GRADUATE THESIS/DISSERTATION APPROVAL FORM AND SIGNATURE PAGE

Instructions: This form must be completed by all master's and doctoral students with a thesis or dissertation requirement. Please type or print clearly as this form MUST be included as page 1 of your thesis or dissertation via electronic submission to ProQuest. All theses and dissertations must be formatted according to the University and department/program requirements. **Reminder:** It is the responsibility of the student to submit any/all edits requested by the Examining Committee to the Faculty Mentor or Supervising Professor for final approval and signature via the Graduate Program Completion Form.

Type: ☐ Master's Thesis ☑ PhD/Doctoral Thesis or Dissertation

Thesis or Dissertation Title: Predictive Analytics on Real-Time Biofeedback for Actionable Classification of Activity State

Author's Name: William M. Mongan

Month and Year: August, 2018

The signatures below certify that this thesis / dissertation (circle one) is complete and approved by the Examining Committee.

Committee Chairperson's Name: Kapil R. Dandekar

Title: Professor

Department: Electrical and Computer Engineering

Institution (*if other than Drexel University*):

Signature: Kapil R Dandekar

Committee Member's Name: Adam Fontecchio

Title: Professor

Department: Electrical and Computer Engineering

Institution (*if other than Drexel University*):

Signature:

Committee Member's Name: Nagarajan Kandasamy

Title: Professor

Department: Electrical and Computer Engineering

Institution (*if other than Drexel University*):

Signature: K. Nag

Committee Member's Name: Genevieve Dion

Title: Associate Professor

Department: ~~Fashion~~ Design

Institution (*if other than Drexel University*):

Signature:

Committee Member's Name: Timothy Kurzweg

Title: Professor

Department: Electrical and Computer Engineering

Institution (*if other than Drexel University*): Pennsylvania State University

Signature: Tim Kurzweg

Committee Member's Name:

Title:

Department:

Institution (*if other than Drexel University*):

Signature:

Predictive Analytics on Real-Time Biofeedback for Actionable Classification of Activity State

A Thesis

Submitted to the Faculty

of

Drexel University

by

William M. Mongan

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy

August 2018

Dedications

To Christina, my wife, for her years of sacrifice to enable me to pursue my goals.

Acknowledgments

This work is made possible by many people. I am grateful to my advisor, Dr. Adam Fontecchio, whose years of guidance in educational and scientific pursuits have motivated me to an academic career. My multidisciplinary committee, chaired by Dr. Kapil Dandekar, along with Genevieve Dion, Naga Kandasamy, and Tim Kurzweg, have been guiding collaborators on this and many other areas of research. Their support has been helpful on this project and also broadly enriching. This project was enabled by medical and human factors guidance from Sociologist Dr. Kelly Joyce and physicians Dr. Endla Anday, Dr. Vineet Bhandari, Dr. Owen Montgomery, and Dr. Michael Weingarten. Through generous support of the National Science Foundation, the National Institutes of Health, and the Commonwealth of Pennsylvania, I have had the opportunity to integrate interesting work with other researchers, including Sayandeep Acharya, Chelsea Knittel, Ariana Levitt, Yuqiao Liu, Ilhaan Rasheed, Khyati Ved, and Shrenik Vora. I am also grateful for the opportunity to extend our research to undergraduate curricula through experiential project work and support from the Drexel University STAR program, the NSF Research Experiences for Undergraduates (REU) program, and a Commonwealth Universal Research Enhancement (CURE) grant with Semanti Basu, Samantha Bewley, Rachel Goeken, Stephen Hansen, Brent Lee, Christopher Lynch, Robert Ross, Patrick O'Neill, and Har Patel, as well as with K-12 teachers Michael Darfler, Kyle Ellis, Barbara Matas, and Anna Schall through an NSF Research Experiences for Teachers (RET). I am deeply thankful for the opportunity to develop my skills in such a collaborative environment.

Our research results are based upon work supported by the National Science Foundation Partnerships for Innovation: Building Innovation Capacity (PFI:BIC) subprogram under Grant No. 1430212. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Research reported in this publication was supported by National Institute of Biomedical Imaging and Bioengineering of the National Institutes of Health under award number U01EB023035. The

content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. This work was supported in part by the Commonwealth of Pennsylvania through the Commonwealth Universal Research Enhancement (CURE) program under award number SAP117558-014.

Table of Contents

List of Tables

List of Figures

Abstract

Predictive Analytics on Real-Time Biofeedback for Actionable Classification of Activity State

William M. Mongan

Adam K. Fontecchio, Ph.D.

Continuous biomedical monitoring has the potential to improve quality-of-care for patients as well as working conditions for medical practitioners over the current state-of-the-art. Currently, Emergency Medical Technicians in the field carry monitoring equipment that can weigh over 50 lbs, and manually communicate information back to hospital physicians. For patients, medical monitoring is carried out using tethered equipment that must remain attached for hours or days, and must be removed when the patient must get up to walk or use the restroom. Data lapses during these disconnected breaks can be misinterpreted by medical staff as a medical event, and true medical events can be missed as a result of non-monitoring. Further, being still for extended periods of time can exacerbate the very risks being treated, due to the increased risk of a blood clot while remaining stationary during monitoring.

Radio Frequency Identification (RFID) technology is traditionally used as a battery-free chip embedded into an item for inventory management. As the chip is placed within the field of an RFID interrogator, one or more interrogation waves are reflected off of the chip and observed at the interrogator site. The reflected signal is encoded by the chip with an identifier which is typically used for inventory purposes. Multiple interrogation signals are typically employed to overcome collisions and to ensure that a viable interrogation takes place while the chip is in range of the interrogator. We take advantage of this property of RFID technology by knitting a metallic antenna around the tag and embedding the system into a wearable garment in an unobtrusive way. We re-purpose the use of RFID by observing small perturbations in the physical properties of the reflected signal for each interrogation of the tag. As the wearer moves about, changes in the knit antenna shape result in changes to the properties of the reflected signal as it is regularly and frequently polled by the interrogator. These physical changes are small and subject to noise interference both from RF,

movements in the environment around the subject, and movements by the subject directly; however, we fuse signal processing and machine learning approaches to estimate biomedical properties of the wearer such as respiratory rate, apnea, uterine contractions, and stationary limbs. As a result, we introduce a wearable technology platform supported by real-time analytical software that enables unobtrusive, continuous, ambulatory monitoring of strain or movement biomedical artifacts.

Part I

Introduction and Background

Chapter 1: Introduction

1.1 Overview and Motivation

Tethered movement-based medical monitoring devices are used in many inpatient and outpatient settings, including Deep Venous Thrombosis (DVT) onset monitoring[1], respiratory monitoring[2], and uterine monitoring during labor and delivery[3]. Continuous respiration monitoring has the potential to predict the onset of acute medical conditions such as apnea[4] and sepsis[5]. Decreased mobility can also lead to acute conditions such as venous thromboembolism (VTE)[6]. Additionally, the use of graduated compression stockings are often used when patients are immobilized, but discomfort during use and prolonged wear may result in reduced compliance[7] that could benefit from an unobtrusive wearable solution.

Tethered monitors require surface area on the body, can be irritating to the body due to the adhesive nature of the sensors, and require that the subject be relatively stationary in the monitoring setting. Requiring stationarity can be counterproductive to the monitoring being sought, and infeasible in circumstances that require several hours of monitoring when other bodily functions arise. Emergency Medical Technicians (EMT) in the field carry monitoring equipment weighing 50-75 *lbs* in addition to potentially moving the patient, and have higher workload interpreting data and communicating information back to hospital and physicians[8]. Further, telemetry of monitoring data may require bulky, possibly tethered equipment, and emergency medical personnel perceptions of ease of use of devices may affect the likelihood of their perceived usefulness and subsequent deployment[9].

Using Radio Frequency Identification (RFID) tags embedded within a knitted antenna and wearable garment "Bellyband" (see Figure 1.3), both strain-gauge movements (such as stretching the garment due to respiratory activity) as well as gross movement (such as walking around a room) can be monitored wirelessly using a passive smart-garment device that requires no tethered power source[10]. A 900 MHz RFID interrogator supplies power wirelessly to the RFID tag via polling

with a wireless signal that is backscattered from the tag and antenna[11]. RFID is traditionally used in inventory management systems, such as that seen in highway toll collection, or retail inventory tracking. As summarized in Figure 1.1, the RFID interrogator polls the field for any available RFID tags, and any tags in the area are excited by the signal.

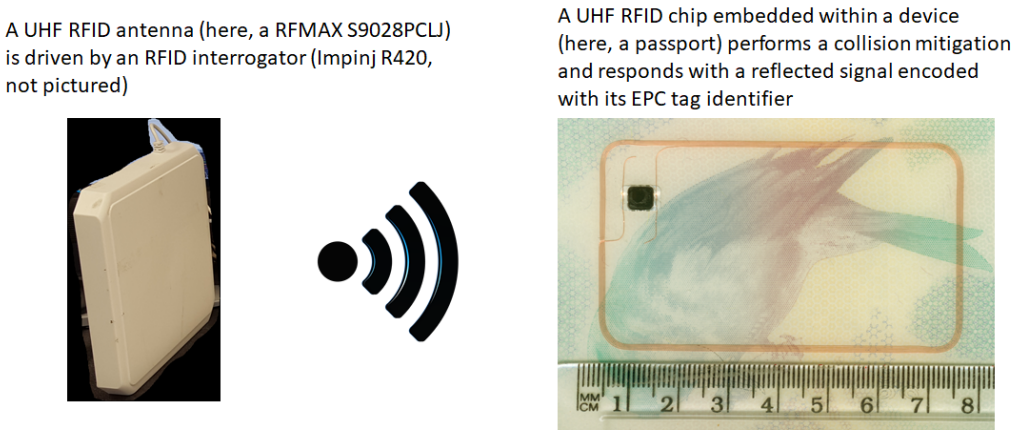

Figure 1.1: A general overview of physical RFID interrogation

As an example, for an infant respiratory monitoring application, the Bellyband is worn about the abdomen or integrated into a onesie or other garment, which stretches and retracts with abdominal respiratory motion. The antenna is regularly polled at a high rate (*i.e.*, 90 Hz), and as the antenna stretches, the strain is observed via changes in the RFID Received Signal Strength Indicator (RSSI) signal power reflected from the antenna, as shown in Figure 1.2. An RFID-based heart rate monitor is enabled via a low-power circuit that disables the RFID interrogation response for a short period of time when an electrocardiogram (ECG) signal is observed from the heart[12]. The circuit and the Bellyband RFID chip are energized using power harvested from the wireless interrogation signal itself, so no battery is required in either wearable device.

To facilitate implementation in the small space of an RFID tag, and with low- to no- power consumption, a non-sophisticated singulation protocol is executed on the RFID tags to reduce the number of collisions in the tag responses. The singulation protocol is based on the slotted-ALOHA anti-collision algorithm in which each tag uses a random-number generator to decide whether to respond in a given time slot. This process is repeated until all tags in the population have responded, at which time the entire process repeats again[13;14]. The RFID backscatter response is encoded

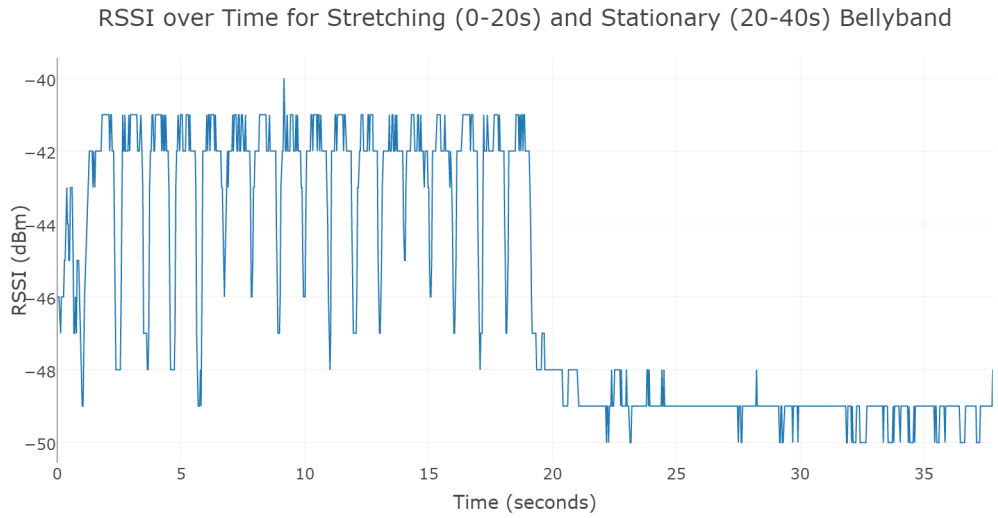

Figure 1.2: RSSI time-series data plotted over time for approximately 40 seconds; although oscillatory motions can be observed every second, smaller perturbations in the data obscure this signal due to the RFID interrogation frequency changing every 200 milliseconds

with a tag identifier such as the 96-bit Electronic Product Code (EPC, *a.k.a.*, EPC96)[15], and the identifier is used as a primary key in a database to obtain the details of the tag being considered (such as the retail product information, vehicle toll billing information, *etc.*). Because collisions are expected when interrogating with RFID, the RFID signal is sent many times, and duplicate backscatter responses are possible. These are resolved at the application layer, for example, by ignoring duplicate interrogation responses within a certain time duration.

It is important to note that RFID was not designed to be used for continuous monitoring applications, especially within the biomedical domain. RFID readers are expensive, have no computation or storage capabilities, and require external devices to utilize networking features. In real-time monitoring applications, RFID technology is re-purposed so that a single tag is repeatedly interrogated, and the duplicate backscatter responses are utilized for their signal properties, such as RSSI or signal phase. Repurposing RFID to track the small changes in signal backscatter properties presents several technical, regulatory, and human challenges[16], which are detailed in Section 2.1. Among these challenges is the need to separate changes in signal properties over time due to strain-gauge motion (*i.e.*, respiratory activity) from those changes due to fading and multipathing from other

objects moving in the field and dynamically blocking or reflecting the signal, and from those changes due to "ambient" motion artifacts from the subject that are unrelated to strain-gauge movement. This is done with no processing on the wearable device, other than that to provide standard RFID interrogation responses as described in this Section; all subject state estimation and processing is performed wirelessly, away from the user, and on the interrogation side.

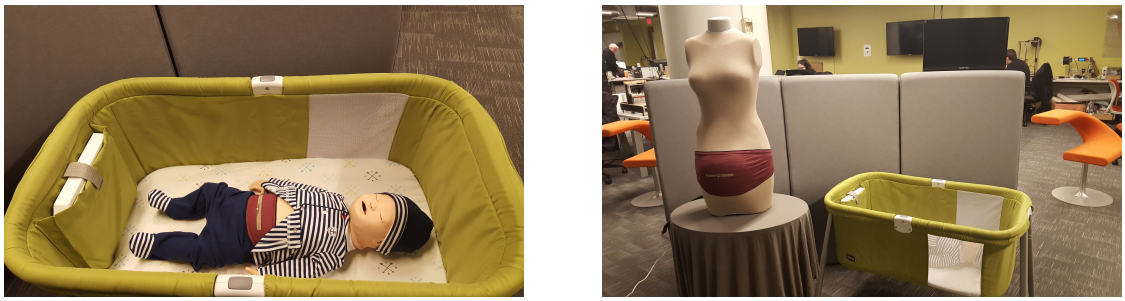

Figure 1.3: The knitted smart-garment device Bellyband uses an embedded RFID tag and knit metallic thread antenna to monitor strain-gauge movements such as respiration (left) or a uterine contraction (right).

Knitted wearable garment devices facilitate ubiquitous monitoring of biometrics: strain-gauge sensing such as respiratory and uterine monitoring[16], non-strain-gauge sensing such as walking or other muscle movements[17], and sub-dermal electrodiagnostics such as heart monitoring[12]. Embedded RFID-based passive devices represent an Internet-of-Things (IoT) network that should be capable of telemetry via the interrogator and among legacy monitoring devices. A software framework is needed to support the physical RFID and IoT infrastructure both efficiently and in compliance with Health Insurance Portability and Accountability Act (HIPAA) regulations[18], and is detailed in Section 4.2.

A wearable passive garment device avoids the need for a tethered monitoring system, so that monitoring can be performed while standing or even ambulatory. We envision ubiquitous, unobtrusive monitoring platform that allows for the collection of individual baseline biometrics while healthy without requiring active device deployment; active sensor deployment may only occur when monitoring values are needed, such as when the patient is experiencing an abnormal state. It is known that human physiological "normal ranges" are subject to individual variation that is not explained by

a chronic condition[19]. For example, individual physiological measures such as respiratory activity, body surface temperature, and pulse are subject to variance due to age and physical activity and can be correlated[20]. Such relationships suggest that wearable monitors can aid in management of chronic disease, such as Type 1 Diabetes, during periods of physical activity[21].

The RFID tag physical backscatter properties perturb as the knitted fabric antenna and smart garment are stretched in space. The antenna is shaped to be resonant at a frequency within the 900 MHz Ultra High Frequency (UHF) RFID frequency band, and the impedance match attained with that shape is degraded as the shape is deformed due to body movements. The signal strength, for example, becomes weaker as the antenna is stretched, and returns to near optimal strength as the antenna is relaxed. However, the tag itself may move closer to the RFID interrogator as the wearer inhales, which strengthens the backscatter signal, working against this designed impedance mismatch. Further, the interrogator frequency changes every 200 milliseconds by federal regulation, without a corresponding change on the RFID tag and knitted antenna; this change in frequency further degrades the impedance match between the antenna and a specific frequency within the band.

Finally, variations in deployment result in variations in physical backscatter properties that render thresholding approaches ineffective. For example, changes in wearer fit creates an initial stretch in the antenna that may extend short of or beyond the shape required for optimal impedance matching with the interrogation signal. Absorption of RF energy by the human body may vary from body to body and result in a degraded or lack of backscatter response to the interrogator. Variations within the physical garment can also result in signal variation, including displacement of metallic threads causing a short of the RF signal with the chip assembly. Changing physical environmental factors can interact dynamically with the signal, such as a person walking through the area, or a conductive object in the field. As a result, the signal patterns can vary from wearer to wearer and band to band, necessitating observation of individual baseline environmental factors. Exacerbating these issues, commercial RFID interrogation equipment may quantize the physical backscatter properties returned to the user for processing, aggregating these small perturbations

together due to rounding.

Despite these challenges, we focus in this effort on wearer state classification, monitoring, and prediction using a single RFID interrogator, a single knitted antenna and RFID tag wearable smart garment device, and a single RFID interrogator antenna. This minimalistic design was chosen to enable a platform framework that could scale to multiple application domains, such as mobile monitoring using an interrogator coupled to a mobile device, using features that are available from commercially available equipment. Single tag classification produces results that can be used as a baseline for multi-sensor fusion research involving wearable reference tags, *i.e.*, those worn on the body but away from abdominal movements during respiratory monitoring.

1.2 Thesis Contributions

The main contributions of this thesis are summarized in Figure 1.4, and include a software framework to securely interface with heterogeneous physical IoT devices (Section 1.2.1), predictive monitoring "sensors" which provide real-time estimation of subject state from wearable IoT smart garment devices (Section 1.2.2), and a framework for evaluating the performance of sensor fusion of these classification and estimation "sensor" algorithms using Expectation Maximization and Gaussian Mixture Models (Section 1.2.3).

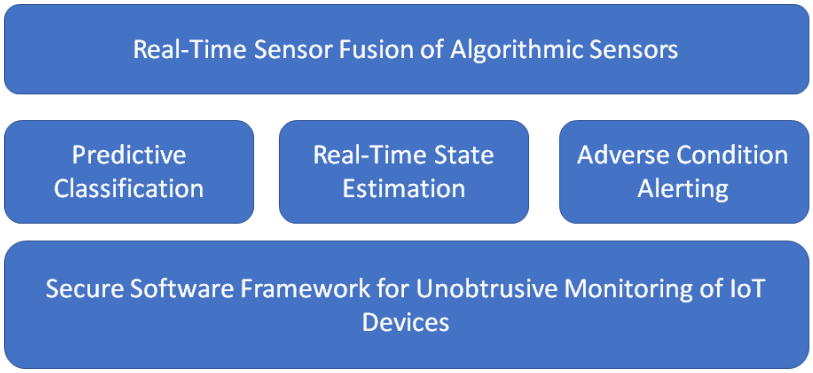

Figure 1.4: Summary and structure of thesis contributions

1.2.1 Secure Software Framework for Unobtrusive Monitoring of IoT Devices

Accurate estimation of subject biomedical state from wearable strain-gauge smart garment devices requires a novel sensor fusion approach due to the dynamic noise environment that is present both on and around the wearer. In order to carry out this sensor fusion, it is necessary to collect the data in an efficient and secure manner. Many of the physical sensors used in this study utilize UHF RFID[11], whose communication is covered by standard protocols such as the Low Level Reader Protocol (LLRP)[22]. However, because interrogation is typically done for instantaneous inventory management, the physical properties of RFID interrogation such as signal strength and phase angle are not typically stored for further processing or long-term use. Further, these physical properties could be considered medical data as they are structured in such a way as to predict biomedical state. Finally, though most of the physical sensors use RFID technology, others are traditional medical devices such as a tocodynamometer, whose data must be collected and time-synchronized to RFID data collected in parallel for comparison study purposes. To enable the efficient collection and secure storage of data from heterogeneous physical sensors, we created a software framework compatible with many physical devices and backend database systems, with a consistent communications and encryption layer for interoperability[10;18]. This software framework has been used in several RFID-based application domains for physical layer data collection and physiological classification, including this effort, movement detection for DVT monitoring, and an RFID-based energy harvesting heart monitor[23]. The heart monitoring application detects changes in the read rate of the interrogator itself as manipulated by the wearable device, so particular consideration to hard-real time constraints is implemented within the software collection framework, as described in Section 4.3.

1.2.2 Predictive Classification and Estimation of Subject State using Wearable Smart Garment Devices

Our IoT monitoring software framework enables classification of biomedical state across several applications. This study focuses on respiratory and uterine monitoring, but has also supported novel work in RFID-based heart monitoring[23] as well as correlated heart and respiratory monitoring[24].

Because we are monitoring for anomalies that may lead to adverse medical conditions for the user, it is not feasible to train traditional classifiers with representative data reflecting all possible user states. For example, we cannot train a classifier with an example heart stoppage or apnea condition, and we cannot expect to be able to train on a uterine contraction since this is the condition that the algorithm seeks to identify as it happens. As a result, classification algorithms must train with a minimal period of time to facilitate immediate deployment, and must only train with "positive" data such as a normal heart beat or respiration. We have developed classifiers to accurately discriminate between biomedical classes such as respiration and apnea in a semi-unsupervised approach [16;17], have estimated biomedical state such as respiratory rate from these classification approaches [17], and have identified anomaly conditions deviating from the short training period that may require attention and/or retraining.

1.2.3 Software Framework for Real-Time Sensor Fusion of Algorithmic "Sensors" Measured from Wearable IoT Smart Garment Devices

Dynamic environmental conditions, such as people and objects moving in the space, can cause RF interference from interrogation to interrogation that is in the same band as the signal itself. Additionally, subject "ambient" movements, such as walking or torso movements unrelated to respiratory activity, that are unrelated to the movements being monitored also appear in the same band as the signal, and create mechanical noise artifacts. A moving interrogator, such as a mobile cell-phone based interrogator, introduces similar noise artifacts as it moves through space with the RFID tag itself. These are mitigated by data fusion at the physical layer and at the application layer. For example, identification and synergy of patterns at the physical layer help inform the artifacts being sought in monitoring: if the phase and signal strength interoperate somewhat uniquely during certain activities, this can be leveraged in detection. Additionally, if certain algorithms are more robust than others under different environmental conditions, this can be leveraged if it is possible to discriminate between them. We have fused physical RFID properties at the physical layer and discriminated against analytical approaches in real-time with significant improvements in detection accuracy [25].

1.2.4 Publications In-Print, To-Appear, and In-Preparation

This section summarizes publications and awards related to and supported by this effort.

Accepted Publications

1. Sayandeep Acharya, **William M. Mongan**, Ilhaan Rasheed, Yuqiao Liu, Endla Anday, Genevieve Dion, Adam Fontecchio, Timothy Kurzweg, and Kapil R. Dandekar. **Ensemble Learning Approach via Kalman Filtering for a Passive Wearable Respiratory Monitor**. *IEEE Transactions of Biomedical and Health Informatics, July 2018.*

2. **William M. Mongan**, Robert Ross, Ilhaan Rasheed, Yuqiao Liu, Khyati Ved, Endla Anday, Kapil Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam Fontecchio. **Data Fusion of Single-Tag RFID Measurements for Respiratory Rate Monitoring**. *IEEE Signal Processing in Medicine and Biology (SPMB) 2017, December, 2017.*

3. Shrenik A. Vora, **William M. Mongan**, Endla K. Anday, Kapil R. Dandekar, Genevieve Dion, Adam K. Fontecchio, and Timothy P. Kurzweg. **On Implementing an Unconventional Infant Vital Signs Monitor with Passive RFID Tags**. *IEEE International Conference on RFID, 2017, May, 2017.*

4. **William Mongan**, Ilhaan Rasheed, Khyati Ved, Shrenik Vora, Kapil Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam Fontecchio. **On the Use of Radio Frequency Identification for Continuous Biomedical Monitoring**. *ACM/IEEE International Conference on Internet-of-Things Design and Implementation (IoTDI) 2017, April, 2017.*

5. **William M. Mongan**, Ilhaan Rasheed, Khyati Ved, Ariana Levitt, Endla Anday, Kapil Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam Fontecchio. **Real-Time Detection of Apnea via Signal Processing of Time-Series Properties of RFID-Based Smart Garments**. *IEEE Signal Processing in Medicine and Biology (SPMB) 2016, December, 2016.*

6. **William Mongan**, Endla Anday, Genevieve Dion, Adam Fontecchio, Tim Kurzweg, Yuqiao Liu, Owen Montgomery, Ilhaan Rasheed, Cem Sahin, Shrenik Vora, and Kapil Dandekar. **A**

Multi-Disciplinary Framework for Continuous Biomedical Monitoring Using Low-Power Passive RFID-based Wireless Wearable Sensors. Proceedings of the *IEEE Smart Systems Workshop 2016, May, 2016.*

7. Shrenik Vora, **William Mongan**, Kapil Dandekar, Adam Fontecchio, and Tim Kurzweg. **Wireless Heart and Respiration Monitoring for Infants through Passive RFID Tags**. *International Conference on Biomedical and Health Informatics (BHI), February, 2016.*

8. Damiano Patron, **William Mongan**, Timothy Kurzweg, Adam Fontecchio, Genevieve Dion, Endla Anday, and Kapil R. Dandekar. **On the Use of Knitted Antennas and Inductively Coupled RFID Tags for Wearable Applications**. *IEEE Transactions on Biomedical Circuits and Systems, January 2016.*

9. **William Mongan**, Kapil R. Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam Fontecchio. **Statistical Analytics of Wearable Passive RFID-based Biomedical Textile Monitors for Real-Time State Classification**. *IEEE Signal Processing in Medicine and Biology (SPMB) Symposium Poster.* Philadelphia, PA. December, 2015.

