwave Magazine*, 2012.
- [254] Gergely V Zàruba, Manfred Huber, FA Kamangar, and Imrich Chlamtac. Indoor Location Tracking using RSSI Readings From a Single Wi-Fi Access Point. *Wireless networks*, 2007.
- [255] Chi Zhang and Xinyu Zhang. Litell: Robust indoor localization using unmodified light fixtures. In *Proceedings of the 22Nd Annual International Conference on Mobile Computing and Networking*, ACM MobiCom, 2016.
- [256] L. Zhang, Y. Zhu, T. Mo, J. Hou, and H. Hu. Design of 3d positioning algorithm based on rfid receiver array for in vivo micro-robot. In *IEEE International Conference on Dependable, Autonomic and Secure Computing*, 2009.
- [257] Le Zhang, Yongxin Zhu, Tingting Mo, Jinlong Hou, and Guoguang Rong. Design and implementation of 3d positioning algorithms based on rf signal radiation patterns for in vivo micro-robot. *International Conference on Body Sensor Networks*, 2010.
- [258] Pengyu Zhang, Dinesh Bharadia, Kiran Joshi, and Sachin Katti. HitchHike: Practical Backscatter Using Commodity WiFi. *ACM SenSys*, 2016.
- [259] Tan Zhang, Ning Leng, and Suman Banerjee. A vehicle-based measurement framework for enhancing whitespace spectrum databases. *ACM MobiCom*, 2014.
- [260] Xiuyuan Zheng, Chen Wang, Yingying Chen, and Jie Yang. Accurate Rogue Access Point Localization Leveraging Fine-grained Channel Information. *IEEE Conference on Communications and Network Security (CNS)*, 2014.
- [261] Shilin Zhu and Xinyu Zhang. Enabling High-Precision Visible Light Localization in Today’s Buildings. *ACM MobiSys*, 2017.