Publications In-Preparation

1. **William M. Mongan**, Ilhaan Rasheed, Robert Ross, Kapil R. Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam K. Fontecchio. **Real-Time Detection of Motion-Based Events Using RFID Tags in Wireless Wearable Garment Devices**.

2. Robert Ross, **William M. Mongan**, Patrick O'Neill, Ilhaan Rasheed, Adam Fontecchio, and Kapil R. Dandekar. **An Adaptively Parameterized Algorithm Estimating Respiratory Rate from a Passive RFID Smart Garment**.

3. Patrick O'Neill, **William M. Mongan**, Robert Ross, Ilhaan Rasheed, Kapil R. Dandekar, and Adam K. Fontecchio. **An Adaptive Search Algorithm for Detecting Respiratory Artifacts Using a Wireless Passive Wearable Device**.

4. Austin Gentry, Brent Lee, **William M. Mongan**, Owen Montgomery, and Kapil Dandekar.

Activity Segmentation Using Wearable Sensors for Deep Vein Thrombosis Condition Detection.

5. Khyati Ved, Ilhaan Rasheed, **William Mongan**, Genevieve Dion, Timothy Kurzweg, Adam Fontecchio, and Kapil R. Dandekar. **Ambient Motion Detection Using RFID-Based Smart Garments**.

Grants and Funding Support Awarded

1. Adam K. Fontecchio and **William M. Mongan**, co-Principal Investigators. Analytics on Real-Time Biometrics from Passive Wearable Smart-Garments; 2017-2018, Commonwealth Universal Research Enhancement (CURE) Formula Grant (SAP117558-014), $75,000.

2. **William M. Mongan** and Adam K. Fontecchio. Drexel University Co-op Funding Award supplement to support undergraduate experiential learning in research; 2017-2018, $7,250.

1.3 Thesis Organization

In this thesis, we detail our overview, background and motivation in Part I, in which relevant background about the technology utilized is detailed in Chapter 2. Related efforts are outlined in Chapter 3. In Part II, we detail our technical approach, including physical layer modeling (Chapter 4), feature extraction (Chapter 5), training, classification, and re-training (Chapter 6), state estimation (Chapter 7), physical layer sensor fusion (Chapter 8), multi-sensor estimate fusion (Chapter 9), and prediction of future biomedical state including interbreath interval (Chapter 10). In Part III, we detail our laboratory experimentation and human clinical trials (Chapter 11). Finally, we summarize results and conclude in Part IV in Chapters 12 and 13, respectively.

Chapter 2: Background

Wearable smart devices have become ubiquitous, with powered devices capable of providing heads-up-display information and collecting real-time biometric information from its users. Typically, these devices require a powered component to be worn and maintained, such as a battery-powered sensor[26], Bluetooth communications device[27;28], or glasses[29;30]. Pregnancy and infant monitoring devices may be uncomfortable to the mother or baby and are subject to signal loss if the patient changes position or becomes mobile because the device must remain tethered to the patient by a belt and plugged into a wall for power[31]. The unobtrusive Bellyband device is knitted into the fabric using conductive thread to which an RFID chip within the fabric is inductively coupled. This chip and antenna is interrogated by ambient sensors similar to that found in an EZ-Pass tollbooth, without requiring a power source or powered sensor on the garment. However, whereas tag interrogation is used traditionally for inventory or account management (*i.e.*, the tollbooth scenario), our work utilizes the RSSI returned from each RFID interrogation to determine different types of motion in the inductively-coupled chip and knit antenna structure as it is moved by the wearer. Movement such as a uterine contraction or an infant rolling in a crib is detected wirelessly via differences in power level perceived from the RSSI from the interrogation signal sent to the garment, as well as variable inductive coupling between the antenna and MAGICSTRAP RFID LMXS31ACNA-011 chip[32] or Monza X Dura chip[33]. The chip is interrogated by standard RFID equipment available commercially off-the-shelf, including the Impinj R420 interrogator[34] and RFMAX S9028PCLJ antenna[35]. Stretching the garment due to motion causes deformations to the knitted antenna, yielding these changes in RSSI without requiring a cumbersome transducer like the current gold standard tocodynamometer for uterine and fetal monitoring.

RFID sensors embedded within biomedical smart textiles can enable real-time, unobtrusive, passive patient monitoring in a number of settings, such as passive monitoring of uterine contractions in a pregnant woman, or respiration for apnea detection in an infant. The goal of this work is to

provide a passive, wireless solution for collecting medical feedback from these heterogeneous sensors, utilizing backscatter readings from RFID tags integrated with knitted antennas and positioned on wearable fabrics to determine distention and, from this, metrics on the rate or intensity of breathing, contractions, or other biomedical information. Several elements of this effort, including real-time capture, post-processing, and big data analytics of RFID data, rely on a software module to capture and store this data in real time from various hardware devices. However, this software module is dependent upon the type of chip being sensed and the type of interrogator being used. Moreover, it is necessary to compare this data to that collected by legacy medical equipment in a clinical trial setting in order to measure performance of and determine viability of the system. All of this necessitates a software framework for collecting data in real-time from heterogeneous medical devices and RFID sensors simultaneously, providing a consistent data representation for each.

The workflow for RFID interrogation is summarized in Figure 2.1, and the system architecture to implement this workflow is described in Figure 2.2.

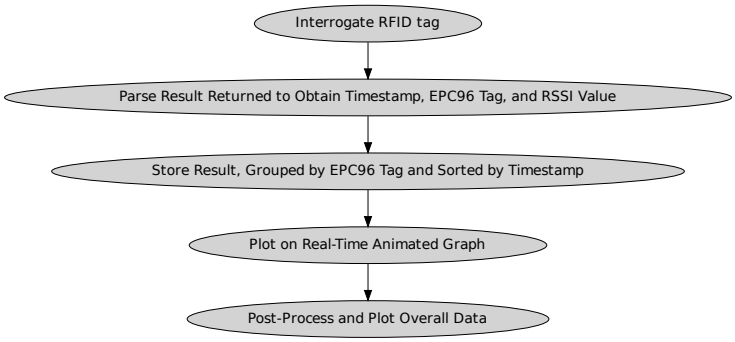

Figure 2.1: RFID Capture Workflow

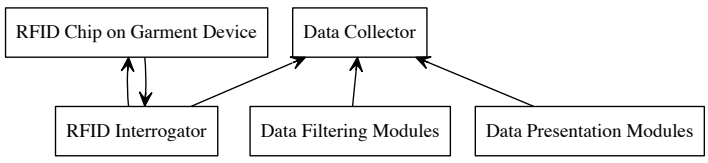

Figure 2.2: General system architecture for RFID interrogation

The result of this workflow is three-fold:

- A real-time animated plot of data collected in a fashion similar to a traditional Tocodynamometer
- A post-processed plot of data collected
- Analytics and interpolation of data collected

Each component of this workflow presents challenges. The RFID interrogator data must be obtained from specifications or reverse-engineered to determine its format and protocol. Different interrogators may communicate with different levels of compliance to the standard LLRP protocol, or they may use their own proprietary formats. Critical to these efforts is real-time software-based signal processing algorithms capable of filtering the live data to detect events such as a uterine contraction, heartbeat, or infant movement. We addressed several challenges in the workflow required to collect and present this data from the RFID interrogator in a manner consistent with "gold standard" readouts. For example, it was necessary to monitor the RFID interrogator using network packet analyzers to determine the protocol used[22], and construct an algorithm compatible with the device, modifying and adapting a library implementation of that protocol[36] for our use. Similar work was required to interface with the tocodynamometer using a legacy interface used to communicate with traditional hospital monitor devices. Additional software was written to communicate with portable RFID interrogators over a Bluetooth connection. This activity is significant because it will enable the presentation of multiple real-time sensor data collected from heterogeneous sensors over various interfaces, including a live web service interface, or Hadoop distributed data cluster. This enables data fusion and signal processing in order to determine if medical events are taking place or are about to take place, or to present a visualization of the movements sensed. Additionally, this data must be animated in a real-time display for medical use, which means that real-time processing must be limited or done efficiently to allow sufficient time for rendering. Further, the data could be collected from several RFID tags, enabling a contour map of the body depending on the tags' placement on the wearable fabric.

Each solution provided here presents further opportunities for investigation. The EPC tags, when interrogated, yield a radiated signal that is interpreted to derive medical feedback. This backscatter may constitute personal medical information protected under HIPAA guidelines. Further, the RFID interrogator will likely itself be a wireless and portable solution, enabling full mobility of the mother or baby; however, these signals must be processed to remove these movements, as they will distort the observed RSSI value. In addition, the module that communicates with the RFID interrogator will need to be migrated to a wireless network, Bluetooth, or other wireless protocol, requiring a modular software design that can communicate securely. Each consideration is addressed in this work.

This multifaceted work includes efforts to design an RFID-enabled antenna that interacts with smart garments made of heterogeneous materials using inductive coupling. Initial benchmark testing of these garments and antenna designs utilized the LLRP Commander software [37] that enables configuration of RFID interrogators for data collection. This data is then exported for post-processing. For more rapid bench testing, we sought a solution that enabled automatic configuration of the RFID interrogator, real-time visualization and processing of the data, and the ability to collect data from multiple interrogators or other devices simultaneously in a manner compatible with the visualization and processing module. This research has enabled signal capture at a range of several feet, and is conducive to mounting interrogation equipment on a ceiling or wall. This, in turn, has enabled research and development in wireless, passive uterine monitoring, fetal heart monitoring, and infant apnea detection, via an RFID-based framework for remote wireless monitoring. This software solution to support simultaneous, real-time collection and processing of RFID-based sensor data is the focus and contribution of this work.

With this software, we were able to design an experimental protocol. Our RFID-based sensors enable several medical device applications, such as uterine monitoring and infant respiration. To simulate these environments for clinical trials, we use the Laerdal SimBaby [38] as a simulator for apnea detection and a pregnant mannequin as a simulator for uterine monitoring, each containing an air bladder, and each wearing a tocodynamometer and an RFID bellyband. The baby is programmed

to execute several respiratory scenarios for detection purposes: for example, breathing at a constant rate for 3 minutes and then stopping. The bellyband and tocodynamometer are each monitored using a thread spawned by the software, and they are plotted together with optional data filtering techniques. The mannequin is actuated using a peristaltic pump that fills the bladder with either water or air to a predefined and programmable pressure or duration. These tests enable two-fold analysis: first, statistical analysis of the data enables detection of "events" (*i.e.*, respiration or a uterine contraction) that we can compare quantitatively to existing medical devices; second, comparisons between RFID chips, antenna designs, and fabric designs, by calculating the range and angle at which reliable measurements can be taken from the SimBaby or mannequin.

2.1 Repurposing RFID at the Physical Layer

Even in idealized environments with little mechanical interference and wideband interrogation, RFID backscatter follows the direct path between tag and interrogator relatively rarely [39], instead following multipath propagation about the environment. The singulation protocol for Class 1, Generation 2 (C1G2) UHF RFID [40] provides for a state machine implementation on each RFID tag and a random number generator that serves as a Slotted ALOHA protocol for collision avoidance. Tags alternate their backscatter responses to allow many or all tags in the field to respond before any one tag responds a second time. Tags "choose" amongst themselves in each round by counting down from a random number prior to responding, repeating when a collision occurs. In this way, a single tag is eventually selected for response. As a result, multipath propagation is of little consequence for inventory management because the EPC tag is eventually read from a single tag regardless of the physical path traveled by the signal itself. The physical properties of successive interrogations, however, are likely to be different. The interrogation frequency and signal paths between interrogator and tag both vary over time: the frequency usually (but not necessarily) follows a deterministic rotating pattern in order to satisfy regulatory requirements in the United States [41], and the signal path can be random and unpredictable as the physical environment surrounding the RFID tag and interrogator may also change over time. Neither a fully stochastic nor a fully deterministic model is sufficient for accurate RFID-based state estimation [39]. Therefore, we developed a hybrid model

that uses statistical multisensor fusion to inspect both deterministic models as well as probabilistic models of the physical properties of RFID backscatter response, described in Part II. Multipath effects will affect all RFID interrogations in the 900MHz band, as the signal waves are coherent for approximately 600 meters per Equation 2.1 (where c is the speed of light in a a vacuum, assumed to be 3×10^8 meters/second, B is the signal bandwidth, assumed to be 500 kHz, and n is the refractive index, assumed to be ≈ 1.00 for air at standard temperature and pressure). With these parameters, the coherence distance L = 599.83 meters.

$$L = \frac{c}{nB} \tag{2.1}$$

The Impinj Speedway R420 communicates data received over each of its four antennae via a wired ethernet connection. The Wireshark packet capture (pcap) utility confirmed that the data being sent by the Impinj unit is based on the LLRP standard[22]. The LLRP standard is extensive, but of particular interest are the trace diagrams indicating opening an RFID session with the Impinj unit (Figure 2.3), reading a tag report from the Impinj unit (Figure 2.4), and closing an RFID session with the Impinj unit (Figure 2.5). The data flow from the Impinj is shown in Figure 2.6, with each message type described in detail (including how to parse the data) as provided by the LLRP protocol specifications.

According to the LLRP Specification[22], each LLRP message has a common format, shown in Figure 2.7:

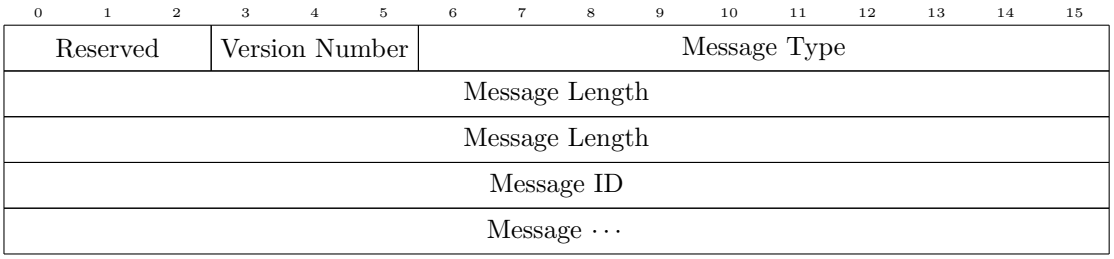

Figure 2.7: Bit-level description of an LLRP protocol message

Based on this LLRP message format, messages can be formulated, sent, read, and parsed based on

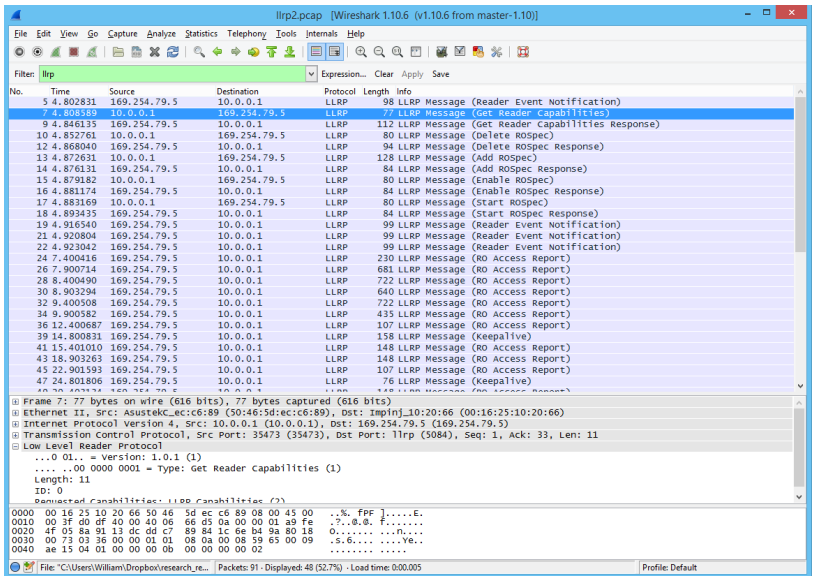

Figure 2.3: Opening an LLRP Session with the Impinj Speedway R420 interrogator, as captured by Wireshark. Notice the raw binary data to be parsed at the bottom of the figure, parsed by Wireshark for analysis.

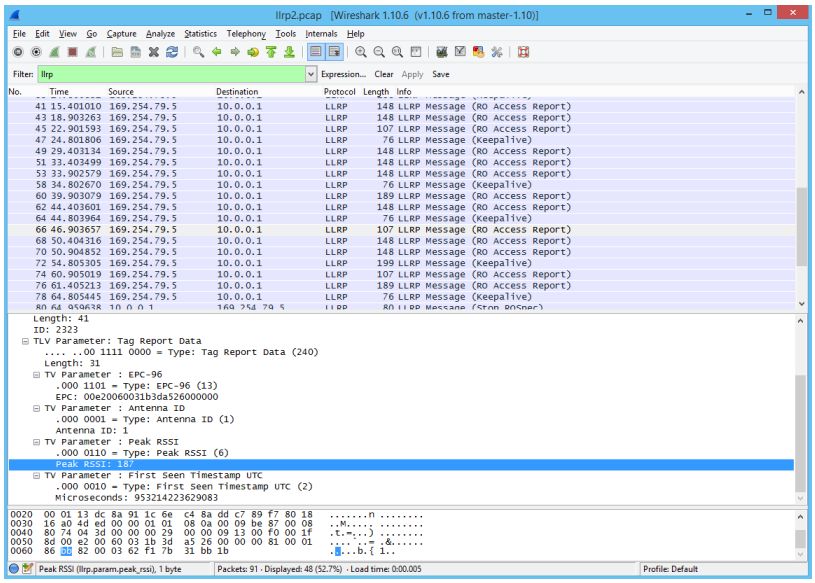

Figure 2.4: Capturing a set of Reader Operation (RO) RFID tag messages from the Impinj Speedway R420 interrogator

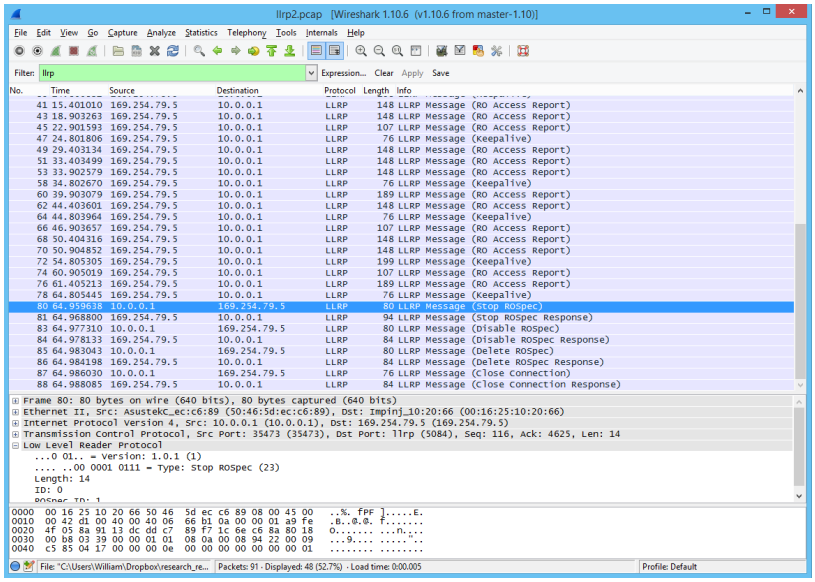

Figure 2.5: Closing an LLRP Session with the Impinj Speedway R420 interrogator

this general format, for an LLRP-compliant device. For example, the *GET_READER_CAPABILITIES* message has a message type of 1, and the *GET_READER_CAPABILITIES_RESPONSE* message type is 11. The parameters for each message type (*i.e.*, the message value) are documented in the protocol specification, and the most relevant of these messages will be illustrated here: the *RO_ACCESS_REPORT* message. This message contains the EPC96, timestamp, and RSSI value of the tags. Because TCP (the transport protocol over which LLRP is run) buffers data, these messages are queued for efficiency and sent in clusters. The timestamp that the message is received cannot be used to infer the time that the RFID tag was read; instead, the timestamp for each message is embedded for each RFID tag read being sent, and should be used instead to ensure accurate timing and spacing calculations during post-processing.

In the *RO_ACCESS_REPORT* message (message type 61), the number of tags being sent in each cluster is inferred from the message length header. Each tag report data point in the message value section consists of the fields for parsing in a loop (until the total message length has been reached) as shown in Figure 2.8:

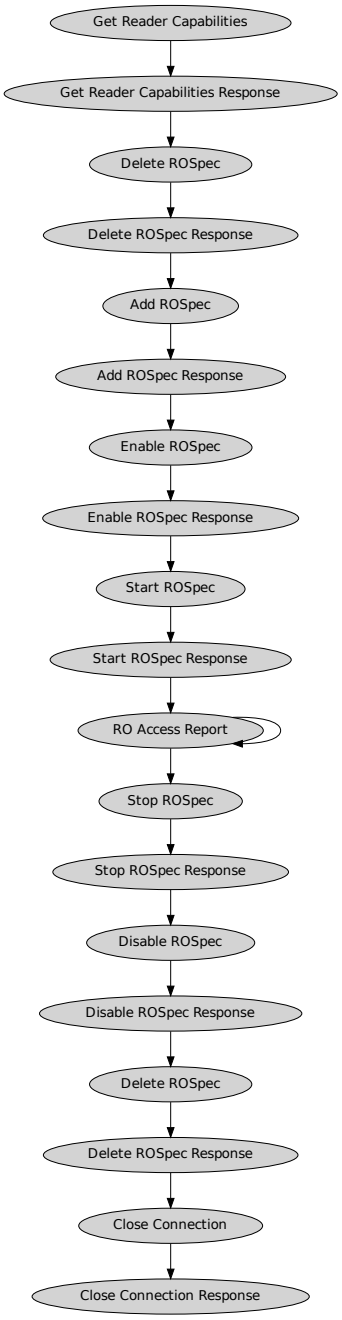

Figure 2.6: Data Flow Inferred for the Impinj Speedway R420 interrogator

Sub-message type (The *Tag Report Data* sub-message type is 240)
Length of this sub-message (this is a portion of the overall message length)
EPC96 Parameter (a 96-bit code indicating the EPC tag of the RFID chip being read)
Antenna ID (the antenna that read the tag, in case there is more than one antenna active)
Peak RSSI
First Seen Timestamp (An absolute timestamp given in nanoseconds that the tag was read)

Figure 2.8: Bit-level description of an LLRP protocol sub-message embedded within the LLRP message format in Figure 2.7

Each of these parameters can arrive in any order within the message; for parsing, each parameter is given an ID code. For example, the EPC96 parameter is code 13, and the Peak RSSI is code 6. These are all defined by the protocol specification. Thus, if a message is read with a type code of 61, it is an *RO_ACCESS_REPORT* message, in which a series of RFID interrogations can be found. For each interrogation, if a parameter of type 6 is read, the value that follows is the RSSI.

This data is read in real-time via a thread that allows the module to execute in the background while data processing and analytics are performed simultaneously. Here, we discuss the thread module that performs this protocol implementation. The LLRPyC [36] module was used as a starting point to read and parse these messages. They are returned in real-time via a callback function, and this function was implemented to add these tags to a data structure being simultaneously processed by a thread for animating the plot in real-time.

However, there were several incompatibilities between the LLRPyC module and the Impinj LLRP interrogator, perhaps because the protocol implementation is incomplete, or because the Impinj is not fully LLRP-compliant. For example, the Impinj sometimes sends data that is extraneous to the LLRP protocol or supplemental to that expected by the LLRPyC module. At other times, empty messages were returned by the Impinj, which could be interpreted as an expected *RESPONSE* message, placing the Impinj and the protocol client implementation fatally out-of-sync. Because the LLRPyC protocol included numerous checks of strict compliance, these incompatibilities occurred frequently and required adjustment to the protocol implementation module. Best practice dictates that one's software modules be lenient about their inputs (that is, forgiving to data received that

is not in strict compliance with a protocol specification, if possible), but strict with their outputs (that is, only to output data in strict compliance with its promised interface and specifications). One naive solution to this problem would be to simply ignore the extraneous message information. However, this is not feasible, because that data would later be mistaken as part of the next message. Although the protocol data units described in this Section are given friendly textual names like *GET_READER_CAPABILITIES*, it is important to remember that the actual data transmitted are the numerical encodings that represent each message type. Because of this, it is important to determine precisely where a message begins and ends. Otherwise, transmitting the length of the message would not be helpful, because the protocol client would not be able to determine where in the message to parse that length, as a result of not being able to determine where the message began. The LLRPyC module did not parse the individual parameters provided by the *RO_ACCESS_REPORT*, so this needed to be implemented in accordance with the protocol specifications described in this Section. These "extra," un-parsed parameters are likely the reason for the protocol client failing on the incoming messages. The protocol client was modified to read this data, parse those parameters necessary for this study, and store the rest without further processing. Once this was completed, the LLRP protocol client module could be used to collect data in real-time for processing. The following sections describe in detail how this module was used to perform real-time processing, post-processing, and analytics on the RFID data collected. We define real-time processing as the ability to collect and store data at a higher rate than the interrogation read rate, and the ability to process the data within a specified time constraint (*i.e.*, respiratory rate detection over a 6 second window).

2.1.1 Integrating Analysis Modules with the Data Collection Framework

We construct an algorithm compatible with the device communicating over the LLRP protocol, modifying and adapting a library implementation of that protocol[36] for our use. Similar work was required to interface with the tocodynamometer using a legacy interface used to communicate with traditional hospital monitor devices. Additional software was written to communicate with portable RFID interrogators over a Bluetooth connection to enable communication with wireless interrogators. The data are time-correlated so that events can be compared between devices. Further, the data

could be collected from several RFID tags, enabling a contour map of the body depending on the tags' placement on the wearable fabric.

This data is read in real-time via a thread that allows the module to execute in the background while data processing and analytics are performed simultaneously. The LLRPyC[36] module was used as a starting point to read and parse these messages. They are returned in real-time via a callback function, and this function was implemented to add these tags to a data structure being simultaneously processed by a thread for animating the plot in real-time.

There were several incompatibilities between the LLRPyC module and the Impinj interrogator, perhaps because the protocol implementation is incomplete, or because the Impinj is not fully LLRP-compliant. For example, the Impinj sometimes sends data that is extraneous to the LLRP protocol or supplemental to that expected by the LLRPyC module.

To facilitate interrogation of multiple tags simultaneously (for example, to create a contour of body movements), the software was modified to also parse and store the EPC96 tag from each read event message. The dictionary was grouped by EPC96 tag and sorted by timestamp before plotting on the animation and graph. This resulted in multiple sets of coordinates to be plotted separately. The animation thread plots each set of coordinates separately, and new tag events are associated with their existing set of coordinates. If a new RFID chip is introduced during the animation sequence, it is dynamically added to the list of known RFID tags, and a new set of coordinates are plotted alongside the existing ones in real-time. For readability, each set of RFID EPC tag data is plotted in a different color, and a legend is provided on the graph to indicate which line corresponded to which EPC tag ID. We are able to detect, read and process four RFID chips simultaneously (more is likely feasible at the expense of the interrogation sample rate, but our investigation has focused on single-tag deployments).

Chapter 3: Related Work

In this chapter, we summarize the state of the art in four synergistic and related areas to this effort. We outline existing work in smart wearable biomedical monitoring devices in Section 3.1. RFID-based tracking and monitoring systems are surveyed in Section 3.2. Data collection and processing systems for wearable sensor monitors are described in Section 3.3, and, finally, we survey Machine Learning approaches in RF wearable systems in Section 3.4. These efforts provide a framework for synergistic intelligent systems in heterogeneous IoT wearable biomedical systems; we build upon these efforts by providing for unsupervised models for real-time learning and classification of wearer state using knitted antennas for RFID-based applications.

3.1 Survey of Wireless Wearable Monitoring Devices

Conductive materials have a variety of uses as wearable textile devices, including data channels and sensors, wearable antennas, as well as in protective clothing materials[42]. Flexible antenna designs enable embedding wearable sensors about the human body; Xu, *et al*, do this by embedding the flexible antenna upon a felt strap suitable for integration into a wearable garment[43]. Even passive sensor smart garments may require a portable processing unit or power source, but have the potential to sense the environment and user state as well as inform active sensors that can actuate directly on the human body[44]. An integrated patch antenna is subject to strain forces that influence its S_{11} characteristics[45]; the wearable sensor in this study is a knitted antenna using conductive thread deployed about a passive RFID tag embedded into the garment in an unobtrusive manner that requires no external power source aside from the interrogation signal itself[11;46]. This knitted antenna takes advantage of the changes in resonant frequency due to strain forces to create a smart garment device that can monitor stretching as well as coarse movement activity.

Other wireless sensors are attached to the user during typical use, but not necessarily integrated into an unobtrusive day-to-day garment for use without awareness of its deployment. A tracheal

sensor attached with medical tape can be used to detect snoring spells that could lead to apnea conditions[47], using acoustic analysis with 89% accuracy.

It has been shown that wireless antennas embedded into garments can be used to monitor chest distension due to respiratory activity using a prototype antenna connected to a network analyzer for monitoring the frequency shift of the antenna due to changes in dielectric properties of the body during respiration[48]. These frequency shifts can range from 4-15 MHz to 120 MHz depending on the depth of inspiration. Our sensor must iterate over frequencies in the 900 MHz band in order to utilize commercially available monitoring equipment, so we use signal strength and other physical RF properties; as a result, we must filter noise artifacts due to the antenna proximity to the human body. Wireless reflections using 802.11 Wi-Fi protocols can also be utilized for subject monitoring using the WiBreathe system, if a transmitter and receiver pair are available in the environment and a single subject is being monitored[49], and for up to 7 subjects in the same room with a high degree of accuracy using the MUSIC algorithm on a set of subcarrier gains observed on the Wi-Fi channel[50], and for multiple subjects as long as they remain 1.5 meters apart[51]. However, like WiBreathe, we will also employ both discrete and spectral analysis for respiratory rate estimation, and fuse those signals into a unified estimate.

Tarassenko and Mason[52;53] use active tags such as ECG and photoplethysmography (PPG) to infer respiratory activity using autoregressive (AR) modeling. Our effort seeks to use a single passive RFID sensor to monitor strain movements such as respiration; the wireless signal is too noisy to use AR modeling directly, and so a stochastic model is employed instead for this purpose.

3.2 Survey of RFID-Based Subject Monitoring

Common physical features reported from commercial off-the-shelf RFID equipment include RSSI, phase angle, Doppler shift, and received timestamp. Although these features are sometimes quantized, and are subject to perturbations due to multipath fading and shadowing as the reflected signal traverses an irregular and dynamic environment, they can be monitored over time to estimate subject state. Often, this is done by fusing features collected from multiple interrogators or from multiple tags placed in the environment. Doppler and RSSI have been used to identify gross hu-

man movements such as walking around a room with an RFID tag[54], and are used to detect those movements in a device-free environment in which the tags are deployed in the environment, and not on the person[54;55]. To overcome the interrogator noise inherent in the Doppler measurement, a fixed RFID interrogation frequency was used[54]; however, this approach is not feasible in the United States under Federal Communications Commission (FCC) regulations[41]. Although these limitations do not exist elsewhere in the world, and can be overcome with custom interrogation equipment and protocols[56–59] or other physical layer infrastructure such as Zigbee[60] or microcontroller/Software Defined Radio (SDR)[61;62], we restricted ourselves in this effort to minimal commercially available equipment. Shadowing due to human movements between the tag and interrogator were detected by observing changes in the read rate[54]. Observing changes in tag read rate across multiple tags is a feasible feature in a device-free application in which the tags are deployed to the environment itself because their configuration is likely to be static; in our on-human single-tag application, the read-rate is not likely to be constant due to dynamic multipath fading artifacts as well as the presence of additional tags entering and leaving the field. Further, we have chosen a non-device-free deployment of a tag about the human body to facilitate rapid deployment in any environment, rather than restricting ourselves to a fixed room or building; additionally, we seek to analyze the fine strain gauge movements of the knitted antenna itself rather than (or in addition to) the coarse movements of the wearer.

IDSense[63] is a user activity tracker that classifies behaviors (interactions with objects in the environment) using RFID signal strength, phase, and read rate. Activities were trained on features of RSSI, Doppler, and Phase, and then classified into five activities using a Support Vector Machine. This effort requires tagging all the objects in question, and a supervised training period, but were able to classify between 68% and 98% accuracy. A similar system uses an array of passive RFID tags and the root mean square (RMS) of phase differences among them to classify handwriting gestures in the air[64]. The system is trained on feature thresholds to classify a particular handwriting gesture. Related systems for customer shopping behavior analysis use phase difference to determine if a tagged item was moved from a shelf to a shopping cart[65], and use RSSI and Doppler window probability

distribution changes to track tagged items and customer interactions[55]. Phase and Doppler-based tag velocity is a useful feature for knitted antenna strain sensing, even though coarse movements are intermixed with this signal. Bhattacharyya, *et al*, use a similar RFID physical model to create an RFID strain sensor using a controllable frequency sweep[66]. Multipath artifacts present as the strain sensor is stretched of 1.5 dBm to up to 3 dBM[66], but these backscatter artifacts are left as future work. We do not enjoy frequency-controlled interrogation; however, we do use a stochastic model to mitigate multipath artifacts. Volk, *et al*, use High Frequency (HF) RFID to monitor vital signs in rodents using an implanted RFID tag[67]. In our work, we utilize an external tag knitted via an unobtrusive knitted antenna, and we chose UHF RFID for its added read range and read rate over HF[68].

When multiple interrogators and/or tags are deployed in the environment, coarse-grained motion artifact tracking is facilitated[55;69] using kernel-based methods such as a Support Vector Machine (SVM)[69–711] or multi-layer perceptrons such as Artificial Neural Networks (ANN)[73;74]. This is often accomplished by deploying reference interrogators or tags in the environment at known locations[75] and using differential RSSI measurements[76–79], differential Doppler shift analysis[80], differential phase/Angle of Arrival (AoA) analysis[81–83]. Specifically, NightCare[75] uses tags placed in the environment near the user to infer movement or presence in the room for sleep patterns, sleep apnea, and bedsores, but not real-time respiratory measurements such as respiratory rate. Fixed environmental configurations are preferable when performing RSSI-based monitoring, because RSSI is subject to variation due to the reflective properties of different materials that may be present in the environment[84], ambient environmental temperature[85], as well as the orientation of the tags[86]. If the antenna is fixed, changes in phase angle can be used to infer the distance between the tag and antenna; again, multiple interrogators and tags are typically deployed to resolve the ambiguity of phase angle changes beyond $[-\pi, \pi)$[87], which occur at distances beyond $\frac{c}{2f} \approx 16cm$[88] for interrogation frequency f and speed of light in a vacuum constant c. We do not enjoy a broad sensor network to inspect, nor do we wish to train on a known environmental configuration such as sen-

[1]SVM-based localization has also been enabled by classifying RSSI from Wi-Fi signals[72].

sor placement; additionally, we seek to limit providing supervised training samples to our learning algorithms. Nevertheless, we employ kernel-based methods including SVM to separate data. In an SVM, each training sample is used to additively perturb a perceptron separation function. We explore alternative semi-supervised training to enable the separating capabilities of the SVM within these constraints in Sections 6 and 6.6.

3.3 Survey of Machine Intelligence and Data Collection Systems for Wearable Sensor Monitors

Classification problems typically involve kernel-based separation[69] and hypothesis testing[89]. Seismic activity was classified using a matched filter[89], in which a reference signal is compared using hypothesis testing to the input signal. Although we will perform maximum likelihood hypothesis testing for classification, matched filtering was a challenge in this application because of the variability in the signal presented during human activity. For example, a brief decoupling of the knitted antenna from the tag due to a large stretch or short from a piece of metallic thread causes the sinusoidal stretching pattern to appear as multiple rapid peaks. A low-pass filter removes many of these artifacts, but sacrifices signal amplitude which would aid in classification by a matched filter.

In addition to monitoring physical sensors for biomedical feedback, it is necessary to communicate and present this information in a manner useful to patients and care providers. Telemedicine refers to the communication of biomedical feedback and medical records between and among providers and patients for the purpose of health maintenance. In this effort (see Section 4.2), we develop data collection, visualization, and integration modules conducive for the remote monitoring of biomedical feedback using wearable sensor devices as well as traditional medical monitors. Telemedicine systems include integration of body sensor networks and real-time communications to a hospital setting[90]. Home use of reserved telemedicine frequency bands is restricted by FCC regulations[90] due to the risk of RF interference from home appliances; however, we use the unlicensed 900 MHz band for RFID-based sensors which, despite the risk of interference artifacts, are conducive for home use.

eCloudRFID[91] is a data collection and presentation framework from RFID sensor networks to interface with external application layer components. Our framework is focused on bridging the

physical layers of heterogeneous sensors, including but not limited to RFID devices, and storing them for processing in a HIPAA-compliant manner.

3.4 Survey of Machine Learning Approaches for RF-Based Monitoring

As we discuss in Chapter 5, it is necessary to aggregate the physical properties of RF backscatter data into higher order features that can be baselined, compared, or modeled for wearer state classification. Some features considered in the literature using RF properties include the number of frequencies above a given power magnitude threshold, the mean and standard deviation of the power spectrum, the minimum or maximum power observed in the spectrum, and the mean or standard deviation of the underlying signal in the time domain over a given window duration[92]. The features are aggregated into a reduced dimension feature space using PCA, and then clustered using an unsupervised method such as k-means clustering to identify one or more states for each data point in the window[92]. Although our underlying signal suffers from noise artifacts and quantization that yields low separability for fully unsupervised classification, and our window size must be kept short to maintain real-time classification performance for biological applications, we will augment classic classification algorithms (see Chapter 6) to allow for unsupervised classification of passive RFID backscatter properties using aggregate features including these.

Many signal processing algorithms are deployed to denoise environmental artifacts present in RF signals, which can enable a signal better characterized by the mechanical movements introduced by the wearer and monitored for physiological state classification. For example, a reference tone composed with a signal can be used to mitigate jitter effects, introduced by interrogator instability in the sampling clock or analog-to-digital converter[93], in the received signal by measuring the phase noise present in the reference frequency band[94]. Although such a reference signal cannot be injected with commercial off-the-shelf RFID interrogation equipment, such a method could be integrated into an SDR custom implementation of the RFID protocol for baseline experimentation against existing RFID equipment.

Other RF machine application domains include classifying the expected feature ranges for an interrogator for security purposes[95], to determine whether the system is communicating with a

genuine interrogator within the sensor network. The classification algorithms are similar to our approach, but we are classifying using a non-fixed RFID tag as well as a flexible knitted antenna whose shape and reflective properties change as the wearer interacts with it. We are not able to cluster features into known classes, and must therefore utilize temporally evolving features for classification. In addition, we must dynamically and synthetically generate training data for the classifier, since it is infeasible to induce biological anomalies for the purposes of training a monitor intended to detect or mitigate those anomalies.

Additionally, deep learning models such as ANN and Convolutional Neural Networks (CNN) are used to automatically classify data including RF properties. Although ANN is a powerful structure that adaptively learns a model from sample data and is robust to environmental variations such as those commonly seen in our application domain, one disadvantage of ANN is that the underlying model generated by the ANN is not directly tied to physical phenomena being observed by the data features. It is challenging to quantify the replicability of ANN performance in a physiological domain since a different model can be generated for every deployment instance. Still, classification using ANN and deep learning models has been used with RF data with success[96], but the ANN may train on the noise in a low SNR environment such as our passive RFID application domain. We utilize ANN in limited applications in which dynamic environmental training is beneficial, but the underlying model can be related to the features in a replicable manner, such as instantaneous ambient movement detection (see Section 6.5).

Part II

Approach

Chapter 4: Modeling of RFID-based Biofeedback

By FCC Part 15 regulations in the United States, UHF RFID in the 900 MHz band must iterate between each of 50 channelized frequencies between 902 and 928 MHz in increments of 500 kHz [41]. This is done to prevent collisions and unwanted interference in the band which is shared amongst unlicensed uses. Generally, interrogators must change frequencies every 200 milliseconds. Unfortunately, the physical properties of RFID backscatter response such as signal strength and phase angle are affected by the interrogation frequency, because the knit antenna is resonant at a particular frequency within the band. Stretching the antenna changes this resonant frequency, which changes the signal backscatter properties, but changing the interrogation frequency also changes these properties. To reduce perturbations due to signal strength and phase, the physical model equations incorporate the interrogation frequency for the RSSI (see the Friis Transmission Formula in Equation 4.1 [97]) and the phase (see Equation 4.2 [98]), where:

- $P_{Rx,reader}$ is the calculated power received at the interrogator given a constant environment,
- $P_{Tx,reader}$ is the interrogator transmit power (configured to be 1 Watt),
- G_{reader} is the reader gain (assumed to be constant),
- G_{tag} is the tag gain (which can change over time with the shape of the knit antenna),
- λ is the interrogation wavelength ($\frac{1}{f}$, given an interrogation frequency f),
- r is the interrogation radius (which can change as the subject moves in space with the tag),
- R is the return loss over the interrogation path,
- v is the tag velocity,
- c is the speed of light in a vacuum,

- f_m is the Doppler shift, or change in phase angle, observed in two successive tag interrogations, and
- α is the interrogation angle (which can change over time as the tag moves in space).

$$P_{Rx,reader} = P_{Tx,reader} \times G_{reader}^2 \times G_{tag}^2 \times \left(\frac{\lambda}{4\pi r}\right)^4 \times R \tag{4.1}$$

$$v = \frac{c \times f_m}{2f \times cos(\alpha)} \tag{4.2}$$

We observe that r, G_{tag}, R, v, and α are dependent upon wearer configuration and are unknown to us. The rest, however, can be configured or remain constant during use. Unfortunately, changes in individual terms are independent of changes in other terms, in that they can be triggered by different types of wearer changes in configuration. For example, r varies with the distance to the interrogator, but G_{tag} varies as the knit antenna is stretched. Using multiple RFID tags including a reference tag on a relatively stationary part of the body, these terms can be isolated; however, it is assumed for this effort that these inter-relationships remain unknown and must be observed stochastically. Therefore, we group these varying terms together to form corresponding measurements in Equations 4.3 and 4.4.

$$\hat{\zeta} = \frac{r^4}{G_{tag}^2 \times R} = \frac{P_{Tx,reader} \times G_{reader}^2}{P_{Rx,reader}} \times \left(\frac{\lambda}{4\pi}\right)^4 \tag{4.3}$$

$$\eta = v \times cos(\alpha) = \frac{c \times f_m}{2f} \tag{4.4}$$

For phase measurement, the Impinj R420 reports both the phase angle and instantaneous Doppler shift. This is advantageous because computing the Doppler from the phase (denoted by ϕ) requires computing the change in successive phase measurements, which assumes that the same antenna and interrogation frequency were used between those interrogations. Phase-based velocity estimation (shown in Equation 4.5) computes a more accurate velocity estimate[55;99] since it incorporates the time (t_x) between those successive interrogations, whereas the reported Doppler shift is subject

to noise artifacts at the interrogator[54]. However, we use the Doppler measurement reported by Impinj to avoid the restrictions imposed by requiring a constant interrogation frequency, tag, and interrogator antenna, since the interrogation frequency changes at a rate of 5 Hz per FCC regulations described in this Section. Per Impinj recommendations[98], we choose a slower interrogation rate to increase Doppler shift resolution and, in turn, velocity estimation accuracy[18].

$$v = \frac{c \times (\phi_{t2} - \phi_{t1})}{f \times (t_2 - t_1)} \tag{4.5}$$

Multiple antennas allow for Direction of Arrival analysis via phase difference[100], but we sought to minimize infrastructure for a wearable device that could then be scaled to multi-tag and interrogator environments.

4.1 Signal Model

RFID received backscatter power and "tag velocity" are, respectively, calculated via Equations 4.1 and 4.2. It is assumed that each measurement is subject to additive 0-mean stochastic noise term $\epsilon(t)$, resulting from quantization or measurement observation error. Additionally, the backscatter power measured by $\hat{\zeta}$ is subject to stochastic multipath fading. This fading term can be modeled with a Rician distribution (or, more generally, a Rayleigh) if the variance of the multipath is known[101]. However, we assume that the tag will be worn by the user right away, so that measurement of the properties such as the phase shift of the backscattered signal is undesirable; these measurements are likely to change upon use as the wearer moves about in the space. Additionally, some commercial RFID interrogators such as the Impinj R420, used in this study, quantize the RSSI component of $\hat{\zeta}$ such that the amplitude and variance between inferred paths is likely to be observed as ± 1. Because the channel properties change with the shape of the knitted antenna, the distance between the tag and the interrogator, as well as due to multipath fading, all of which is quantized by the interrogator, we cannot reliably classify each data point into a backscatter path *a priori*. When the properties of the multipath environment are known, such as might be the case in a fixed environmental configuration, the RSSI properties from each multipath propagation can be modeled[102].

Additionally, fluctuations in RSSI can be modeled with respect to the distance between the tag and interrogator as well as the distance to the groundplane [103]. A log-normal distribution has been used when the environmental configuration characteristics can be measured *a priori* to estimate the path loss exponent [104]; however, because we use a dynamic environment in which the tag moves with the wearer, we use other stochastic feature modeling approaches such as a Markov Switching Model, Hidden Markov Model, and Markov Chain Monte Carlo, to classify the state of the channel in real time.

The tag gain components G_{tag} and G_{reader} are elaborated in Equation 4.6, where A is the effective antenna aperture of the interrogator or the tag due to the Friis Transmission Formula:

$$G = 4\pi A\lambda^{-2} \tag{4.6}$$

The remaining factors in Equation 4.6 are constant factors or related to the interrogating frequency, which is constant between the interrogator and the tag. The effective aperture A_{reader} of the interrogator remains constant, while that of the tag and knitted antenna (A_{tag}) changes over time as the antenna is stretched and returned. Combining Equation 4.6 with Equation 4.3, we obtain Equation 4.7:

$$\begin{aligned} \hat{\zeta} = \frac{r^4}{(4\pi\lambda^{-2}A_{tag})^2 \times R} = \frac{P_{Tx,reader} \times (4\pi\lambda^{-2}A_{reader})^2}{P_{Rx,reader}} \times \left(\frac{\lambda}{4\pi}\right)^4 \\ \hat{\zeta} = \frac{r^4}{A_{tag}^2 \times R} = \frac{P_{Tx,reader} \times A_{reader}^2}{\lambda^4 \times P_{Rx,reader}} \end{aligned} \tag{4.7}$$

Using Equation 4.7, we observe that a change in frequency across adjacent 500 kHz channels results in a constant factor increase in $\hat{\zeta}$ of $\xi \approx 0.009$, for all channel increases from 0 to 49 (the constant factors revert back on a channel iteration from 49 to 0, so we only consider the constant factor present in channel iterations 0 to 49). Here, we include the conversion from Watts to dBm

units, which we denote by $\hat{\zeta}'$.

$$\begin{aligned} \hat{\zeta}' &= 10log_{10}\Big(\Big(\lambda^4 \times 10^{0.1\times P_{Rx,reader}}\Big)^{-1}\Big) \\ \hat{\zeta}'_{t2} - \hat{\zeta}'_{t1} &= -10log_{10}\frac{f_{t1}^4}{f_{t2}^4} \\ f_{t2} &= (f_{t1} + 500kHz), f < 927.25MHz \\ \xi = \hat{\zeta}'_{t2} - \hat{\zeta}'_{t1} &= 10log_{10}\frac{f_{t1}^4}{(f_{t1}+500kHz)^4} \approx 0.009 \end{aligned} \tag{4.8}$$

$P_{Rx,reader}$ (RSSI) is quantized to the whole number, so this residual term is not modeled precisely and results in a small sawtooth pattern in $\hat{\zeta}$ that repeats as the interrogation frequency iterates from 927.25 MHz back to 902.75 MHz (channel 49 to channel 0). Despite the small magnitude, because we are observing oscillatory patterns in the data, it is desirable to remove any known oscillatory patterns. Therefore, we correct the calculation of $\hat{\zeta}$ per Equation 4.9:

$$\zeta = \hat{\zeta} + \xi \times \lfloor (49 - 2 \times \frac{f - 902.75 \times 10^6}{10^6}) \rfloor \tag{4.9}$$

These corrected physical measurements are used in Chapters 8 and 9 to perform time-series analysis and estimation of subject state using a wearable RFID-based smart garment device. Here, we focus on structuring the ζ and η physical signals for strain gauge analysis. ζ is already highly quantized to integers along the dynamic range, which is often under 5-10 dBm of RSSI. Further, both the tag radius and effective aperture are subject to perturbations due to motion artifacts, introducing a factor of r^4 and A_{tag}^{-2} into the same signal; often in our efforts, one factor is a noise artifact while the other represents the desired signal. RFID-based inventory applications are not concerned at the application layer with these perturbations, because the EPC tag component is the desired signal and can arrive over any path; ranging applications can use fixed RFID tags and thus hold constant the effective aperture component in favor of the radius. Machine learning techniques are needed to inspect the aggregate ζ signal in real time to separate motion artifact components from one another.

The tag velocity η signal uses the reported Doppler shift to compute the instantaneous tag velocity. Small shifts in phase are reported more accurately by the Impinj in some applications[55], and so it is desirable to use the phase for this calculation instead. However, this would require discarding records except for those of the same interrogation frequency, because this also perturbs the phase observation and subsequent velocity calculation. In other countries, continuous interrogation at a single frequency is possible, and so a phase-based observation is facilitated[55]. When the read rate is large (*i.e.*, 25 Hz or more), a phase-based velocity measurement is used since most records will use the same interrogation frequency; however, it is time prohibitive to wait the 10 seconds that would be required to complete a full cycle through all interrogation frequencies for this purpose, and the Doppler-based velocity measure is typically favored for this application.

For fixed-tag applications, the tag delay spread can be used as an indicator of multipath propagation[105], because these different reflective paths require different echo time durations to arrive at the interrogator. Tag responses with a round trip time within $\frac{1}{2}\widehat{\sigma_{\Delta t}}$ on either side of the mode of all observed Δt, with $\widehat{\sigma_{\Delta t}}$ defined as the standard deviation of those values of Δt within one standard deviation of the overall mean Δt.

Spurious spikes are possible due to power loss resulting from taking a non-ideal path in a multipath environment, or due to unwanted mechanical noise artifacts from ambient movements. To eliminate these spikes, all peak amplitudes are identified in the window, and those power spikes smaller than the first quartile of all power spike amplitudes, that do not correspond with a velocity change greater than the first quartile of all velocity spike amplitudes, are removed.

This smoothed signal is converted to a square wave using classification via Fisher Linear Discriminant Analysis (LDA)[106]. The resulting ζ and η are arranged as tuples in $\mathbb{R}^2$ and separated into stretching data points and non-stretching data points. Although there is some variation from theoretical physical signal properties due to slight variations in physical characteristics among RFID tags, as well as variations between human subjects in their exact movements[54], we wish to avoid fully supervised training because a semi-automated approach is desirable, and because full supervision of the start and endpoints of the observed activity, such as respiratory behavior, would be difficult.

However, LDA requires training classifications, so we employ a semi-unsupervised approach. The first 30 seconds of data is observed and assumed to be "normal" respiratory activity of unknown parameters.

To provide this training data, a Hidden Markov Switching Model is applied to these 30 seconds of data to search for the centroids of the stretching and non-stretching records. A Hidden Markov Switching Model assumes that data being considered falls into N distinct probability distributions or "regimes." In our case, we assume that these are Gaussian distributions, and that there are $N = 2$ such regimes to be considered (stretching and non-stretching). A Markov Switching Model works by estimating the properties (*i.e.*, the mean μ and the standard deviation σ) of these distributions, and then predict with what probability a data point fits into each distribution. One advantage of a Markov Switching Model is its Markov property: the classification of the previous data point is considered when classifying the current data point, so some temporal dependency is considered. This is advantageous in our applications because temporal persistence in each state is assured: if one takes a breath, it will cause a relatively prolonged stretch of the band relative to the interrogation rate. Therefore, a Markov Switching Model constructs a $2^N \times 2^N$ probability state transition matrix that describes the probability of switching from any one regime to any other (including the current) regime. Each probability is the Bayesian likelihood $P(\theta|x_t, x_{t-1})$ for regime θ and data point x_t. The probability distribution of x can be computed easily by observation of the mean and standard deviation of the population. For $N = 2$, we also have $P(\theta = 0) = 1 - P(\theta = 1)$ by the law of total probability. A search using Gradient Descent via the Maximum Likelihood product of the probability chain is used to find an optimal or approximately optimal set of distributions for the conditional probabilities $P(\theta|x_t, x_{t-1})$. Gradient Descent over the distributions' properties according to their maximum likelihood approaches the ideal distribution parameters as the number of Gradient Descent iterations increases by maximizing the probability of classifying a given data point to either regime in that iteration.

Subsequent data windows are fit to this model to obtain the transition points between stretching and non-stretching activity. ζ and η tuples are separated into these two categories according to their

classification by the Hidden Markov Switching Model. Because a Markov Switching Model separates data into two classes without assigning meaning to those two classes, class 0 is arbitrarily defined for consistency to be that class whose mean value of ζ is smaller in that window. The Markov Switching Model produces two distributions of observed data, roughly into two classes (stretching and non-stretching). The multivariate separability between the two classes is defined by Fisher's LDA measure in Equation 4.10, for covariance matrix Σ_θ and mean μ_θ in class θ.

$$s = \frac{(\Sigma_0 + \Sigma_1)^{-1}(\mu_1 - \mu_0) \cdot (\mu_1 - \mu_0)^2}{((\Sigma_0 + \Sigma_1)^{-1}(\mu_1 - \mu_0))^T(\Sigma_0 + \Sigma_1)((\Sigma_0 + \Sigma_1)^{-1}(\mu_1 - \mu_0))} \tag{4.10}$$

Using Equation 4.10, we determine the likelihood ratio of a data record belonging to one or another of these distributions. Classifications result in a square wave that approximates the desired signal from the physical observations of ζ and η. The likelihood of classifying into class θ given data record x and Markov predicted class centroids $\hat{\theta}$ is given in Equation 4.11.

$$l(\theta|x;\hat{\theta}) = \sum_{\theta=0}^{1}(-1)^\theta((x - \mu_\theta)^T\Sigma_\theta^{-1}(x - \mu_\theta) + ln(\Sigma_\theta)) \tag{4.11}$$

The likelihood scores from Equation 4.11 are evaluated via a sigmoid function (Equation 4.12) centered around an offset $\hat{\mu_1}$ equal to the mean of those scores computed for data records that were classified into class $\hat{\theta} = 1$ by the Markov Switching Model.

$$s(x;\hat{\mu_1}) = (1 + exp(-x + \hat{\mu_1}))^{-1} \tag{4.12}$$

Additionally, a cumulative density function p is defined via a histogram for those likelihood scores classified by the Markov Switching Model into each class: $l(\theta|x;\hat{\theta} = 0)$ and $l(\theta|x;\hat{\theta} = 1)$. The likelihood ratio of the Markov Switching Model classifying a given score into one class or another measures the confusability between the two classes. This ratio is given via Logistic Regression in Equation 4.13.

$$r(\theta|\hat{\theta}) = (1 + exp(-ln(p(\theta|\hat{\theta} = 1)(1 - p(\theta|\hat{\theta} = 0))^{-1})))^{-1} \tag{4.13}$$

These likelihoods s and r are combined by taking their product. Rather than apply a simple threshold *a priori*, as is typical in discriminant modeling and analysis[107], the changes in classification are identified dynamically by inspecting peaks of the signal formed from the product rs. Dynamic classifications allow tolerance for changing environmental conditions without specifying a threshold for depth of stretch given the distance from the interrogator and a complete model of the environment. Strain gauge activity such as respiratory behavior causes likelihood transitions that shift from one class to another throughout the duration of the stretch. Because we are already able to determine lack of stretching activity in the window (see Chapter 6) and changes in underlying environmental conditions (see Section 6.6), we can ignore these cases here and identify the shifts in likelihood classification to form a square wave from the backscatter strength signal ζ and the velocity signal η.

4.2 Software Infrastructure

For comparison testing across interrogation devices and RFID garment devices, it is preferable to obtain data from existing devices in a format compatible with our RFID time-series data. Where possible, digital data is collected using client driver software modules from devices such as the Philips 50XM fetal monitor, and emitted for graphical display or statistical analysis as described for our RFID-based sensors. Side-by-side data plots from the RFID reader and 50XM tocodynamometer are shown in Figure 4.1, with a comparison of filtered data between the two devices shown in Figure 4.2.

The complete monitoring solution itself is completely wireless and requires no external battery source. The architecture of the physical deployment solution, including simulation equipment, is described in Figure 4.3. Here, a Raspberry Pi contains the **Client** and **Server** packages, as well as the **Database** backend. It drives the *RFID Interrogator*, *Tocodynamometer*, or other interrogation device monitoring a subject, mannequin, or test device, over ethernet or a serial connection. This interaction is encapsulated into each **Client** subclass, so that the resulting data is a stream of time-series values corresponding to the interrogation measurements to be dispatched to the **Server**. Groups of time series data are sent to the server and database using an HTTP PUT body message, detailed in this Section. A second Raspberry Pi contains the **Visualizer and Detector** modules

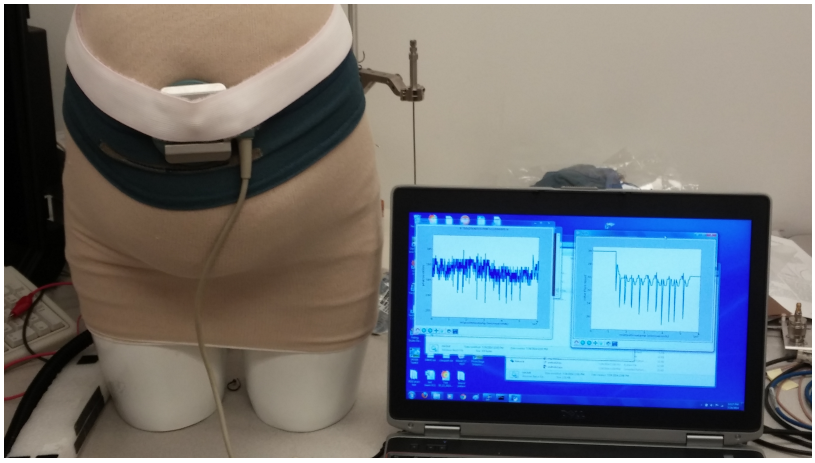

Figure 4.1: A tocodynamometer pressure sensor is placed above a Bellyband on a "pregnant mannequin" and monitored using our software (the RFID data plot is to the left, and the tocodynamometer data plot is to the right).

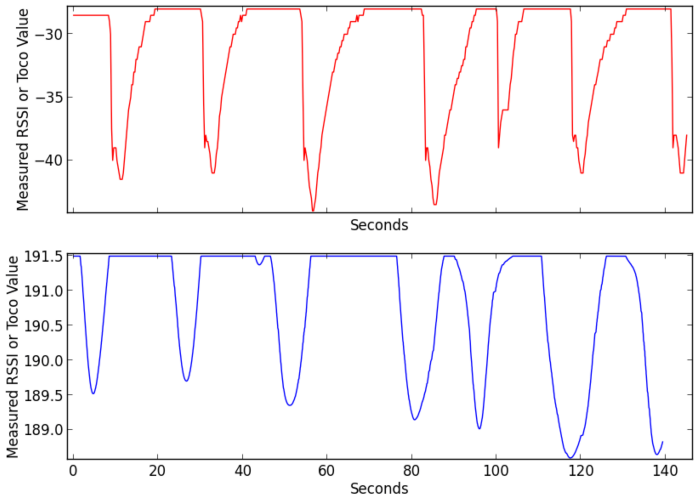

Figure 4.2: Visualization of data collected from a Tocodynamometer (top) and RFID (bottom), with a Gaussian filter and saturation point applied

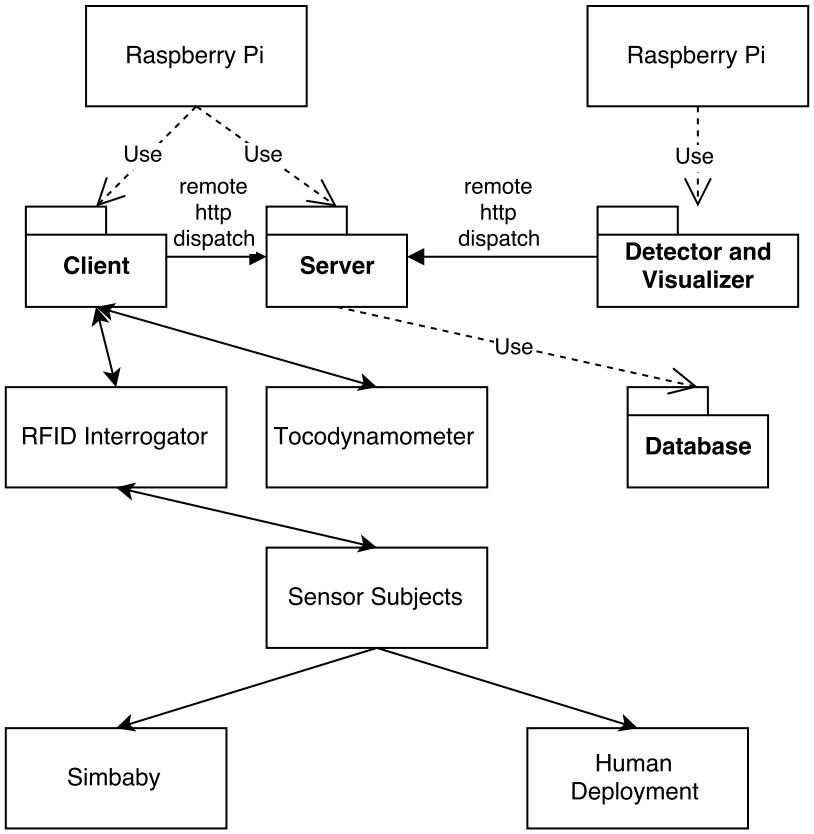

Figure 4.3: Communications diagram of the sensor data collection system that interfaces with the interrogator software described in Figure 4.8; here, a single entity or software class is depicted as a rectangle, a packaged collection of software classes is depicted as a folder, and arrows indicate that an entity invokes ("uses") or communicates with ("http remote dispatch") another entity

(because visualization is CPU intensive), and uses HTTP request messages to the **Server** web service to poll for time-series measurement data periodically.

The database storage layer was separated from the interrogation layer to enable flexibility in sensor interrogation design, so that new sensors and interrogators, including non-RFID comparison devices, could be deployed without the need to re-deploy a database solution for each one. Additionally, this encapsulation enables the design of a generic database interface, allowing for multiple database implementations. This was useful in overcoming compatibility issues between database server software such as MySql and the ARM implementation provided for the Raspberry Pi. The database schema is given in Figure 4.4.

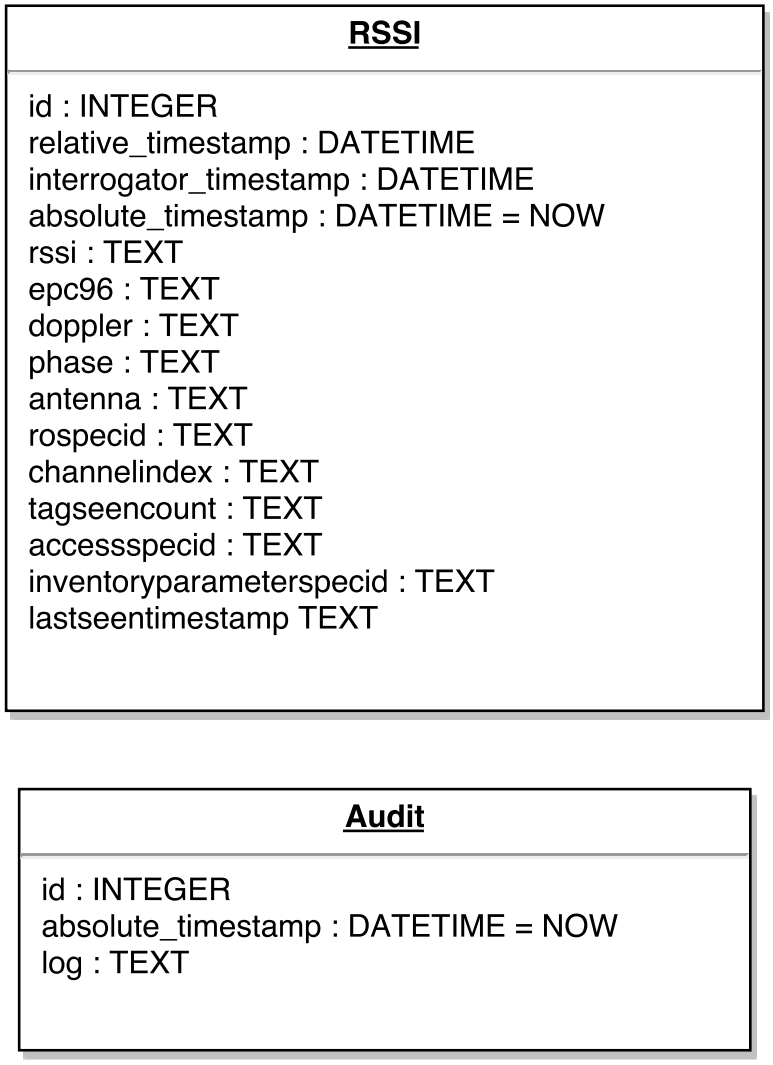

Figure 4.4: Database schema for the encrypted RFID data and HIPAA audit table

```
"data": [
    {
        "db_password": "xxxx"
        "rssi" : -50,
        "relative_time": 9.12345,
        "interrogator_time":
          1/1/2016 23:59:59.12345,
        "channelindex": 10,
        "epc96": "2015abc",
        "antenna": 1,
        "doppler": 0,
        "phase": 0
    },
    ...
]
```

Listing 1: HTTP PUT body to insert a tag interrogation into the database backend

The web server loads an SSL key and certificate for the server to enable encrypted HTTPS communication with the interrogator, and exposes the database table to the interrogator and to the data processing layer via a series of URL endpoints that return all or some of the time-series data contained therein. These service endpoints include:

- **POST** `/api/rssi/<x>/<y>`: Retrieve interrogation data, optionally between timestamps `x` and `y`
- **PUT** `/api/rssi`: Insert one or more interrogation records into the database
- **GET** `/api/audit`: Retrieve the contents of the audit log

To insert one or more tag interrogations into the database, an HTTP PUT body is appended to the `/api/rssi` URL endpoint, representing a JavaScript Object Notation (JSON) array of one or more interrogation objects, as shown in Listing 1. Because the `db_password` provided as part of the PUT request is used to encrypt the data, that password is needed as part of the HTTP body of any retrieval request as well; this is the only required element of the data retrieval web service endpoints. The JSON HTTP body required with all HTTP POST endpoints is shown in Listing 2.

Data are provided to the web service layer as a plaintext HTTP PUT body transmitted encrypted

```
{
    "data": {
        "db_password": "xxxx"
    }
}
```

Listing 2: The HTTP POST body to retrieve data from the database requires only the database password used to encrypt the data when it was inserted using a PUT request

via HTTPS, and the database module encrypts the values using the user password also specified in the HTTP PUT body prior to insertion into the database.

4.3 RFID Physical Infrastructure

The overall architecture of this software prototype is described by the following package diagram, shown in Figure 4.5. It describes the interactions between seven independent but communicating modules, that interface between the underlying protocol and the hardware, producing a standard data format that the processing modules can parse. This prevents differences in protocols or protocol implementations from impacting the modules in the layer above, because the lower-level modules serve as adapters providing common interfaces to them.

The lowest-level of these packages is the **Data Reader**, which is implemented by two modules: the **LLRP Data Reader** and the **Bluetooth Data Reader**. Each module interfaces with its respective device (the Impinj for the LLRP Data Reader and the IP30 Bluetooth for the Bluetooth Data Reader). Although their protocols are different, each reads RFID tag event messages from its respective device and constructs a dictionary object containing (at a minimum) the EPC tag, Antenna, RSSI value, and Timestamp. This dictionary is then converted to an XML document, either by the LLRPyC client **LLRP Protocol** module, or by the **XML To Dict** module. Regardless, the resulting XML document is identical and is depicted by the code listing below and diagrammed in Figure 4.6.

<RO_ACCESS_REPORT>

<Ver>1</Ver>

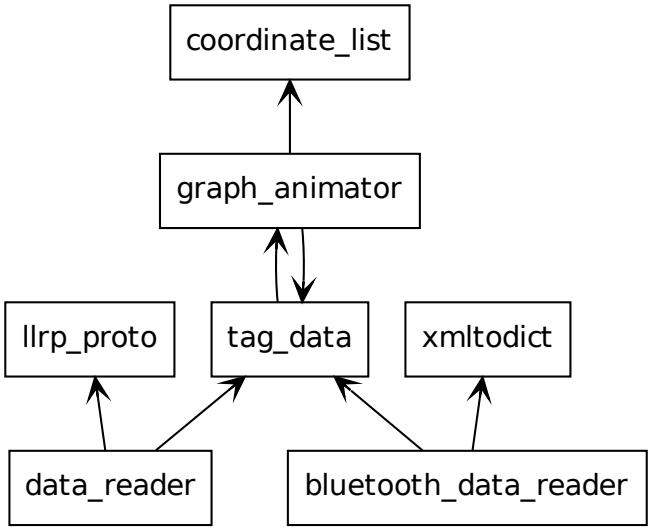

Figure 4.5: RFID UML Package Diagram

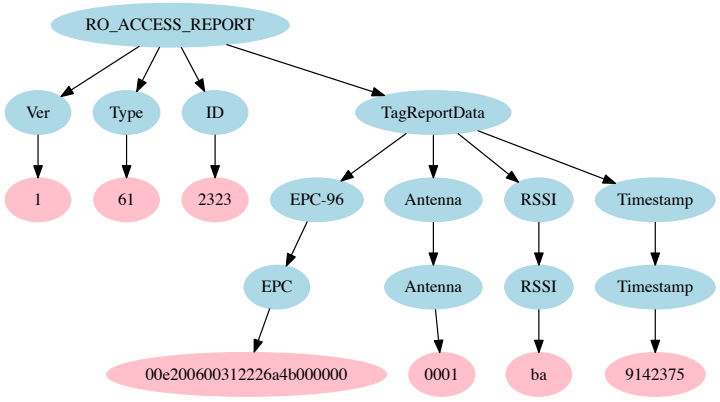

Figure 4.6: Structure of a Sample Read Event Message XML Document

```
<Type>61</Type>
<ID>2323</ID>
<TagReportData>
        <EPC-96>
                <EPC>00e200600312226a4b000000</EPC>
        </EPC-96>
        <RSSI>
                <RSSI>ba</RSSI>
        </RSSI>
        <Timestamp>
                <Timestamp>9142375</Timestamp>
        </Timestamp>
        <Antenna>
                <Antenna>0001</Antenna>
        </Antenna>
</TagReportData>
</RO_ACCESS_REPORT>
```

This XML document is parsed and the underlying parameter values are passed into an array of data structures provided by the **Tag Data** module. This module holds the sequence of RFID read event messages (represented by the XML structure in Figure 4.6), and provides a suite of functionality to the presentation layer, including plotting and interpolating the data, finding read gaps in the data in which the RFID tag may have been stretched for a period of time, or animating the plot in real-time. The animated plot implementation became significant enough that it merited its own sub-module, the **Graph Animator** module, which reads data live via a thread from the **Tag Data** module, stores its X and Y coordinates in a data structure provided by the **Coordinate List** module, and then plots any points that have not yet been plotted, rendering a live animation for the viewer. The **Coordinate List** module provides a list of X and Y coordinates and groups them

by EPC96 tag, so that different RFID chips can be plotted separately by the **Graph Animator**.

After sending the `Impinj Enable Custom Extensions` message, the ROSpec can be configured with various Impinj custom extensions that result in additional measurements emitted from the interrogator for data collection. A Wireshark network trace of the `Add ROSpec` message showing the encodings for the Impinj vendor ID and these parameters (as indicated by the Impinj specifications[108]) is shown in Figure 4.7.

Each time a tag interrogation report is received from the interrogator, the client software emits an HTTP PUT command to the web service including the data from that interrogation. This, in turn, inserts the tag into the remote database. This is done through an asynchronous callback invoked by the LLRP finite state machine each time an `RO_ACCESS_REPORT` message is received containing tag interrogation data[22;36;109].

A summary of interactions among the interrogator modules and its communication with the backend web service layer using a remote HTTP connection is described in Figure 4.8. Additionally, the interactions between the modules in the database and web service layer are described in Figure 4.9.

We configured the RFID interrogator to target 90 Hz interrogation rate of all tags, with small fluctuations in read rate at "channel boundaries" when the interrogator hops from one frequency to another[18]. These fluctuations were removed during data processing via interpolation and resampling, and were suitable for time-series Fourier analysis.

Portability: Bluetooth Extension

The Impinj Speedway is a powerful but fixed RFID interrogator that can read multiple RFID chips at a high rate (approximately 30 per second). This is useful for research and development purposes, and may be appropriate for fixed-installation in a room or near a hospital bed; however, given the passive nature of the RFID chip and proximity antenna, it is desirable to integrate this with a portable interrogator that could be carried during mobility or relocated as-needed. This is useful in the biomedical context for pregnancy uterine monitoring, as this can enable truly mobile monitoring of uterine activity.

```
TLV Parameter: Custom parameter (Impinj - Tag report content selector)
    .... ..11 1111 1111 = Type: Custom parameter (1023)
    Length: 97
    Vendor ID: Impinj (25882)
    Impinj parameter subtype: Tag report content selector (50)
  TLV Parameter: Custom parameter (Impinj - Enable serialized TID)
      .... ..11 1111 1111 = Type: Custom parameter (1023)
      Length: 14
      Vendor ID: Impinj (25882)
      Impinj parameter subtype: Enable serialized TID (51)
      Serialized TID Mode: Enabled (1)
  TLV Parameter: Custom parameter (Impinj - Enable RF phase angle)
      .... ..11 1111 1111 = Type: Custom parameter (1023)
      Length: 14
      Vendor ID: Impinj (25882)
      Impinj parameter subtype: Enable RF phase angle (52)
      RF phase angle mode: Enabled (1)
  TLV Parameter: Custom parameter (Impinj - Enable peak RSSI)
      .... ..11 1111 1111 = Type: Custom parameter (1023)
      Length: 14
      Vendor ID: Impinj (25882)
      Impinj parameter subtype: Enable peak RSSI (53)
      Peak RSSI mode: Enabled (1)
  TLV Parameter: Custom parameter (Impinj - Enable optimized read)
      .... ..11 1111 1111 = Type: Custom parameter (1023)
      Length: 29
      Vendor ID: Impinj (25882)
      Impinj parameter subtype: Enable optimized read (65)
      Optimized read mode: Enabled (1)
    TLV Parameter: C1G2 Read
        .... ..01 0101 0101 = Type: C1G2 Read (341)
        Length: 15
        OpSpec ID: 5
        Access password: 0 (0x00000000)
        10.. .... = MB: 2
        Word pointer: 0 (0x0000)
        Word count: 2
  TLV Parameter: Custom parameter (Impinj - Enable RF doppler frequency)
      .... ..11 1111 1111 = Type: Custom parameter (1023)
      Length: 14
      Vendor ID: Impinj (25882)
      Impinj parameter subtype: Enable RF doppler frequency (67)
      RF doppler frequency mode: Enabled (1)
```

Figure 4.7: Modified `Add ROSpec` message sent to the Impinj R420 in order to enable custom interrogator measurements including the Doppler and phase values

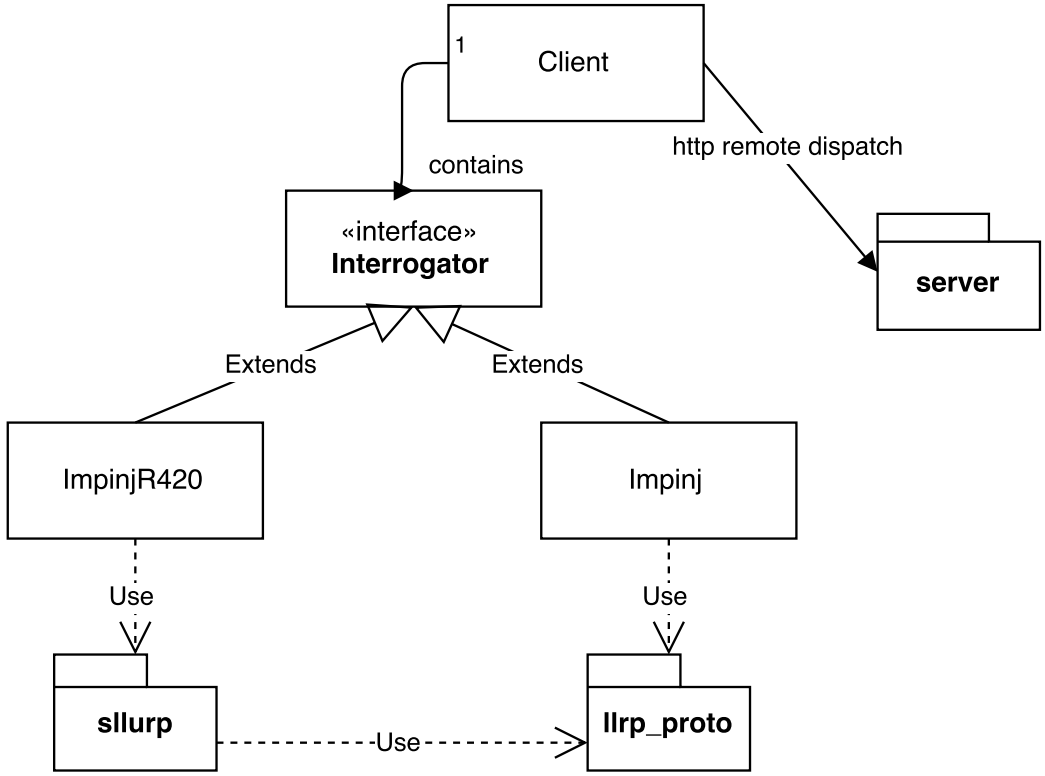

Figure 4.8: High level design of software interface to drive the RFID interrogator, storing time-series data captured into the backend database via HTTP web service calls to the web service depicted in Figure 4.9

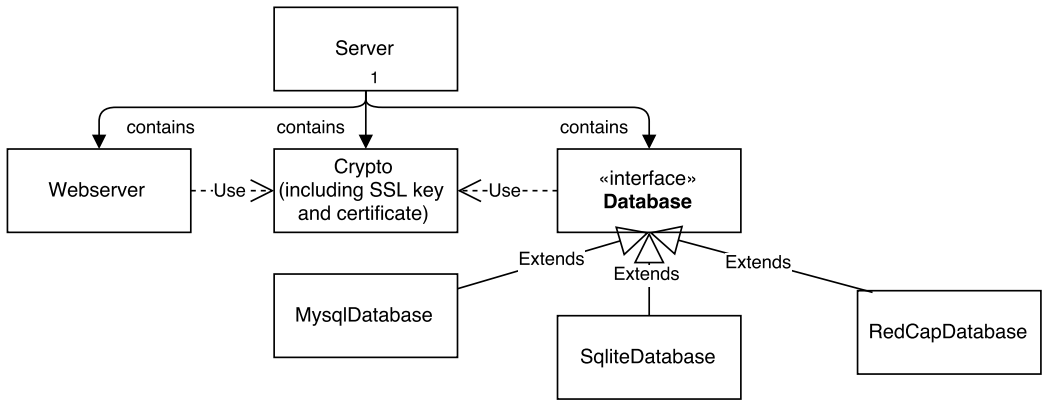

Figure 4.9: High level design of the backend database structures exposed by the web service; here, software class entities are represented as boxes, and arrows between those entities indicate that the entity "contains" another, invokes functionality in ("uses") another, or implements ("extends") a generic interface with functionality specific to a particular implementation (*i.e.*, a particular database installation)

To facilitate this, the Intermec IP30 Bluetooth RFID reader[110] (see Figure 4.10) was used to collect the raw RFID data for the software. Unfortunately, the IP30 did not output its data according to the LLRP protocol, but instead allowed for a configurable output format. This is unfortunate because it is not, by default, compatible with the RFID processing software described in this chapter. A software layer was written to adapt the LLRP protocol server to the analysis framework for compatibility.

Figure 4.10: Bluetooth Portable RFID Interrogator

Because the Impinj device acts as an LLRP server operating over TCP/IP, a TCP/IP server module was written with a similar interface to the LLRP protocol server, but, instead of emitting LLRP protocol messages, creates a Bluetooth connection to the RFID reader device. The LLRP protocol client takes the read event messages from the Impinj RFID reader and converts them into an XML document, so the Bluetooth RFID server module creates and emits this same XML message. This way, the processing modules of the software can operate without regard to which type of device is being used to interrogate the RFID chip.

Originally, the IP30 device was programmed to automatically emit read event messages as they are interrogated; however, these events are clustered and sent several at a time, with only a single timestamp for the entire message. The time-sensitivity of the signal processing required by this study is such that it is necessary to provide a precise timestamp for each read event message received, time-stamped by the interrogator at the time it was received (and not at the time it was sent to

the client). It was determined that individual polling of the IP30 results in the received timestamp being provided; therefore, the Bluetooth RFID server module was modified to continuously poll the IP30 for read event messages. The protocol of the Bluetooth RFID server module is as shown in Figure 4.11.

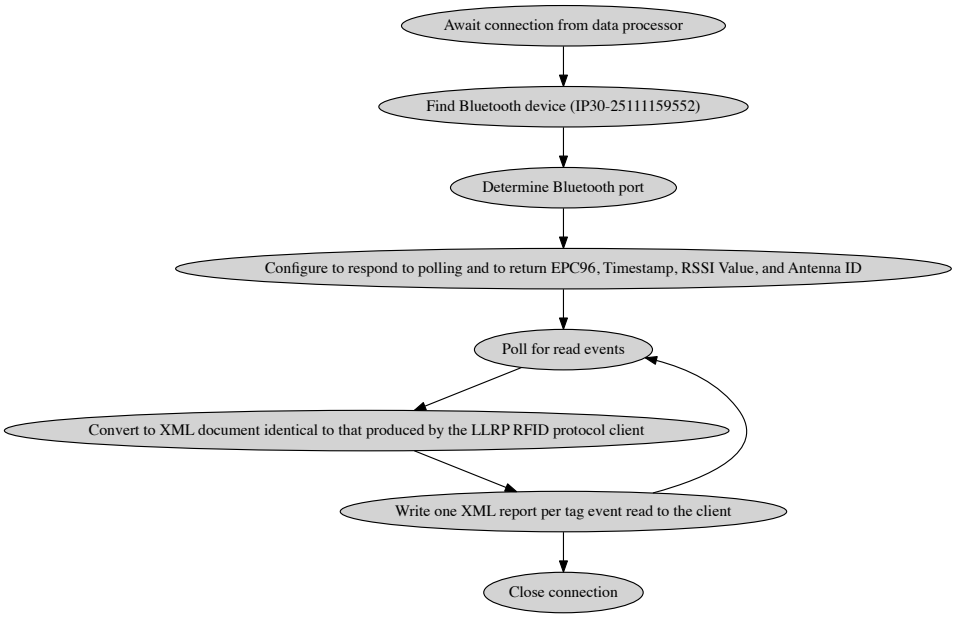

Figure 4.11: Bluetooth Portable RFID Server Module Custom Protocol Workflow

Upon making a Bluetooth connection to the interrogator, the following messages are sent to begin continuous polling for RFID tags in the field:

READ EPCID TIME RSSI ANTENNA REPORT=EVENT

READ POLL

These commands result in repeated messages returned to the interrogating client of the following structure, which is converted from a serial text-based event message (see Figure 4.12) to the familiar XML-based RFID event message as seen in Figure 4.6 for processing.

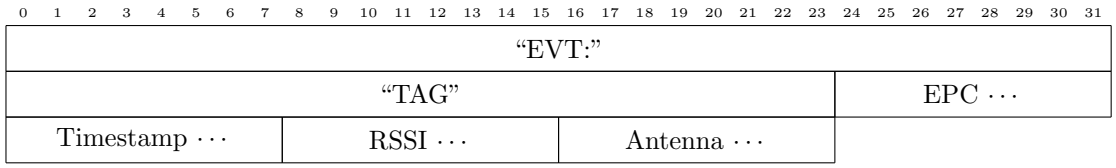

Figure 4.12: Bit-level description of an RFID event message sent by the Bluetooth IP30 portable interrogator

One challenge highlighted by the portability of the Bluetooth RFID reader is that movement of the reader, the RFID chip, the fabric, or the patient will alter the received signal strength sensed by the interrogator and thus processed by the software. One potential solution to this problem is the deployment of ambient sensors on other parts of the body to detect and compare movement unrelated to the medical event being monitored, or deployment of Gradiometer electrodes [111].

Tocodynamometer

One benefit of this framework is its applicability to heterogeneous signal sources, including LLRP-based RFID interrogators, Bluetooth-based RFID interrogators, and other biomedical sensing devices. A critical element of this research is the ability to compare RFID data collected to data collected from a traditional medical device. This is accomplished by substituting our reader module (see Figure 4.5) with one we developed to read a particular tocodynamometer device using its proprietary protocol over a serial interface traditionally used to support external hospital monitors. Instead of the RSSI value, the transducer pressure monitor value is emitted into the framework's data store, and is filtered and processed using the same framework modules we developed in this work. The experimental scripts were re-run with both the RFID and the tocodynamometer software modules running, and with both the bellyband and the tocodynamometer transducer attached to the air bladder (see Figure 4.2). Per the Philips specifications [112], tocodynamometer data are transmitted serially over the RS232 interface for processing. The format of each message is shown in Figure 4.13:

0 1 2 3 4 5 6 7	8 9 10 11 12 13 14 15	16 17 18 19 20 21 22 23	24 25 26 27 28 29 30 31
DLE (0x10)	STX (0x02)	Block Data	Block Data
Block Data ···			
DLE (0x10)	ETX (0x3)	CRC HIGH	CRC LOW

Figure 4.13: Bit-level description of a serial data message from the Philips 50XM tocodynamometer

The CRC bytes are checksums for error detection and/or correction. The 50XM uses the CRC polynomial $x^{16} + x^{12} + x^5 + 1$, this corresponds to a CRC coefficient vector of 0x11021 [112]. The **Block Data** field varies depending on the type of data observed, but we are interested in type 67 for the pressure transducer. Other data types can be ignored, and this packet is structured with a three-byte type field, followed by four three-byte fields corresponding to the recent history (oldest to newest) of heart rate, which is repeated for a second heart rate sensor, and repeated again for the maternal heart rate sensor. These represent the most recent second of data, collected at 0.25 second intervals. These are bitmasked to observe the least significant 12 bits, and divided by 4.0 to obtain the fractional component of each value. The next four bytes represent two seconds of tocodynamometer values, oldest to newest, at 0.5 second resolution. These are divided by 2.0 to obtain the fractional component of each value. A two-byte heart rate mode, one-byte pressure transducer mode, and a one-byte SPO2 reading completes this message structure.

The letter 'G' is sent to the tocodynamometer over the RS232 interface to begin data collection, and the letter 'H' is sent to terminate the connection. The message data, including the CRC checksum bytes, are echoed back to the tocodynamometer to acknowledge the packet and continue the transaction.

4.3.1 Modifications for Real-Time Performance

Further improvements were needed to the sllurp library, which were also implemented to the web service and interrogation layers, to enable real-time monitoring performance. Two problems were observed during RFID interrogation: first, the network-level processing of the data required more time than the period between receiving two interrogation reports. This resulted in a processing lag for each interrogation report that worsened over time and eliminated real-time performance.

Second, the TCP buffer at the network layer of the receiver was filling faster than the tags could be forwarded to the library for processing, exacerbating the delay and resulting in dropped packets resulting in "gaps" in the data of up to 10 seconds.

The problem was caused by the processing that took place for each tag interrogation report prior to passing the report data to the consumer for storage. This processing could include parsing the data, determining whether the report contains relevant interrogation data to the processor, and deciding whether a complete LLRP message had been received. Because LLRP uses asynchronous networking, the data read from the network card may or may not represent a complete LLRP message, and may require additional data still arriving from the network. Unfortunately, it was not possible to configure the Impinj R420 to group interrogations into a single report, such that multiple tags could be parsed with a single network call. Although the Impinj R420 provides this functionality to reduce the number of network calls, it is implemented by discarding the duplicate packets because it is believed that they are redundant if the goal is to perform inventory management of a single tag. To address this problem, a producer-consumer queue was created in the LLRP library and replicated at the web server, interrogator, and database modules[18]. A separate thread executes the parsing code in parallel, extracting data from the queue until a complete message is received, or until the queue becomes empty (this is the "consumer" behavior). The parsing thread suspends if the queue is empty, yielding its processing time to the producer thread. The producer-consumer thread model is illustrated in Figure 4.14. If the interrogation computer has multiple CPU cores, these two threads can run in parallel, enabling real-time performance; however, because most of the producer thread's behavior is networking I/O, it regularly blocks until asynchronous network data arrives, yielding its CPU time to the consumer thread to "catch up" by processing any lingering data on the queue. Previously, the network I/O and incoming data processing was a synchronous process (the "former path" depicted in Figure 4.14), but has been reorganized to two processes that asynchronously share a queue. Thus, with this approach, we have observed no lag or data gaps at a 90 Hz interrogation rate by moving the data parsing behavior of the sllurp library from a series to a parallel process.

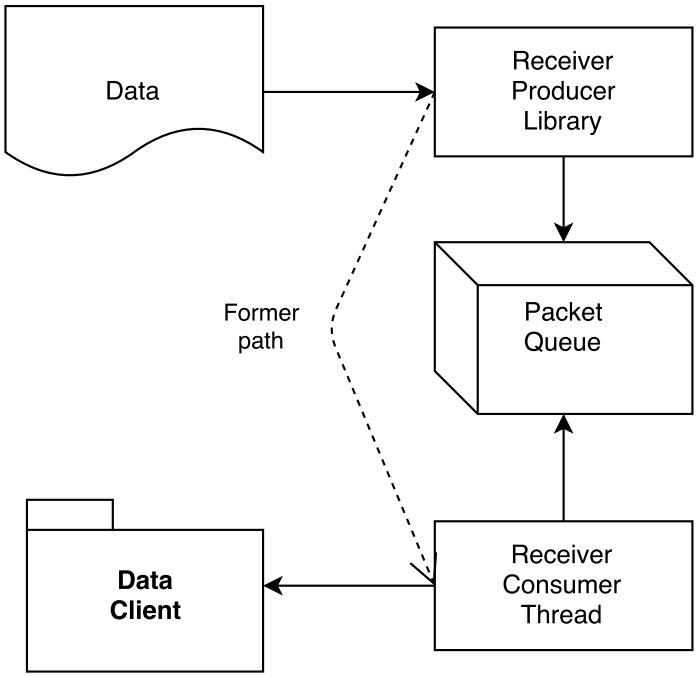

Figure 4.14: The classic Producer-Consumer threading model enables a producer thread to append data to a queue, where a consumer thread retrieves that data and performs deferred processing while the producer thread simultaneously obtains or generates more data to be processed

Additionally, the sllurp library was modified to disable Nagle's TCP algorithm. Nagle's algorithm delays transmission of network data until a sufficient amount has been received to offset the overhead in addressing and transmitting the packet[113]. At first consideration, this would appear to help alleviate the issue of processing delays due to TCP packet overload; however, the producer-consumer model shown in Figure 4.14 mitigates this at the application layer. As a result, enabling Nagle's algorithm only induces a delay in receiving real-time interrogation data, which is preferable to eliminate. To do this, Nagle's algorithm was disabled in the sllurp library, and instead the producer-consumer thread model was implemented at the interrogator, web service, and database layers with a time-delay. In this way, packets are only transmitted over the network at a pre-defined interval that is determined by the application; when packets are transmitted, all queued interrogation messages are sent in a single message packet, reducing the processing overhead of the network transmission. Because most of our applications consider data from the past 1 to 6 second time horizon ("window"),

it is only necessary to deliver processed interrogation data to the signal processing layer every 1 to 6 seconds. We have chosen a more conservative 750 milliseconds transmission delay, which has maintained our desired 90 Hz interrogation rate.

Modifications to the llrp and sllrp libraries[22;36;109] to enable Doppler and phase reporting as well as the performance controls and enhancements are given in *diff* format in Appendix B.

4.3.2 Security Considerations

HIPAA law provides data security and privacy guidelines and regulations for systems that store or transmit protected health information (PHI). It covers data at-rest and data in-transit, which includes data stored on a database system and data transmitted over a network. As discussed in this Section, our system stores and transmits RFID physical data features. Although we do not store personally identifiable information that could be considered PHI, we choose to secure our system in accordance with HIPAA regulations[18] anyway, in case a database could be linked back to an individual on the basis of other information such as the computer used to collect the data, the date, time, and location in which collection took place.

For data in-transit, we use HTTP over SSL (HTTPS) to encrypt data between the client and the database for storage. At the database, data are encrypted using the Advanced Encryption Standard (AES)[114], a symmetric key strategy, and store the data in a database or clinical-trial data storage system such as REDCap[115]. One challenge is that the data tend to remain constant except when a motion event takes place. Typically, symmetric-key ciphers encrypt the same plaintext to the same ciphertext, leading to a vulnerability in time-series data in which patterns may emerge. Here, the physical RFID property values are encrypted to the same ciphertext, which presents a "known ciphertext vulnerability" in which the original data (or, at least, their underlying meaning) can be extracted from the ciphertext. The REDCap interface does not permit automated data retrieval in order to maintain clinical privacy, although we can integrate posting data from the interrogator to REDCap automatically via its RESTful interface[116]. To mitigate this issue, the data can be manually exported from REDCap onto an encrypted hard drive, and a web server database module is provided that is capable of converting and reading that format as if it had been captured natively

by an interrogator in real-time. Even when using REDCap, the data we store is encrypted using the AES key, anyway, to maintain compatibility with the web service layer which expects encrypted data to be exported from REDCap.

To overcome this, we encrypt database records using the Time-Based One-Time Password Algorithm (TOTP) (RFC 6238)[117] by utilizing an automatically generated timestamp, without requiring human entry of a time-based salt, which is random data added to encrypted plaintext and subsequently removed on decryption, used to perturb the corresponding ciphertext for known inputs. This will enable encryption of individual rows in a database and will ensure unique ciphertexts for the same database value recorded multiple times. We generate a new key for encryption of each individual record of data. This key is generated from a symmetric-key password from the user, plus the timestamp from the record, and is re-generated on-demand for each row upon symmetric decryption. The time-salt typically entered manually by the user when using TOTP for Two-Factor Authentication can be automatically provided by the programmatic timestamp. With this scheme, multiple recordings of the same data value will not have the same ciphertext. The scheme uses a TOTP generator as part of its key and can automate the encryption and storage of data in streaming applications. Thus, the TOTP validation window (that allows the user to validate time-salts that occurred prior to or after the true current time-salt, in case of time synchronization of speed-of-entry issues) can be reduced to 1, since the programmatic timestamp will always be synchronized.

Chapter 5: Feature Extraction

In order to classify subject state, it is necessary to compute features of the physical properties of the RFID backscatter signal. The Impinj Speedway R420 provides three physical measures for each interrogation: the RSSI, Doppler, and Phase angle. We correct these measures for perturbations in the interrogation frequency (which is also provided by the R420 for each interrogation) using Equations 4.1 and 4.2.

We now classify subject state from these measures: for example, whether the band is being stretched or is stationary, or whether the RFID tag is being moved through the field. If the features are fully separable, then this problem reduces to that of identifying a perceptron, a function that defines the boundary that separates the data. Data are often not cleanly separable, and noise in the data can shift data points across the dividing boundary in error. When data are not linearly separable, the perceptron polynomial order can be increased, rendering the data "linearly separable" in higher order space. However, this exacerbates the noise inherent in the data, as the perceptron boundary becomes artificially non-linear and the boundary margin between the data classes becomes smaller. The result is overfitting[118], which can be overcome with additional sample "training" data. However, this training data comes at the cost of prolonged training periods, where real-time data must be collected for a long period of time in order to satisfy this constraint.

A separating kernel function such as a perceptron works by separating data vectors x_i into classes y_i ($y \in 0, 1$) for each time epoch i. A linear perceptron provides a linear function for performing this separation by computing weights w that discriminate training data x about a threshold. This threshold is often defined as 0, such that the sign of the resulting classification y can be used as the class, as shown in Equation 5.1. As each training vector x_i is provided to the perceptron, the weights w are updated such that an abstract "dividing line" is maintained that separates all training vectors within each class. It is often the case in practice that the data is not perfectly separable, even during training. When this occurs, the weights could instead be computed to maximize the

margin between the maximum number of data elements between the two classes, as is done with an SVM[71].

$$y = sign(w_0 + \sum_{i=1}^{k} x_i w_i) \tag{5.1}$$

Typically, our aim is to classify perturbations in the physical properties of the RF backscatter signal that result from deformations of the knitted fabric antenna. However, these backscatter signal properties are dependent upon the impedance match between the antenna and the tag, which is altered as the knitted antenna shape is deformed due to strain, as well as the distance between the tag and the interrogator, which is altered as the garment itself is moved about in space. Noise artifacts are introduced into the observed signal because stretching and coarse movements may occur simultaneously, because training classification of continuous movements such as respiratory activity is difficult since brief pauses occur naturally in-between expiration and inspiration which would present as stationary data but would not be classified as non-breathing, and because RF noise artifacts may reduce or eliminate complete separability between the classes. Additionally, perceptron-based classification requires training samples that will reflect future conditions; changing environmental conditions and unrelated human movements may alter these conditions, limiting the separating power of the established perceptron, and necessitating monitoring or retraining of these conditions over time. Absent these challenges, a significant number of training samples is needed to ensure that the weights w are adjusted to adequately separate the data; perfect separation is only assured if an infinite number of training samples are provided; Hoeffding's Inequality (see Equation 5.2[118]) yields a misclassification probability upper bound for a desired accuracy and number of training samples.

$$Pr(\|Y - E[Y]\| > \epsilon) \leq \delta = 2e^{-2n\epsilon^2} \tag{5.2}$$

Hoeffding's Inequality gives the probability of misclassifying the expected value of the prediction of new data vectors $E[Y]$ from the given training data set Y beyond an error bound ϵ. This upper-bounded probability is given as δ, which is exponentially related to the number of training samples

n and the error bound ϵ. Thus, the probability of misclassification due to a failure to generalize a model from the training set is reduced as the number of training samples n is increased, or as the error bound ϵ is relaxed[119]. Vapnik and Chervonekis further constrain Hoeffding's Inequality by considering only those models that uniquely alter the classification predictions in order to avoid double-counting misclassifications for purposes of computing this probability bound. The resulting VC-Dimension is the maximum number of data vectors that can be "shattered," or separated by the classifier[120], and is related to the dimension of the training data vectors. The VC-Dimension is given in Equation 5.3[118], and iteratively relates the required number of training data vectors n to the probability bound δ via the VC-Dimension d_{VC}. d_{VC} is defined as the number one greater than the dimensionality of the training input vectors for linear perceptrons such as the SVM. The relationship between δ and d_{VC} explains the "curse of dimensionality" that results from training a classifier on too many data features, resulting in overfitting and misclassifications due to loss of generality of the separating line between the data classes.

$$n_{k+1} \geq 8\frac{1}{\epsilon^2} ln \frac{4(2n_k^{d_{VC}} + 1)}{\delta} \tag{5.3}$$

As a typical example, if the mean of the RSSI and standard deviation of the RSSI are used as classification features, they would be organized into vectors in $\mathbb{R}^2$, and $d_{VC} = 3$. Equation 5.3 converges after approximately 4 iterations to $N \geq 5 \times 10^8$, if an error bound of $\epsilon = 0.1\%$ and $1-\delta = 95\%$ confidence, as shown in Equation 5.4. This is an infeasible 192 days of classifier training. $N \geq 3 \times 10^4$ training samples are required for $\epsilon = 10\%$ and $1 - \delta = 90\%$, or 17 minutes of training data for an interrogator polling at 30 Hz. It has been similarly observed that $N \geq d_{VC} \times 10^4$ training samples are typically required for $\epsilon = 10\%$ and $1-\delta = 90\%$[118], but in practice, $N \geq d_{VC} \times 10$ has been found to meet the $\epsilon = 10\%$ and $1-\delta = 90\%$ tolerance bounds[121], or 1 second of training data samples at 30 Hz polling. For a tolerance of $\epsilon = 0.1\%$ and $1 - \delta = 95\%$, approximately 5 hours of training data samples at 30 Hz polling ($N \geq 534 \times 10^3$) are required. Because the number of training samples required varies inversely with the square of the tolerance ϵ, a small relaxation of tolerance results in rapidly diminishing training requirements; for example, approximately $N \geq 4 \times 10^3$ training samples

are required for a tolerance of $\epsilon = 1\%$, which yields a more manageable 2.3 minutes of training at 30 Hz polling.

$$N_{k+1} \geq 8\frac{1}{0.001^2}ln\frac{4(2N_k^3+1)}{0.05} \tag{5.4}$$

In addition to the time constraint to collect training data, it is often the case that example training samples are only available for some, but not all, classes under consideration. This is a common occurrence in anomaly detection, in which nearly all data observed falls into the "normal" class. While methods exist to compensate for the imbalance between training classes, such as the SMOTE method[122] or a semi-supervised approach[123], the imbalance of training data is exacerbated in our biomedical applications, because the "anomaly" class (such as apnea) is medically undesirable, or is an event that must be classified even during training (such as a uterine contraction).

Classification approaches that overcome these limitations are discussed in Chapter 6, but we describe in this Chapter the data features extracted from the physical RFID measures that inform the approach.

5.1 First-Order Features

First-order features of the data include statistical features such as the mean or standard deviation. To extract these features, a window of data of a specified size is observed, and a new time series of data is emitted consisting of the statistical measures of each window. As the window size grows, the features converge to the population statistical features, which is desirable for removing variance due to noise artifacts; however, the separability between the classes becomes confused as the potential exists to include individual data points that cross classification boundaries. The window size, then, should be large enough to incorporate one complete artifact being observed (*i.e.*, a respiration); it is not possible to know this window size *a priori* for all subjects, and an estimate is used based upon the read rate of the interrogator (*i.e.*, 0.5 seconds of data).

Once the features are collected, their suitability for classification can be evaluated based upon their mixture between classes. In other words, the separability of two features is inversely related to

the overlap between their histograms. We can evaluate this more formally using the Fisher Linear Discriminant Ratio Analysis[106] (FDR), defined in Equation 5.5. Here, the mean and standard deviation of the feature time series data computed in each class x is computed as μ_x and σ_x are compared: a greater difference in the means between the two classes indicates good separability of the feature, and a smaller difference in the standard deviation also indicates good separability, since the feature data are less likely to overlap.

$$F = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2} \tag{5.5}$$

For our application, one challenge is identifying data into one class or another, since the anomaly class is often undesirable. Thus, we cannot label the data into classes using human testing. However, we expect that the results of this study will inform generalizable algorithm development, and therefore FDR analysis will not be required on every trial. Therefore, a Laerdal SimBaby[38] programmable mannequin (see Figure 13.1) was used to collect stretching and non-stretching data for comparison. Statistical features were computed on windows of stretching data and on identical-sized windows of non-stretching data, and the FDR of each feature was computed according to Equation 5.5. We performed this analysis with and without data filtering applied (see Section 5.3). This resulted in the FDR values shown in Table 5.1.

Table 5.1: A listing of Fisher Linear Discriminant Ratio scores for several features of RSSI data, indicating separability of the data between the actuating and non-actuating class, for unfiltered physical RFID data and Kalman-filtered physical RFID data.

	Minimum	**Maximum**	**Median**	**Mean**
Unfiltered	1.25	0.34	2.14	2.23
Filtered	1.56	0.93	2.19	2.16
	Skewness	**Kurtosis**	**Standard Deviation**	**FFT Average Frequency**
Unfiltered	0.46	0.73	1.20	0.04
Filtered	0.32	0.13	0.91	0.04

From Table 5.1, we observe low separability among these first-order features, indicating mixture

between the data. This is unsurprising, given that some time is spent during each band stretch in a stationary position: the band is at rest in-between "breaths," and is also not moving during the period in which the band is fully stretched prior to retraction. This result suggests that a temporal relationship such as a Markov Chain (discussed in Sections 6.3 and 6.6) will better inform state classification. However, it is common to observe RSSI dynamic range from fully stretched to at-rest on a human body of 1 to 2 dBm of RSSI, as opposed to a more typical 5 to 10 dBm of RSSI at distances of 2 to 5 feet; the reduction of RSSI dynamic range is due to the absorption of RF energy by the human body that would have been reflected by the antenna in open air. Noise artifacts and multipathing can account for nearly all of the dynamic range in this situation, and deeper feature relationships are needed to reliably classify the state of the band and wearer.

5.2 Unbalanced Classification

Although separability between all of these features is minimal with some overlap between the two classes, due in part to the inclusion of stationary periods in the non-stationary class, we perform classification on these first-order features to provide a baseline upon which to compare more sophisticated approaches. Since evaluating the FDR of these features is an offline process, it was possible to collect the data from a mannequin and label it accurately. However, real-time classification of that data would require labeling at least some of the data online from the human subject. This is because environmental factors, depth of breathing, and distance between the tag and interrogator all affect the dynamic range of the features and, thus, the perceptron that best separates them. It is biologically infeasible to collect labeled data for both classes. Methods exist to perform classification of anomaly data by oversampling the anomaly class[122], but even this assumes that at least some anomaly example data can be observed and labeled. Since the mean and standard deviation were the most separable unique features, in the sense that the minimum and median are related to the more separable mean and thus discounted, we adapt the oversampling method by generating an entire class of training data using the samples collected in the single observable training class. To do this, we reflect those data points above the mean of means about a horizontal line passing through the smallest mean, and about a vertical line passing through the mean of the standard

deviations. Observed data is only reflected above the mean of means to reduce or eliminate those data points collected during non-stretching transitional periods between "breaths." This still results in an imbalance between the classes, since only a subset of training data are reflected into the anomaly class. To balance the classes, a normal distribution with a mean equal to the 5^{th} percentile of all observed means, and a standard deviation of all observed means. This distribution was chosen to capture most training data points at the lower end of the observed training period, which are likely to be those transitional points between actuating and non-actuating, while allowing for some overlap of the data into the margin between the classes due to the observed confusability between the classes during FDR analysis. The resulting feature set collected from a SimBaby mannequin, and the synthetic anomalies injected for classification, are depicted in Figure 5.1.

The SVM[71;124] is an algorithm to compute a perceptron between classes of training data, such as that observed here. There exist Two-Class and One-Class versions of the algorithm, and the One-Class SVM[125] is designed to handle anomaly detection in which only one class of training data can be observed. This approach is well-suited to our problem domain; however, the One-Class SVM draws a boundary around all sides of the training data, leading to misclassification of data near or just beyond the boundary as anomaly data, as well as misclassification of breaths that are unusually deep. The latter concern is easily mitigated by inspecting the anomaly data after it is identified; however, the Two-Class SVM with injected simulated anomalies helps to mitigate both issues. Additionally, the temporal relationship between classifications in time can be exploited with a Two-Class SVM, because we can essentially measure the relative distance from each classified point to the perceptron margin. We exploit this temporal trend as a classification feature in Section 6.3.

5.3 Filtering Strategy

Digital filtering of extracted features can be performed online, in real-time, prior to classification. Filtering generally removes brief spikes in the observed data, which is more likely to be a noise artifact than a biological artifact which will present a wider peak. Because filtering typically involves fitting a curve or a moving average across the data, some signal will be lost from the data in the form of dynamic range; as a result, the standard deviation is reduced across the first-order features. This

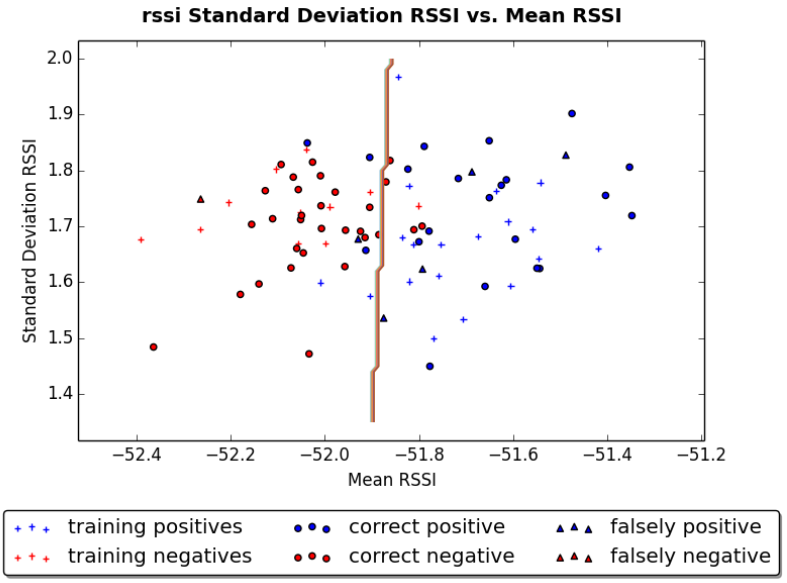

Figure 5.1: Bellyband data observed on a programmable mannequin, classified into actuating (blue) and non-actuating (red) classes: non-actuating training examples are generated using a subset of the actuating training examples, and the resulting Support Vector Machine classification performance on subsequent testing data is shown as circles (correct) and triangles (incorrect)

is evident in Table 5.1, in which the filtered separability is slightly smaller than the corresponding unfiltered feature separability. Nevertheless, filtering and smoothing will facilitate higher-order feature extraction such as spectral properties of the oscillatory signal, because the filters also serve as a low-pass filter that removes the higher frequency brief spikes presented by noise artifacts. An example low-pass filter on ζ (defined in Chapter 4) is shown in Figure 5.2.

We use three types of digital filters for our approach to facilitate a specific goal of the approach. For temporal smoothing in which data is likely to track a particular target that may change over time, particularly when we can model the expected changes to the target due to physical inputs, we use a Kalman filter[126] (see Figure 5.3 for an example effect of Kalman filtering on RSSI data). To filter short-term windows of data to identify temporal trends, such as long-term RSSI deviations due to ambient motion artifacts such as walking down a hallway during monitoring, we use Locally Weighted Scatterplot Smoothing (LOWESS)[127]. Finally, when filtering to smooth out brief and/or rapid spikes in the data as a low-pass filter for denoising, we use a polynomial fitting multi-pass

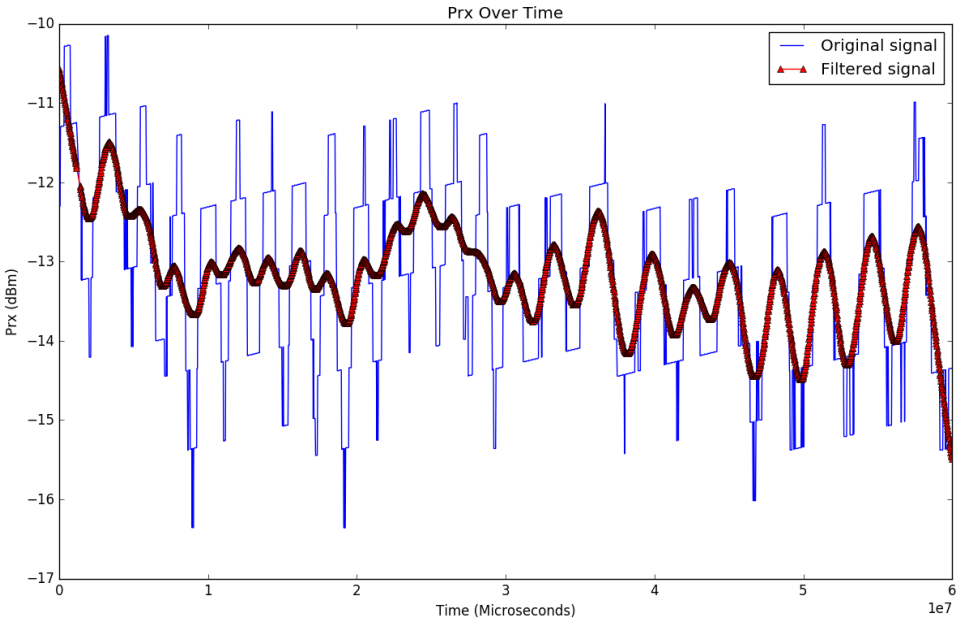

Figure 5.2: Received power ζ feature filtered using a low-pass filter

Savitzky-Golay filter[128] or, for longer-term white noise reduction, a multi-pass moving average filter (which approaches a Gaussian filter as the number of filter passes grows).

One challenge when using filters is that they are often parameterized: the Savitzky-Golay filter requires a moving average window size and a polynomial order for curve fitting; however, we attempt to adapt these parameters to capture the estimated duration of the observed artifact and the estimated noise in the data. The parameters are chosen based upon the estimated rate of the observed phenomenon and the sampling rate, to obtain a moving window that is approximately one-half period in width. Like moving average filters, the Savitzky-Golay filter smooths data in the time domain, but has limited attenuation of higher frequencies regardless of polynomial order[129]. The multi-pass order of the filter is chosen to magnify the dominant frequencies in the data window; because respiratory behavior is oscillatory but not necessarily periodic, multiple dominant frequencies may be observed in the same time window. Because the convolution of the Savitzky-Golay filter window with the input data corresponds to multiplication in the frequency domain, repeated application of the filter has a multiplicative effect on the frequency response on the same order as

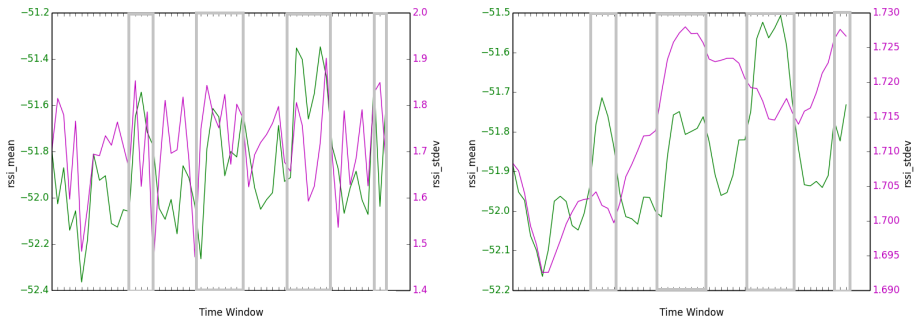

Figure 5.3: Bellyband RSSI data over time before application of a Kalman filter (left), and after (right), with mean and standard deviation plotted over 8-second time horizons. The RSSI mean can be more easily observed after filtering from the actuating (highlighted in boxes) and non-actuating time windows.

the number of passes selected (see Equation 5.6 for filter coefficients a and data window x). The multiplicative effect attenuates non-dominant frequencies relative to the dominant one, and sacrifices signal dynamic range for enhanced low-pass filtering. The benefit of attenuation diminishes as the number of passes increases, as the dynamic range of the signal is reduced; it is infeasible to eliminate artifact components entirely without sacrificing useful signal. Therefore, a limited number of filter passes is taken, even though this will leave behind a high-frequency, low magnitude modulation within the data. As shown in Equation 5.6, by the distributed property of convolutions, the multi-pass filter can be prepared *a priori* by convolving the coefficients a (or multiplying the corresponding frequency spectra) with themselves, without regard to the data window x, until the desired order is obtained.

$$\begin{aligned} a * x &= \mathcal{F}^{-1}(\mathcal{F}(a)\mathcal{F}(x)) \\ a * (a * x) &= \mathcal{F}^{-1}(\mathcal{F}(a)\mathcal{F}^{-1}(\mathcal{F}(a)\mathcal{F}(x))) \\ a * (a * x) &= (a * a) * x = \mathcal{F}^{-1}(\mathcal{F}(a)^2\mathcal{F}(x)) \end{aligned} \tag{5.6}$$

5.4 Higher-Order Features

When the band is not actively stretching, it is either in a stationary stretched position, or in a stationary position at-rest. During these periods, the power distribution across the frequency spectrum should approach white noise, for large enough windows that non-Gaussiean multipath artifacts are summed with perturbations due to environmental changes. One potential feature to extract from the data is the maximum oscillatory magnitude observed in the Fourier Transform of a small window (*i.e.*, 0.5 seconds) of RSSI data.

It is difficult to test the separability of this feature since these periods occur frequently in both classes, since the band frequently remains briefly stationary even during actuating periods. However, by observing the duration of each respiratory activity on the SimBaby programmable mannequin, we remove what we expect to be this intermediate data, and observe an FDR of 3.85. This FDR is 0.49 when the intermediate stationary periods are considered, so it remains difficult to obtain separable training data for this feature for use with a perceptron. Nevertheless, Figure 5.4 depicts visually separable cases of actuation and non-actuation, which we will exploit for classification in Section 6.3.

Finally, we observe that two distinct distributions should be present in windows of Bellyband data if stretching activity is taking place, regardless of the magnitude of stretch of the knit antenna itself. One distribution represents the noise within, and the other represents the oscillation of data values during stretch activity. Because we do not know the properties of the two distributions we would search for in the data, we construct a Markov Chain Monte Carlo [130;131] (MCMC) simulation to test the hypothesis that two distributions exist within the data. The MCMC simulation works by randomly selecting possible distributions in the search space and evaluating whether each subsequent set of distributions is a better fit than the previous set. If an improvement is found, the MCMC moves to that set of distributions and selects another nearby set of distribution parameters in the search space to continue evaluation. If an improvement is not found, the search randomly selects a new set of distributions to evaluate. As with any search algorithm, it is necessary to avoid converging upon a locally optimal solution that is not the globally optimal solution, so an MCMC simulation

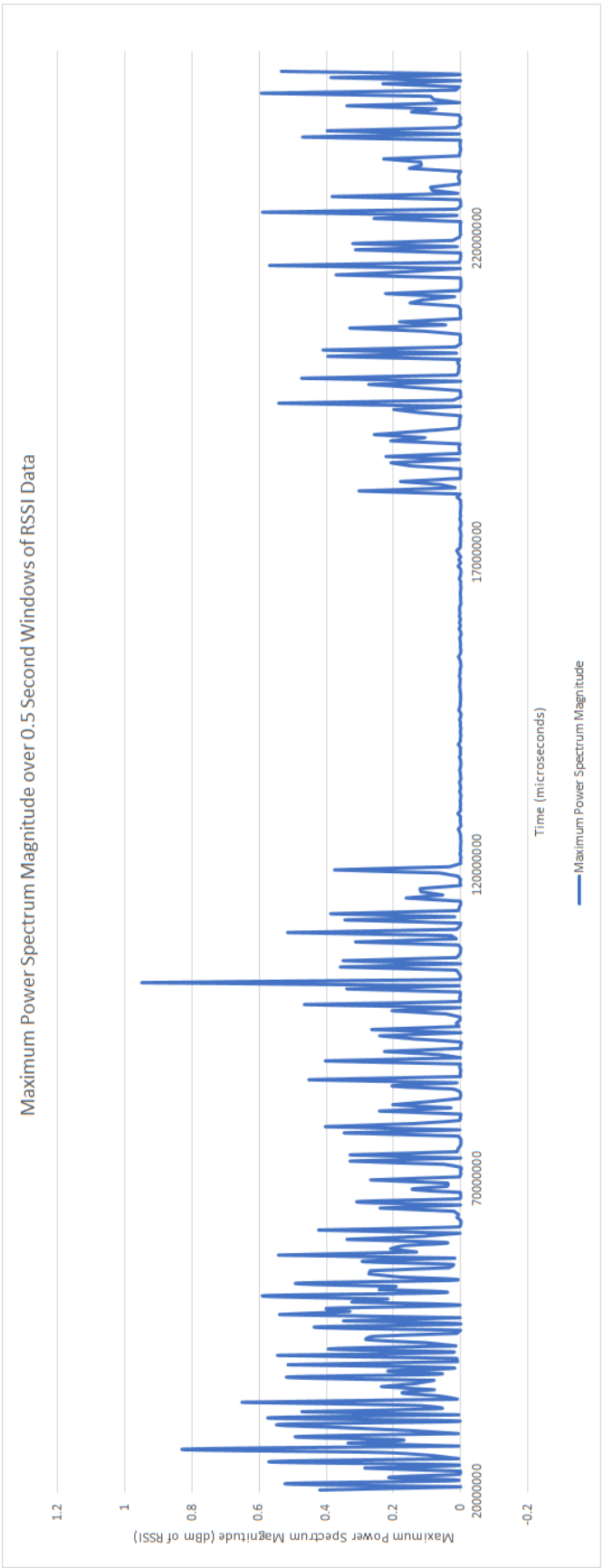

Figure 5.4: Bellyband power spectrum maximum magnitude from a Fourier Transform of 0.5 second windows of data; SimBaby respiration can be observed at a high rate (30 respirations per minute) from 20-60 seconds, a slower respiration rate (15 respirations per minute) from 60-120 and 180-240 seconds, and non-breathing from 120-180 seconds

randomly warps in the search space even if an improvement is found with a small probability. Given infinitely many iterations, the MCMC simulation should visit locally optimal solutions in the search space more often than those with random warps; if no optimal solution exists, the simulation should visit all possibilities approximately equally often, and at a rate greater than the random warp term used when a converging solution is found. The simulation converges with sufficient iterations; we chose 10,000 iterations which is sufficient given the small size of the data window yielding a more limited number of discrete distributions that would fit the data. In the case when no optimal solution exists, we assume that only a single distribution exists in the data accounted for by noise, and the band was not stretched during the time period under consideration. We will use the results of the MCMC simulation for both classification (Sections 6.3 and 6.4) and for semi-unsupervised re-training (Section 6.6).

Chapter 6: Activity State Classification

As discussed in Chapter 5, the Support Vector Machine was preferred for inferring subject state using RFID interrogation. This decision was informed by the FDR metric, but other approaches and statistical features exist. For example, k-Means Clustering[132] groups collections of data, like the SVM, but then computes the nearest centroid to a new datum point while the support vector machine draws a line or hyperplane to separate the groups. In addition, features of the data distribution, such as skewness and kurtosis, can inform the shape of data points for comparison to training data using methods like SVM and k-Means Clustering. Feature-based classification of RSSI data is difficult because of the confusability between the two classes, as can be seen in the histogram in Figure 6.1, and as confirmed by the Fisher separability ratios computed in Table 5.1. The goal of this Chapter is to perform effective classification of subject state (*i.e.*, breathing or non-breathing) using noisy RF backscatter features observed from the knitted antenna being worn by the user.

Analysis of the power spectrum utilized a t-test for hypothesis testing against the null hypothesis that respiration activity is taking place. A t-test is described in Equation 6.1[133], where s is the sample (the maximum power spectrum in the past $T_s = 0.5$ second window), μ and σ^2 are the mean and variance, respectively, of all samples collected during the training period, and n is the number of training samples collected. The t-test is inspired by the discriminant analysis carried out in Section 4.1, and enables analysis of the feature without a separating perceptron, which is made difficult by the mixture of both classes even during stretching activity, while requiring a minimal amount of training data to be effective (40 samples, or 20 seconds of data using 0.5 second windows).

$$t = \frac{s - \mu}{\sqrt{\frac{\sigma^2}{n}}} \tag{6.1}$$

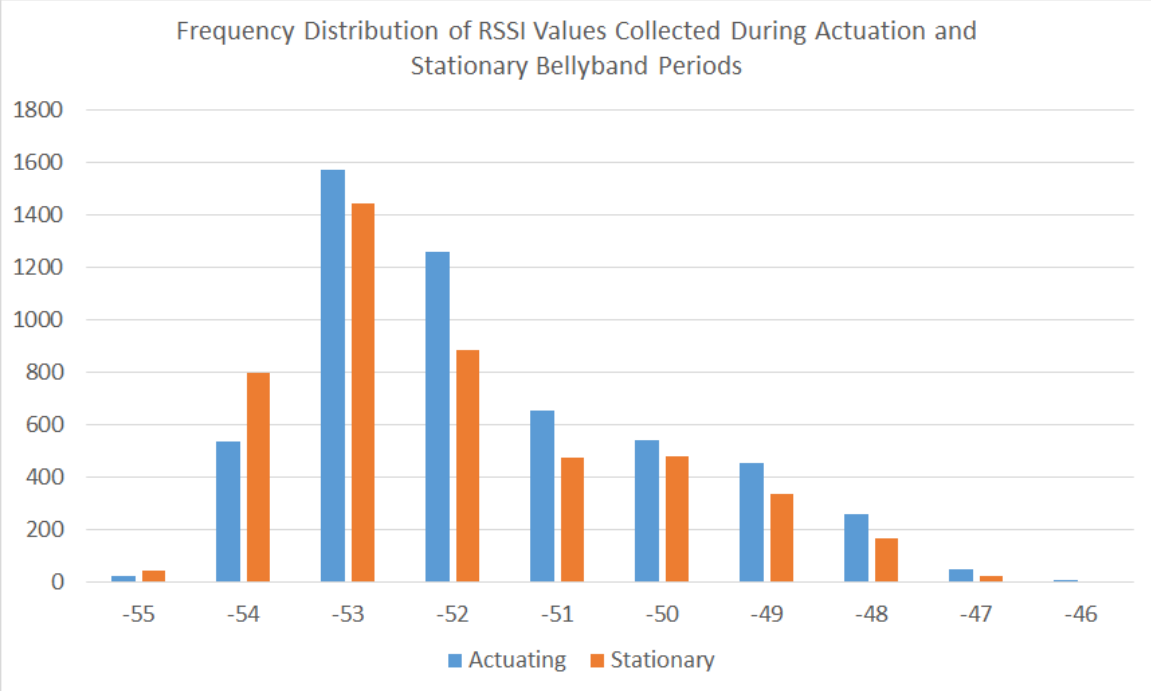

Figure 6.1: Histogram of RSSI values collected and separated into actuating "breathing" (left) and stationary "non-breathing" (right) classes, showing the high degree of overlap between the classes

6.1 Software Classification Framework

To conduct RFID sensor interrogation experiments, it is necessary to implement signal processing and machine learning software to query the web service in real-time to obtain RFID interrogation data as it is collected from the interrogator. Because of the multi-threaded asynchronous service-based approach to implementing the server, it is possible to simultaneously make requests of the web service using HTTP while the server is also handing requests to insert data from the RFID interrogator(s).

In order to process the data collected and visualized into decisions that can be acted upon or evaluated further by medical personnel, we implement an additional software module that emits a binary signal when it determines that an event such as a contraction or apnea situation occurs. To do this, filtering modules were created to reduce noise and quantization issues within the data; these modules are used in post-processing to facilitate data collection in the research and development

phase, but are compatible with real-time analysis. Here, our approach is to create a scenario and program a pumping device such as the Laerdal SimBaby or a software-controlled peristaltic pump to inflate an air bladder at known rates and durations. We then use our collection software to collect the corresponding RFID data for analysis. Our goal is to determine when a certain amount of "stretching" has occurred on the band; for this research, we quantify this by determining the frequency with which the band is stretched by the pump against the known quantities of our testing script. We do this by computing the Fast Fourier Transform (FFT) of the collected data, and seeking the frequency with the highest magnitude to determine the observed rate. This method has been used to accurately determine the inflation rate of the air bladder, and has been further improved by applying a Gaussian filter to the collected data, as shown in Figure 6.2.

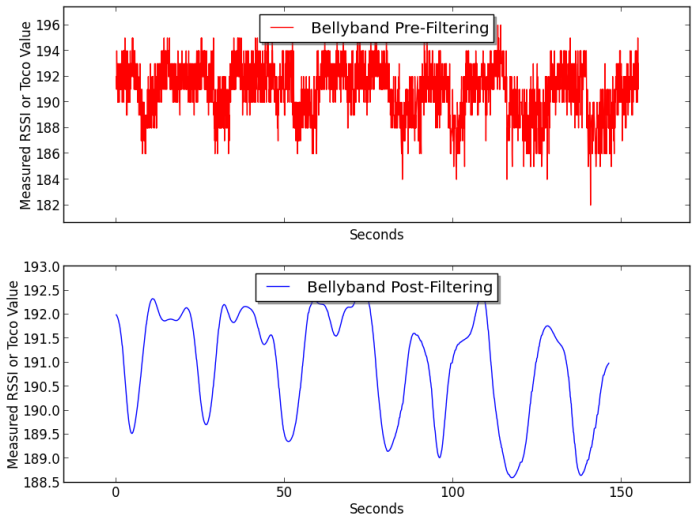

Figure 6.2: The raw data (top) collected from the RFID interrogator shows quantization and noise removed by applying a Gaussian filter (below).

In order to process the data collected and visualized into decisions that can be acted upon or evaluated further by medical personnel, we developed an additional software module that emits a binary signal when it determines that an event such as a contraction or apnea situation occurs. Filtering modules were created to first reduce noise and quantization issues within the data. Our goal is to determine when a certain amount of "stretching" has occurred on the band; quantified by determining the frequency with which the band is stretched by the pump against the known

quantities of our testing script. The rate of stretching is determined by taking the FFT of the collected data, and seeking the frequency with the highest magnitude to determine the observed rate. The FFT has been used to accurately determine the inflation rate of the air bladder, and has been further improved by applying a Gaussian filter to the collected data.

Because a separate web service continuously interrogates the RFID tag and presents that data in the form of a service endpoint that queries windows of data, the statistical signal processing module can incorporate and compare these approaches without disrupting the data collection layer. The SVM library requires a matrix consisting of vectors of features for the training data, in which each vector corresponds to a window of training data collected, and each element of that vector corresponds to a feature such as the mean or standard deviation of that window. Subsequently, a vector of features of data corresponding to the most recent unknown window is provided to the SVM for classification. Adding additional features for study, as was done with the hypothesis testing p-value, is accomplished by augmenting these vectors with additional window feature values. To add additional statistical approaches, additional service endpoints are added to the signal processing module that query the interrogation module as needed for data.

The interactions between the components at the data processing layer are described in Figure 6.3. The **Processor** class is a superclass implemented by a number of signal processing subclasses, and is responsible for receiving augmented data from the **Detector**, appending that data to a data structure provided by the Pandas Python data library, separating the data by antenna, RFID tag, and frequency channel, and notifying the subclass processors in use that new data has arrived for processing. Those subclasses, in turn, perform signal processing, statistical inference, or machine learning algorithms as appropriate to the application, and return aggregate values for reporting, graphical plotting, and emergency alerting by the **Detector** class (for example, an alert might be raised if the respiration rate falls below a threshold such as 15 respirations per minute).

6.2 Comparison of One-Class *vs* Two-Class Classifiers

With this framework we can read heterogeneous signal sources, including LLRP-based RFID interrogators, Bluetooth-based RFID interrogators, and other biomedical sensing devices. A critical

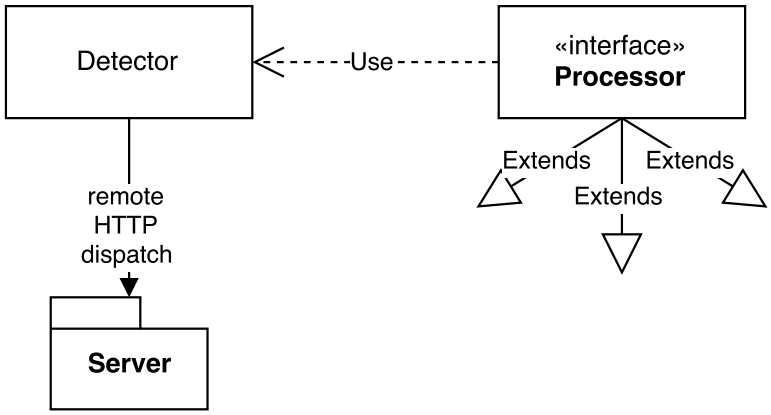

Figure 6.3: Modular implementation of signal processing software that utilizes the web service layer to obtain live RFID interrogation data in real-time; here, the **Processor** is a generic interface that is implemented ("extended") by numerous implementation-specific processors, such as the Short Time Fourier Transform (STFT)

element of this research is the ability to compare RFID data collected to data collected from a traditional medical device. This is accomplished by substituting our reader module with one we developed to read a particular tocodynamometer device using its proprietary protocol over a serial interface traditionally used to support external hospital monitors. Instead of the RSSI value, the transducer pressure monitor value is emitted into the framework's data store, and is filtered and processed using the same framework modules we developed in this work.

For respiratory data analysis, 5 minutes of continuous SimBaby data was collected with the baby breathing from 0-60 seconds, non-breathing from 60-100 seconds, breathing from 100-210 seconds, non-breathing from 210-240 seconds, and breathing from 240-300 seconds. Although this testing setup is not biologically feasible, spending long periods of time in each state facilitates accuracy testing because the class is known *a priori*, although real-time application is feasible. Further, although multi-class classifiers have been applied in our research for comparison, they are impractical for the respiratory application, since it is biologically infeasible to collect training data in the non-breathing class. Instead, single-class novelty detector approaches are employed to detect outliers from the "breathing" training class. The data was taken and cut into windows of various numbers of samples. To measure the separability of statistical features such as the mean and standard

deviation between the "breathing" and "non-breathing" classes, the FDR[134] was applied to several statistical features of the windows, and predicted the mean and standard deviation to be separable and therefore good candidates for our classification analysis.

SVM[71] and hypothesis testing were studied for inferring subject state using RFID interrogation. Other approaches and statistical features exist. For example, k-Means Clustering[132] groups collections of data, like the support vector machine, but then computes the nearest centroid to a new datum point while the support vector machine draws a line or hyperplane to separate the groups. Additional classifiers such as a moving average, decision tree, and elliptic curve are useful depending on the shape of the data one expects to collect; these are discussed when One Class classifiers are investigated in Section 6.2.1. The statistical features to collect for these classifiers were informed by the FDR metric. Features of the data distribution, such as skewness and kurtosis, can inform the shape of data points for comparison to training data using methods like SVM and k-Means Clustering. Because a separate web service continuously interrogates the RFID tag and presents that data in the form of a service endpoint that queries windows of data, the statistical signal processing module can incorporate and compare these approaches without disrupting the data collection layer. The SVM library requires a matrix consisting of vectors of features for the training data, in which each vector corresponds to a window of training data collected, and each element of that vector corresponds to a feature such as the mean or standard deviation of that window. Subsequently, a vector of features of data corresponding to the most recent unknown window is provided to the SVM for classification. Adding additional features for study, as was done with the hypothesis testing p-value, is accomplished by augmenting these vectors with additional window feature values. To add additional statistical approaches, additional service endpoints are added to the signal processing module that query the interrogation module as needed for data. As a prototype effort, the FFT was added to the statistical processor to determine the rate of respiration of the subject in addition to merely the state of the subject's respiration.

The software uses a certain percentage of the data windows as a training set to compute the known classifications, constructs a Support Vector Machine from a training set, whose breathing

and non-breathing states were known, on a given window size, and then checks the accuracy of its predictions against the rest of the collected windows. It was observed that smaller window sizes seem to conflate the two activity states, as evidenced by the shrinking margin for each SVM line. However, smaller window sizes are desirable because it means that predictions can be made on smaller data sets, meaning that alerts can be made in a shorter amount of time. An example SVM separation on training data, and the subsequent data collected and classified, is shown in Figure 6.4.

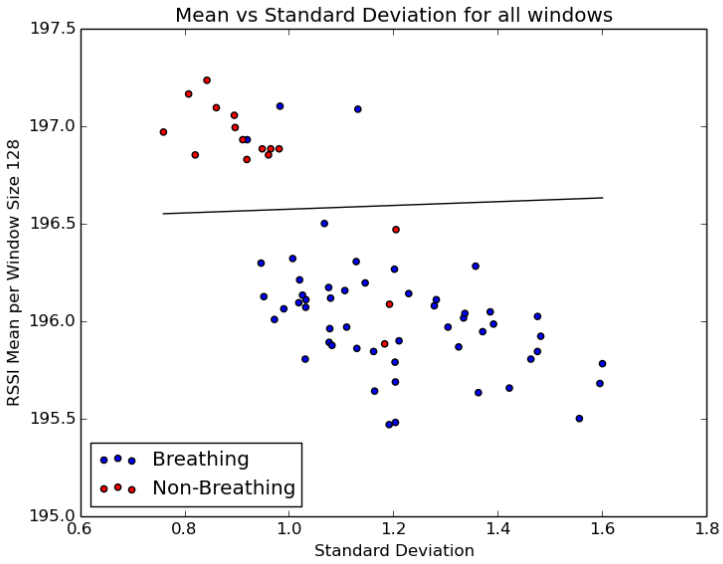

Figure 6.4: SVM "hyperplane" separation of 128-sample windows

A One-Class SVM approach utilizes only breathing training data, since non-breathing training data is infeasible in practice, and was compared against the Multi-Class SVM approach as well as two additional statistical approaches that also allow training only on the breathing class. First, a $\mathcal{X}^2$ test on the standard deviation of each window was compared against the mean standard deviation of a training set of window standard deviations. This test assumes that the subject is breathing as the null hypothesis, and if a sampled window standard deviation is found to be an outlier from this training set, the null hypothesis is rejected and it is predicted that the subject is not breathing. The second approach is a t-test on the mean of each window compared against the mean of a training set of windows. Here, the null hypothesis assumed a 0-mean window after subtracting the training set mean from the window sample mean, essentially indicating that the sampled mean deviates from the

breathing sampled mean (such that breathing is unlikely). Since it is assumed that non-breathing would produce only 0-mean noise, a null hypothesis of non-breathing can be established here. If the mean is found not to deviate from a 0-mean, the null hypothesis is rejected and breathing is inferred.

These tests were run for various window sizes: 128, 64, 32, and 16. For each of these window sizes, different sized training sets were used as well: 25% of the data set, 12.5% of the data set, and 6.75% of the data set. In general, SVM outperforms the predictive capability of the $\mathcal{X}^2$ test alone for all window and testing set configurations. Further, the $\mathcal{X}^2$ test outperforms the t-test, which is explained by the stronger FDR classification metric on the standard deviation of the observed data. With small windows and a reasonably sized training set consisting of only breathing data, and the addition of the $\mathcal{X}^2$ p value as a separable feature, a One-Class SVM was run with approximately 70% accuracy, just under that of the Multi-Class SVM classifier. The One-Class SVM is encouraging in that it does not require non-breathing training data to classify, and performs well with smaller window sizes. This means that detection of breathing and non-breathing states can be done within 4 seconds (given the window size and read rate of the interrogator), following a short training period.

6.2.1 Effectiveness of One-Class Classifiers for Anomaly Detection

As described in Section 6.2, the One-Class Support Vector Machine performed comparably to its Two-Class counterpart, indicating potential usefulness of One-Class anomaly detectors for classifying real-time RFID biomedical data from these passive smart textiles. Several One- and Two-Class classifiers were employed against the data collected to compare accuracy over a number of configurations. Specifically, a One- and Two-Class SVM, a Two-Class Decision Tree, a One- and Two-Class Least Squares Anomaly Detector[135], and a One-Class Elliptic Envelope Anomaly Detector.

The data collected (at between 30-55 Hz) was filtered via a Kalman Filter[126], aggregated into window sizes, and the mean and standard deviation of each window was taken to create new data sets. Window sizes of 30-600 in increments of 30 were taken. The windows either proceeded without any overlap, or a 50% sliding window overlap was utilized for each window configuration. Further, different numbers of the first windows encountered were used as training data, from 10% to 90%

in increments of 10%. It is worth noting that these percentages would inform the amount of time used for training observations in production usage, as one cannot assign a percentage of data when the device is intended for continuous use. In collecting this data, the SimBaby was deployed and programmed to breathe for one minute at a rate of 28 per minute, then to cease respiration for one minute, before repeating.

For many configurations, we observed a correlation between the training set size and Receiver Operating Characteristic (ROC) Area Under the Curve (AUC), as well as between the window size and the accuracy. For One-Class classifiers, this can be met by allowing the subject to wear the device under normal and observed conditions for a specified period of time before entering the classification phase. Because the subject is being observed, the class can be assumed to be the normal (*i.e.*, breathing) state. There was a stronger correlation between the window size and the accuracy, particularly above a window size of 300 (approximately 5-10 seconds at 30-55 Hz). For example, the ROC AUC results for various window size and training set configurations for the One-Class Least Square Anomaly Detector and for the One-Class Support Vector Machine are given in Figure 6.5.

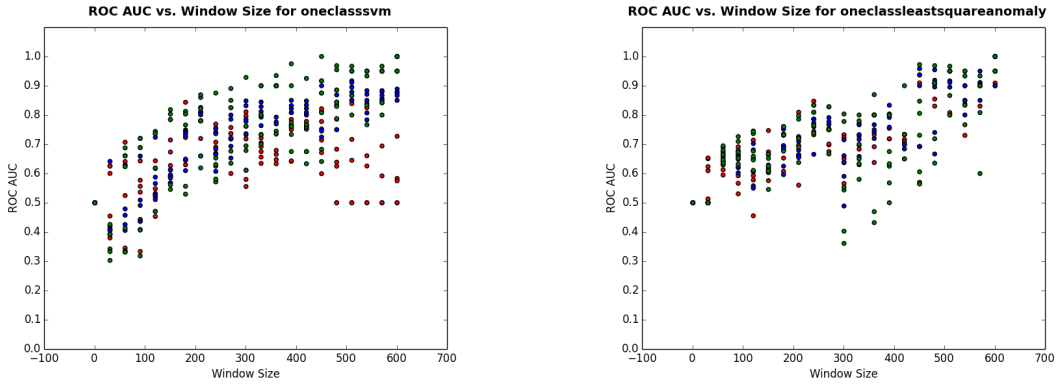

Figure 6.5: One-Class Support Vector Machine (left) and Least Square Anomaly Detector (right) executed over each specified configuration, and ROC AUC plotted against window size. Darker nodes, towards the upper portion of the plot, indicate larger training set sizes.

The SVM constructs a space around the supplied training class data: in this experiment, a certain amount of breathing data from which the mean and standard deviation are computed in various window sizes. If there are several mean/standard deviation combinations that typically

indicate respiration, the constructed learning spaces can be disjoint. This provides flexibility for the anomaly detector by indicating multiple "breathing" spaces rather than constructing a single convex hull around the training data. This approach proves useful here because the standard deviation varies more than the mean when the breathing state class changes. This is because the mean can vary with the depth of breathing, or the proximity to the interrogator, but the individual data points fluctuate from the mean while breathing is taking place. This intuition is confirmed by the FDR, which predicts greater separability on the standard deviation. However, the risk of constructing multiple learning spaces is overfitting, in which the training becomes too specified to accurately detect anomalies. This risk motivates the study of and comparison to the other classifiers discussed in this Section.

The SVM performed better with larger windows, likely due to the sensitivity of the standard deviation to classification as predicted by the FDR. Although its ROC AUC also improves with a larger training set size, even smaller training sets yield reasonable classification with window sizes larger than 300 (approximately 5-10 seconds). By contrast, the One-Class Elliptic Envelope constructs a single ellipse to encapsulate the training data; although its average ROC AUC is 0.59 as compared to the SVM average ROC AUC of 0.71, it performs better with larger training sets without requiring larger windows. It is useful, then, for detecting potential anomalies after a short period of time. This has the potential to inform the classifiers that rely on longer window periods, such as the SVM, using and ensemble learning approach.

The Least Squares Anomaly detector performed better with larger windows, even with moderate training set sizes, and performed comparably to the SVM with an average ROC AUC of 0.69. Finally, the Moving Average Classifier resulted in an average ROC AUC of 0.73, but its performance was scattered among the various window and training configurations. This is likely because this Moving Average Classifier gives equal weight to anomalies detected in either statistical feature chosen (in this case, mean and standard deviation). The classifier could be weighted to assign preference to anomalies detected in the window standard deviation.

6.3 Respiratory State Classification

Classification via Hypothesis Testing on Spectral Features

Using the spectral maximum on a 0.5 second window of data, we collect 20 seconds of training data, observe the mean and standard deviation of the spectral maxima, and run a Student-t hypothesis test on subsequent windows of data. It is assumed that normal breathing is taking place during the 20 second training period, but require no manual intervention for data labeling. The null hypothesis states that this normal respiration continues to take place. If the spectral maximum is a 95% outlier and the sample value is below the observed training mean (because abnormally deep breaths are not under consideration), a cessation of respiration has taken place. Every cessation is not necessarily an apnea condition, because a temporary pause between inspiration and expiration is normal; rather sleep apnea is defined as a reduction in respiratory activity by 95% for a period of 10 seconds or more[136]. To avoid classifying temporary pauses between breaths, an arbitrary limit less than or equal to the 10 second threshold was chosen to consider an apnea detection, and this detection should continue beyond the 10 second threshold. Because the mechanical SimBaby respiratory delays are approximately 0.5 seconds between inspiration and expiration, the expiration-inspiration delay is defined by the respiratory rate, and a minimum respiratory rate of 15 was used, a 4 second threshold was used in this study. An example t-test classification is shown in Figure 6.6.

We chose not to omit small magnitude samples in the training set, although these are likely to be non-breathing "pauses" between inspiration and expiration, because this would require a threshold or percentile and would perturb the training sample set. It is likely that a certain percentile of training data represents these pauses, and this could be trivially considered for training samples in which respiratory activity is particularly sparse.

Using Temporal Information from Posterior Classification Probabilities

Another method to improve classification results is to utilize the temporal relationship within the data. Perceptron-based separation provides a line for classifying data into two or more classes. This line is rigid in N-dimensional (often linear) space, and hence only the magnitude of the data, and not temporal information, is considered. Platt scaling is a technique that takes the output of a

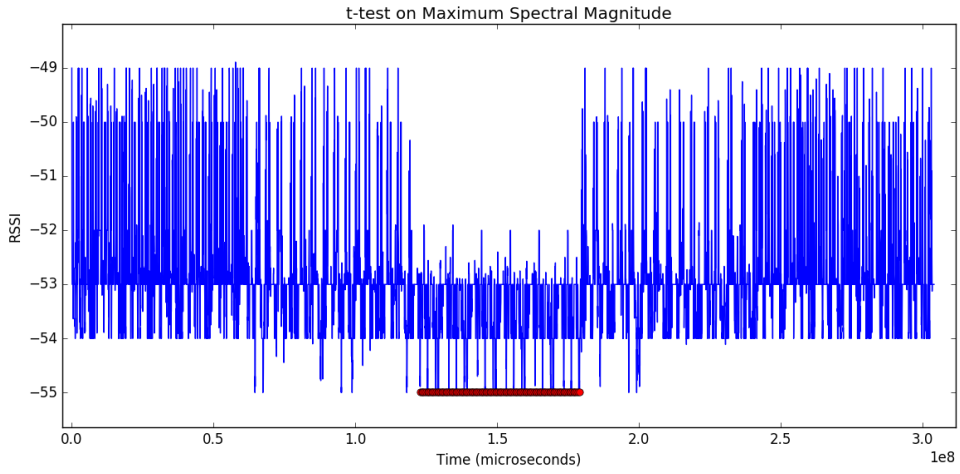

Figure 6.6: Using a t-test on a SimBaby mannequin programmed to cease respiratory activity from 120-180 seconds, breathing at a rate of 15 from 60-120 and 180-240 seconds, and breathing at a rate of 30 from 0-60 and 240-300 seconds; a 20 second training period was used, and detection of respiratory cessation is denoted in red

classification model and transforms it into a class based probability distribution [137]. This technique is applied in the Two-Class SVM in order to correlate the distance of a data point from the separating margin to the probability of the data point belonging to a class. The RSSI readings obtained from the Bellyband is temporally related and temporal trends are captured by using sub-sections of training data to generate linear SVMs. These SVMs prevent the Two-Class SVM from misclassifying data points collected during a transition period, such as between breaths for respiration monitoring. By using a soft threshold or sigmoid function, we can observe the decrease in posterior classification probability, warning of a possible upcoming change in classification. We inject synthetic anomalies as described in Chapter 5 and fit a Two-Class SVM perceptron, with the results fit into a probability distribution using logistic regression [137]. Overall, using probabilistic classification and cross-validated training sets, the ROC AUC for an 8-second window and 25% training set improved to 0.94; the precision/recall AUC was 0.94 (see Figure 12.4).

6.4 Uterine State Classification

Uterine monitoring during labor and delivery is typically done via a visual plot, which we provide as shown in Figure 4.2. However, for purposes of telemedicine, mobility, and automated detection, it is desirable to classify uterine strain gauge data as we did with respiratory data.

To do this, we perform unsupervised classification using a Markov chain. This is a feasible approach over traditional classification because we are classifying coarse-grained changes in the data as opposed to rapid and brief movements as found with respiratory movement. A Markov model can be applied to establish two broad classifications of data within the window, accounting for their temporal progression as well as magnitude. We construct a Hidden Markov Model using a brief period of "training data" which is assumed to be semi-unsupervised, unlabeled, "normal" activity. We then observed the benefit of fusing the tag velocity measurement v with the RSSI power ζ in Chapter 8 when estimating the start and duration of strain gauge activity in the presence of noise artifacts. The classification feature set utilizes this relationship and is given by the tuple $X = [d, r, \zeta, v]$ where d is a vector of Doppler measures, r is a vector of RSSI normalized to the window mean, ζ is the $P_{Rx,reader}$ defined in Chapter 5, and v is the tag velocity computed by the phase and interpolated for any missing values due to changes in frequency, antenna, or responding tag. An example unsupervised Markov switching model classification is given in Figure 6.7.

As this model is the most unsupervised approach taken in this work, its results are not perfectly accurate; rather, the classifications rapidly switch between the classifications between two data samples. However, these classifications can be used as a higher-order feature that can itself be classified using k-means clustering, determining which "peak widths" are noise and which should be merged together into a single movement action. This effort is described in Chapter 10. For classification purposes, however, we build upon this Markov model for temporally-aware classification of RFID data by constructing the MCMC method, which was discussed in Chapter 5 as a method to test whether one or more data distributions exist in a small window of data. This is a useful method to identify periods of time when no band strain activity is taking place. In the case of uterine monitoring, the state classes are reversed from the respiratory monitoring case: with respiratory

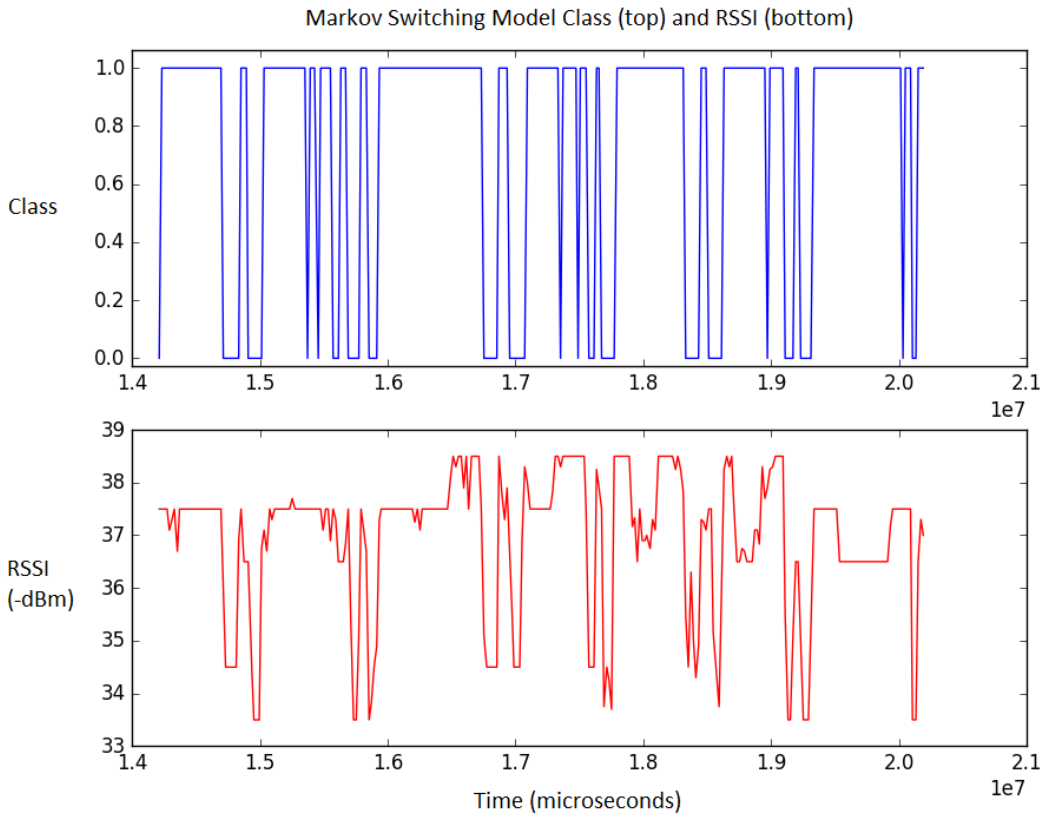

Figure 6.7: Result of a Markov Switching Model applied to RFID data for unsupervised classification of wearer state

monitoring, the primary class is "breathing," or active stretching of the band and knitted antenna. The anomaly class is an apnea case. During labor and delivery, the common case is a stationary band about the abdomen, and the rare case being classified is a uterine contraction, which should stretch the band. This changes the nature of classifier training, and we instead test the hypothesis that only a single Gaussian distribution exists in the data as white noise using MCMC.

MCMC is a probabilistic model that uses randomness to walk the search space, and is not a random algorithm in terms of the data itself. From the MCMC, we determine whether the rate of random-walking is greater or equal than the random-warp probability on convergent solutions. If the rate is greater, the simulation is not converging even on local optima, and the data is a single distribution likely to be white noise across the spectrum. Even if the MCMC simulation does converge, we can inspect the properties of the two distributions identified, because the simulation can assign individual data points to each of the distributions found. The "residual distribution" left over after removing the likely "white noise" data should be oscillatory in nature, reflecting the stretching nature of the band, with some of the noise artifact from the other distribution removed. This residual distribution is useful for determining changing environmental conditions not related to the activity being monitored, so that changes in the environment can be detected for retraining conditions (see Section 6.6).

6.5 Ambient Motion Detection and Cancellation

For detection of ambient, non-stretching motion of the Bellyband, we measure the change in channel-normalized RSSI over time by a **Processor** module that computes its slope over small time windows. Each data window is filtered using LOWESS regression filtering [127], and the slope of the filtered data is used as a classification input. An ANN [138] is trained on the resulting slope data and tag velocity in order to classify those movements that are due to ambient artifacts as opposed to more rapid and oscillatory respiratory movement. These sudden movements can then be removed or disregarded in subsequent respiratory rate processing.

6.6 Training and Re-Training

One goal of this effort was to perform classification while requiring minimal or no supervised training. We refer to the approaches outlined in this Chapter as "semi-unsupervised" because they typically require a nominal period of time, typically 20-60 seconds, of data collection under "normal" activity. An expert is not required to label the data nor to provide any intervention to the classifiers; rather, a lay-person observes the subject during this period to ensure that normal respiratory activity is taking place. During this time, training sample data are collected that define the environment, relative distance from the interrogator (measured in physical properties of the RFID backscatter signal such as RSSI), and, if needed, the relative depth of strain gauge activity. If these properties change, it is necessary to re-train the classifiers, but the unsupervised nature of the classifiers means that the lay-person might not be in the area when these conditions are observed. It is useful to automatically determine when these retraining conditions exist, and the parameters could be useful in automatically re-training the classifiers as well. The Markov methods described in Sections 6.3 and 6.4 provide a useful framework to determine when it is necessary to re-train the classifiers.

The MCMC simulation attempts to fit two distributions to small (*i.e.*, 0.5 seconds) windows of data. When only one distribution exists, no movement is taking place with the band, which was used to inform classification of non-moving state on both respiratory and uterine applications. However, the MCMC simulation also determines the parameters of the two distributions such as mean and standard deviation, as well as which data points are more likely to reside in which distribution. As the window slides in time, the number of data points belonging to each distribution should adjust as more motion artifact is captured in the data window. One of the two distributions should remain mostly stationary except for misclassifications at the boundary by the MCMC simulator, as one of the two distributions is Gaussian white noise in which no motion is captured. The other "residual" distribution, which only exists when movement artifacts are captured, has parameters that change as the window moves. As more data points are captured into this residual distribution, the mean of that distribution increases, and then subsequently decreases as the band is relaxed back to its original shape. If the range of the recent residual distribution mean has fallen below

three standard deviations from that observed during the training period, movement activity depth, rate, or other environmental factors have changed (such as obstructing the path between the tag and interrogator). In effect, this classifier determines when the dynamic range of the band has diminished even after separating noise from the signal. Retraining conditions are observed as shown in Figure 6.8, which was collected using the programmable mannequin SimBaby, programmed to cease respiratory activity from 120-180 seconds. Shallow breaths are also captured outside of this range; they are left here because they correctly align with the data captured, and can be filtered depending on the biological application (*i.e.*, by alerting after a certain period of inactivity for the respiratory application, or after a certain period of activity for the uterine monitoring application).

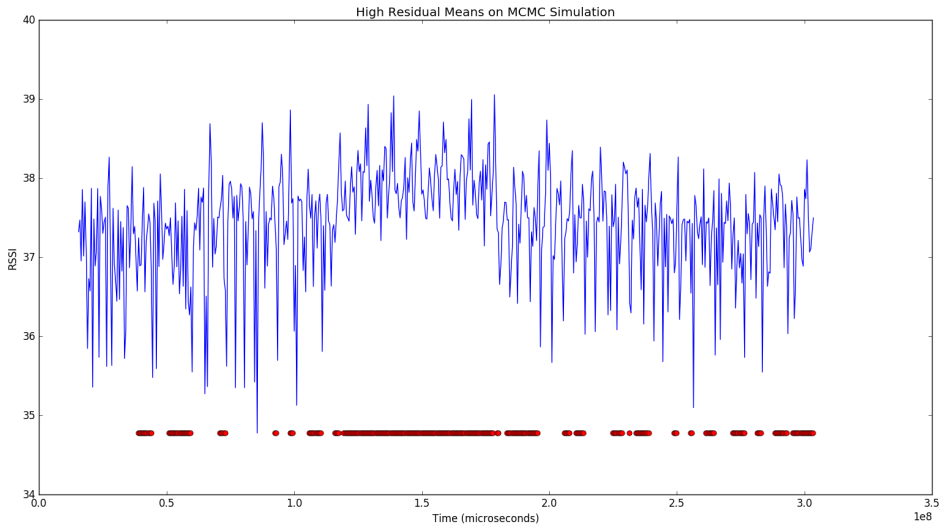

Figure 6.8: Retraining conditions are indicated in red along the X-axis of the mean of the residual distribution after removing noise by an MCMC simulation

Chapter 7: Activity Rate Estimation

The classification approach taken in Section 6 can be useful for classifying anomalies such as cessation of breathing, or detecting a uterine contraction. However, it is also useful to determine the rate of the observed activity. To do this, a Gaussian filter is applied to the data, followed by an FFT. The peak frequencies observed from the FFT are extracted and passed to an Inverse FFT (IFFT) to reconstruct the data with less noise in the signal. The weighted average, or "spectral centroid" is found by identifying the global or local maxima magnitudes within the FFT and computing a weighted average around a specified width w, as shown in Equation 7.1:

$$\frac{\sum_{i=k-w}^{k+w} f_i F_i}{\sum_{i=k-w}^{k+w} F_i} \tag{7.1}$$

for FFT local maxima magnitude F at frequency bin i, and corresponding frequency f. The frequency bin k is chosen by $k = \operatorname{argmax} F$. Because spectral energy for respiratory analysis tends to be concentrated at lower frequencies, the two or more frequencies with maximum FFT magnitude are likely to be close together without being immediate neighbors or within the spectral window w. Because the sliding window will better center these competing oscillations at different times, it is desirable to capture the one(s) most centered in the current time window. To account for this, K is first chosen as a set of locally maximum frequencies relative to the initial choice of k, as shown in Equation 7.2. Subsequently, k is selected as shown in Equation 7.3, where p is the spectral phase of the corresponding FFT frequency. The spectral centroid is used to predict the rate of RFID antenna stretching, as shown in Figure 7.1.

$$K = \forall_{x \in F} | F[x] \geq \frac{1}{2} F[k] \tag{7.2}$$

$$k = \underset{k \in K}{\operatorname{argmin}} |p[k] = \min p[K] \tag{7.3}$$

Spectral density rate estimation suffers from a few challenges, which we address in this Chapter and via sensor fusion, discussed in Chapters 8 and 9. First, spectral analysis of Gaussian white noise theoretically presents a uniform power distribution; however, in practice, some perturbations are possible, and non-Gaussian interference will contribute to the observed spectral density. Second, the windows considered are small (*i.e.*, 0.5 seconds), and may not encompass a full oscillatory respiration. Parts of these artifacts will still be extracted via the FFT for a sufficiently sized window, but it cannot be known for sure what size window will suffice, and lengthing the window causes a lag in convergence to the true respiratory rate. For comparison purposes, we augment the FFT with a discrete approach that considers 0.2 second windows of data and takes the first order difference, counting the rate at which these small tangent lines change the sign of their slope.

The Fourier Transform frequency resolution is dependent upon the size of the window and the sampling frequency. The sampling frequency of 20-100 Hz is higher than the biological phenomenon being observed; however, it is desirable to use a short window for purposes of real-time estimation. The small window and relatively high frequency limits the number of frequency bins available, and lowers the frequency resolution by increasing the spacing between subsequent frequency transform bins. The spectral centroid interpolation method captures spectral information from adjacent frequency bins by weighting via their relative magnitudes. Quinn [139] improves upon spectral estimation with asymptotic mean square error proportional to the inverse cube of the window size, and then combines two such estimators using the previous and the subsequent frequency bin by minimizing the covariance matrix [140].

The periodic rate of a physical phenomenon can be estimated from successive sliding windows of data by using spectral analysis. The result is a series of frequency estimates from which the overall rate is inferred. Quinn's interpolation method [140] reduces variance about the true frequency to near the Cramér Rao bound; however, significant noise artifacts in the data due to multipathing and environmental changes cause fluctuations in individual frequency estimates. Because the window

size is small, even a small variance in the estimated frequency lead to large rate estimation errors.

To overcome non-stationary perturbations in the estimated frequency over time caused by static and dynamic environmental artifacts yielding multipathing, a Hidden Markov Model may be applied to identify more likely progressions of identified frequencies in which transitions among a stationary set of frequencies is more probable[141]. The Markov Chain is initialized to assume that the frequency track is a random walk whose largest possible departure is bounded[142], while larger observed steps are discounted as possible outcomes. The likelihood of each observed frequency estimate, then, is proportional to that frequency's contribution to the overall spectral magnitude. The Baum-Welch algorithm applies Bayesian inference to obtain the most likely current frequency given the current observation and historical "track" of frequencies. The algorithm provides an iterative approach to frequency estimation, as the track probabilities evolve over time and thus depend only upon the most recently observed frequency and the immediate predecessor. Because historical frequencies are sometimes discarded, the true track can become lost if a significant frequency is inferred to be a noise artifact. This leads to a situation in which all frequency estimates are equiprobable or could even sum to 0. When this occurs, the entire track is re-initialized and begins to re-converge from the current starting point; this results in higher frequency estimate variance during this period. The estimates are not used during these convergence periods. When algorithm convergence can introduce such estimation errors, or when environmental factors lead to algorithm retraining conditions (see Section 6.6), other estimation approaches are applied and fused, as discussed in Chapter 9.

Quinn's frequency interpolation[140] of Fourier transformed spectral respiratory data is used to identify locally optimal frequency estimates of the current short-time window, and the resulting interpolated frequencies and magnitudes are applied to a Hidden Markov Model to track frequency progression over time[141;142] to obtain a maximally likely frequency estimate. The resulting algorithm for interpolated frequency estimation using a Markov chain is given in Appendix A.

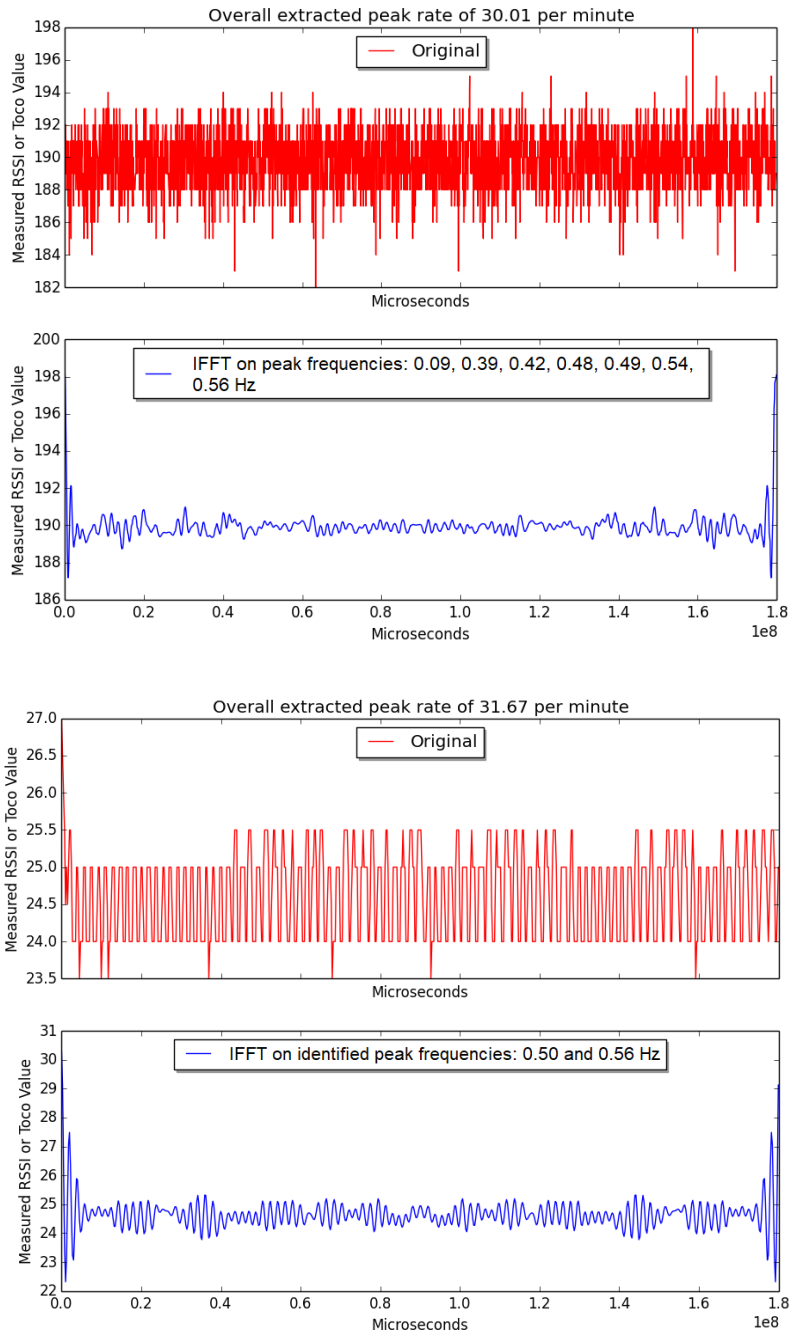

Figure 7.1: Bellyband data collected and filtered to estimate the target respiration frequency of 0.5 Hz (top), and tocodynamometer data simultaneously collected and filtered from the same mannequin, also estimating the same target respiration frequency of 0.5 Hz (bottom); the IFFT average yielded a predicted rate of 30.01 respirations per minute (top), as opposed to an estimated ground truth of 31.67 respirations per minute (bottom)

Chapter 8: Single-Sensor Multi-Measure Fusion

Spectral analysis of the signal as shown in Chapter 7 assumes that the signal is mostly "clean;" interference is likely to be dominated by a movement frequency, but unrelated "ambient" movement artifacts interfere with the dominant frequency selection, especially if the window under consideration is short as is conducive to real-time estimation. One naïve approach to separating oscillatory signal from ambient motion artifacts is to remove trendlines from short time windows of the data. Indeed, this has been effective in removing some such artifacts; however, short bursts and circular movements may not be identified from the signal in this way. However, perturbations in the tag velocity v (Equation 4.2) and signal strength ζ (Equation 4.3) can be fused and classified into activity type, including knit antenna stretch. We begin by computing δ, the first-order difference of the correlation between short windows of the tag velocity and signal strength (see Equation 8.1).

$$\delta = corr(\Delta\zeta, \Delta v) \tag{8.1}$$

As the band stretches, the tag velocity and signal strength are correlated, resulting in well-defined perturbations in the correlation measure during these periods. A rolling RMS is computed from the short-time δ series, and the resulting measure is converted to a square-wave binary classifier by identifying the inflection points of the rolling RMS (identified as the zero-crossings of the second-order difference). An example of such a classification is shown in Figure 8.1.

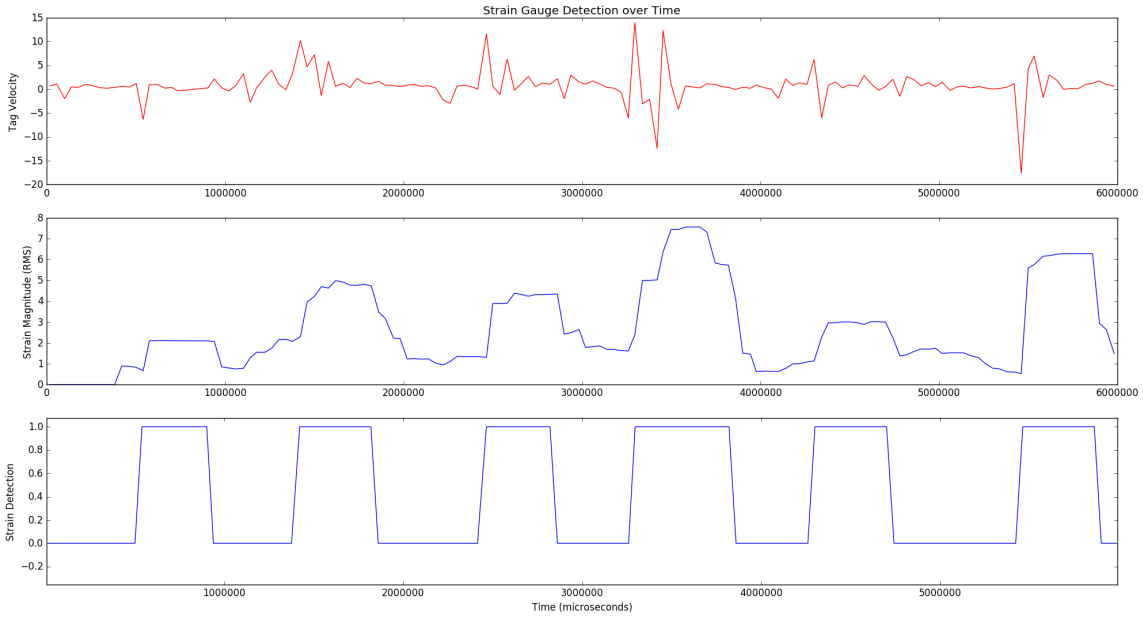

Figure 8.1: Correlating the tag velocity and signal strength (top) to form a binary classifier of strain gauge motion artifacts (bottom) by locating the inflection points of a rolling RMS measure (middle) against δ

Chapter 9: Multi-Sensor Fusion

The rate estimation using spectral analysis presented in Chapter 7 accurately computes the rate of band stretching (*i.e.*, respiratory rate) using interpolated[140] Fourier Transform peak frequencies de-noised in the time domain using a Markov Chain[141;142]. The accuracy of the estimate varies with the length of the window being considered; however, a longer window also concedes the assumption that respiratory activity is periodic, when in practice it is oscillatory with small perturbations in the sinusoidal period. Therefore, it is necessary to augment the spectral estimate with additional time-domain measures to detect small shifts in period. In this chapter, we discuss and fuse these additional measures for a unified rate estimation that improves upon the accuracy of the spectral analysis alone.

Our fusion approach[25] is based on Expectation Maximization of a multivariate Gaussian Mixture Model (GMM) constructed from point estimates from each measurement estimate. It is assumed that each measure is subject to both environmental (for example, multipath perturbations or a moving subject) and process estimate (for example, spectral leakage or a spurious peak) noise that must be considered stochastically. However, each measure estimate is informative of the true underlying measurement, and can be modeled according to their relative historical variances. The variances of each measure should be about equal even during subject movement, as the underlying true measure should vary with each individual measurement if they are completely correct. Therefore, the variance of the recent history of each measure is used as an estimate of uncertainty of the current point estimate. The point estimates themselves are used as the means of Gaussian distributions with variance equal to the recent historical variance. These are fused by constructing a GMM from these individual Gaussian models, with a full covariance matrix computed to consider which measures vary most closely together to better inform the fused result.

The likelihood L of each measure is computed against the GMM, and the final fused estimate y is a weighted average of the individual measure point estimates μ_i for each measure i, as shown in

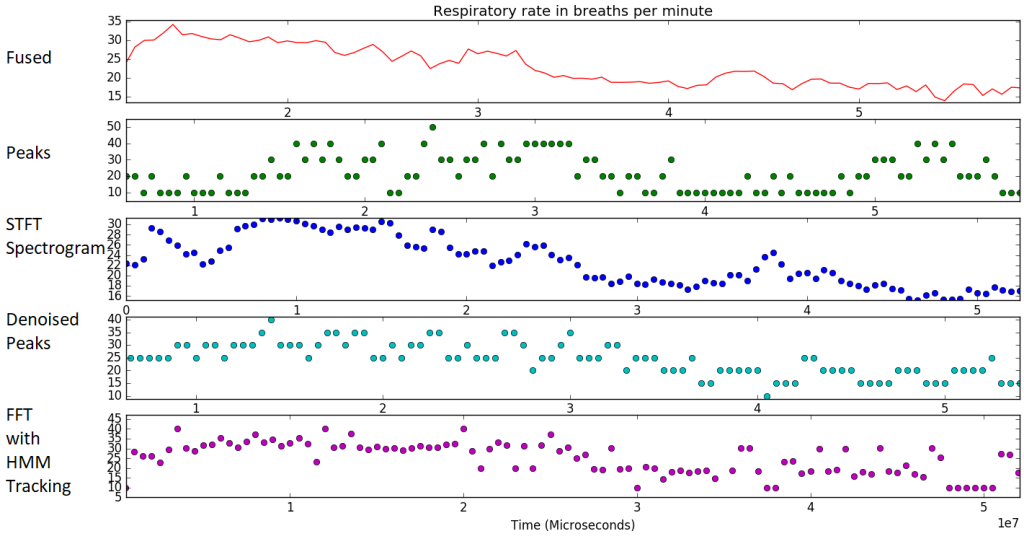

Figure 9.1: Results from the Expectation Maximization sensor fusion algorithm removing noise artifacts from individual measurement estimates

Equation 9.1. The vector of fused estimates y is then filtered using a Kalman filter to obtain the final tracked estimate. By incorporating the recent variance of the measure estimate, this approach can improve estimation accuracy even when using only a single measurement estimate.

$$y = \frac{\sum_{i=1}^{n} \mu_i (\log L)^{-1}}{\sum_{i=1}^{n} (\log L)^{-1}} \tag{9.1}$$

An example sensor fusion is shown in Figure 9.1, in which a human subject breathed at a rate of 30 per minute followed by a rate of 15 per minute, for one minute intervals. Here, peak detection and spectral analysis are fused with their denoised and Markov Chain tracked counterparts (in which signal is sacrificed to reduce the signal to noise ratio).

9.1 Fusion Framework Architecture

To facilitate the rapid prototyping and deployment of wearable intelligent measurement algorithms, we designed a Fusion Framework package that interfaces with the secure data and physical layer, synthesizing data from heterogeneous IoT sensors and measurement algorithms to compute a fused

estimate. The reference architecture for this framework is shown in Figure 9.2, and includes the following modules:

- **Measure**: The Measure module implements an estimation algorithm using data that is automatically provided to it. This abstraction enables the rapid prototyping of filtering and estimation approaches independently of physical layer details. This module is extended by any number of subclasses, each of which implements a single algorithm with only a single required *process*() method.

- **Perturber**: The Perturber module is an optional module and implemented by one or more subclasses that simulates the effects of fading due to multipath, shadowing, or changing environmental conditions.

- **Fuser**: The Fuser module collects recent histories from the Measure modules and applies a multisensor fusion algorithm to compute a fused estimate from the noisy measures and underlying data.

- **GroundTruth**: The GroundTruth module provides time-specific error calculations against a known record, if one exists.

- **Sensor**: The Sensor module defines the fusion configuration parameters, including which measurement algorithms to apply, if and how to perturb the input data, what ground truth to apply for accuracy measurement, and how to fuse those measurements together. The Sensor module is equivalent to a real-time **Processor** module used for real-time classification (see Section 6.1).

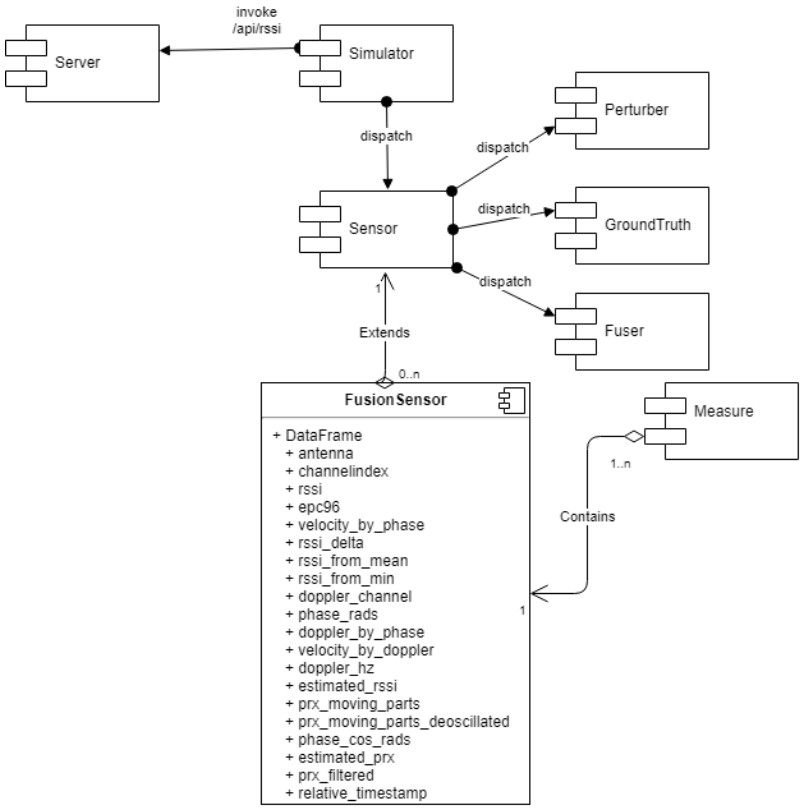

Figure 9.2: An architecture diagram of the Fusion Framework that interfaces with the secure data collection and interrogator driver module and respiratory rate measure fusion algorithm modules

Chapter 10: Prediction of Interbreath Interval

We classify the beginning and end of strain-gauge activity using the Markov Switching model developed in Section 6.4 (see Figure 8.1) and the correlated features defined in Chapter 8 (see Figure 6.7). Once respiratory activity is accurately classified for an infant, the interbreath intervals (IBI) between each breath can be used to probabilistically model a predictive estimate of the next breath. This is useful both as a predictor of upcoming respiratory activity for use as a closed-loop feedback system for respiratory therapy devices, and as a predictor of apnea due to predictable patterns in the IBI variance (namely, that IBI variance becomes unstable immediately before an apnea condition) [143]. Interbreath interval in infants has been modeled by a decaying power law [144], with the associated tail of this model fit to a log-normal distribution [143] and analyzed using an autoregressive model [143].

The log-normal probability density function modeled from the power law IBI relationship is given in Equation 10.1 [143], for IBI history H at epoch k, time t, breath timestamp u, and fitted $AR(p)$ model μ of recent IBI times.

$$f(t|H_k) = (2\pi\sigma^2(t-u_k)^2)^{-\frac{1}{2}} e^{-0.5\times ln(t-u_k)-\mu(H_k)^2\sigma^{-2}} \tag{10.1}$$

A programmable mannequin SimBaby was used to breathe at a pre-programmed rate, with classified respiratory activity shown in Figure 10.1. Even when breathing at a constant rate, variations are expected in the interbreath interval. As before, mechanical artifacts, environmental changes, and process variance all lead to noise inherent in the signal. Therefore, the observed IBI and the modeled prediction are each somewhat inaccurate but informative.

To more accurately estimate the observed interbreath interval, the generated square wave is amended with k-Nearest Neighbor clustering to detect the peak widths in an unsupervised manner. Outlier peaks which are likely to be artifacts are merged with their largest neighboring plateau, whether it is a "high" peak or a "low" peak. To incorporate the noisy current observation with

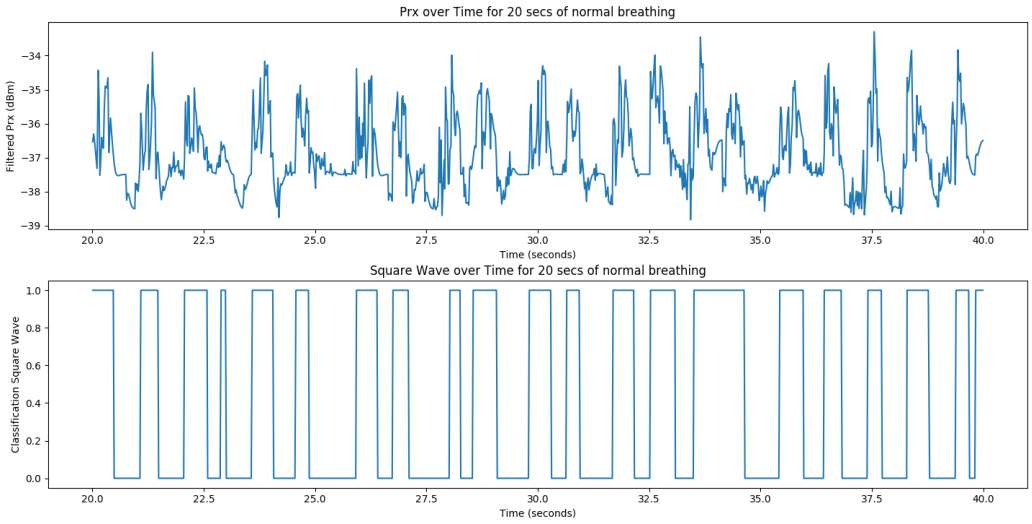

Figure 10.1: Respiratory activity classified as a square wave for interbreath interval prediction

the prior predictive estimate into a fused estimate for use in IBI modeling of the next predictive IBI, we construct an Affine Projection adaptive filter[145] from the observed IBI x_t, predicted IBI y_t, and error $e_t = x_t - y_t$ at epoch t. At each epoch t, the observed value x_{t-1} and prior IBI history H are used to predict the next IBI y_t. An example of predictive convergence of the estimated IBI values according to the probabilistic model[143], and the convergence of the adaptive filter are shown in Figure 10.2.

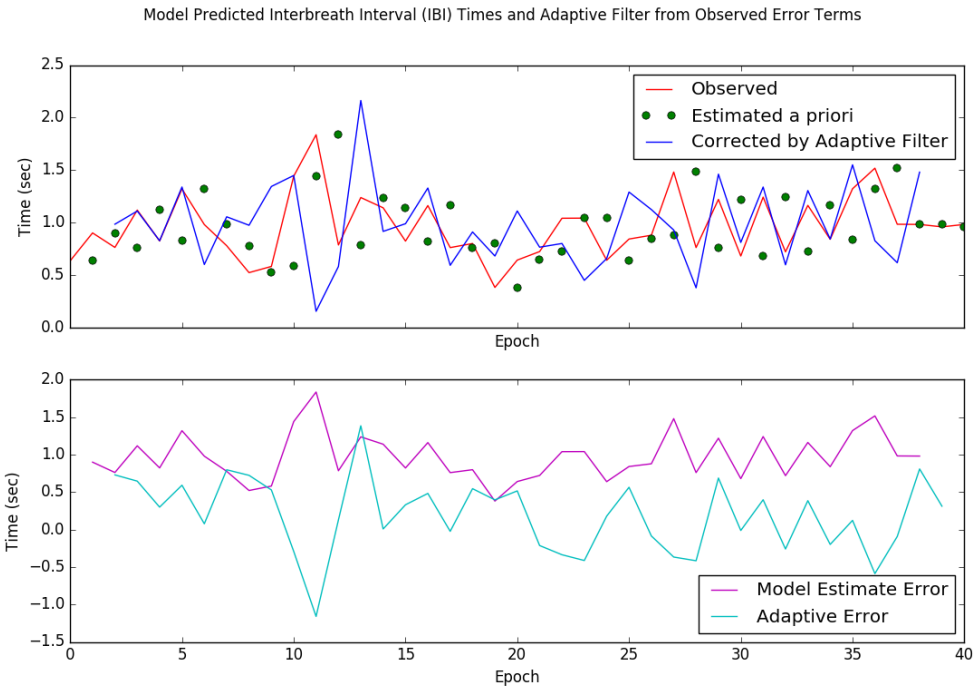

Figure 10.2: Convergence of adaptive filtering algorithm using an autoregressive model of probabilistically estimated IBI according to a log-normal historical distribution

Part III

Experimental Protocol

Chapter 11: Experimental Validation and Human Experimental Protocol

We designed an experimental protocol to evaluate and compare the effectiveness of our smart garment designs as well as the effectiveness of our statistical processing approaches. As discussed in Chapter 2, a Laerdal SimBaby programmable mannequin and a manually actuated pregnant mannequin are used for laboratory testing of the software and physical sensors. For comparison to ground truth, a respiratory monitoring device is used to establish baseline respiratory activity, and a Philips 50XM Tocodynamometer is used as a pressure sensor during uterine monitoring and during mannequin simulations and experimentation. The Tocodynamometer used is shown in Figure 11.1. The SimBaby and pregnant mannequin can be seen in Figure 1.3, and an example experimental data collection from the RFID Bellyband and Tocodynamometer from a pregnant mannequin is shown in Figure 4.1, with post-filtering results shown in Figure 4.2. In all cases, the RFID interrogator antenna is positioned between 1.5 and 3 feet from the Bellyband tag, and oriented to point toward the chip and knit antenna.

Figure 11.1: A Philips 50XM Tocodynamometer

Simulated infant respiratory experimentation consisted of applying a smart-fabric Bellyband to the abdominal area of the SimBaby, and the Laerdal SimBaby software (see Figure 11.2) preprogrammed via a scenario to automatically iterate between respiratory patterns at defined intervals. This enables the forced presentation of biologically undesirable phenomena such as apnea.

For classification experiments, a SimBaby scenario was created that oscillated between periods

of respiratory activity at a rate of approximately 30 breaths per minute for a period of one minute, followed by a period of one minute without respiratory activity, repeating for a predefined period of at least 3 minutes. For respiratory rate estimation experiments, this scenario was modified slightly to incorporate different respiratory rates for pre-defined intervals. As a canonical example, the SimBaby was programmed to breathe at a rate of 30 per minute for one minute, followed by a rate of 15 per minute for 1 minute, followed by a one-minute prolonged apnea condition of no respiratory activity, followed by 15 per minute for 1 minute, and, finally, 30 per minute for 1 minute. Although the SimBaby does have some small local variation in its respiratory activity even with a pre-programmed rate, the mannequin does actuate the correct number of times in each programmed period. The estimator Fusion Framework described in Chapter 9 imports a ground truth configuration for error estimation that aligns with the scenario used by the SimBaby. Movement artifacts were simulated via scripted manual movements of the SimBaby in space by pulling the surface upon which the SimBaby rested through space, out of the way of the interrogation equipment.

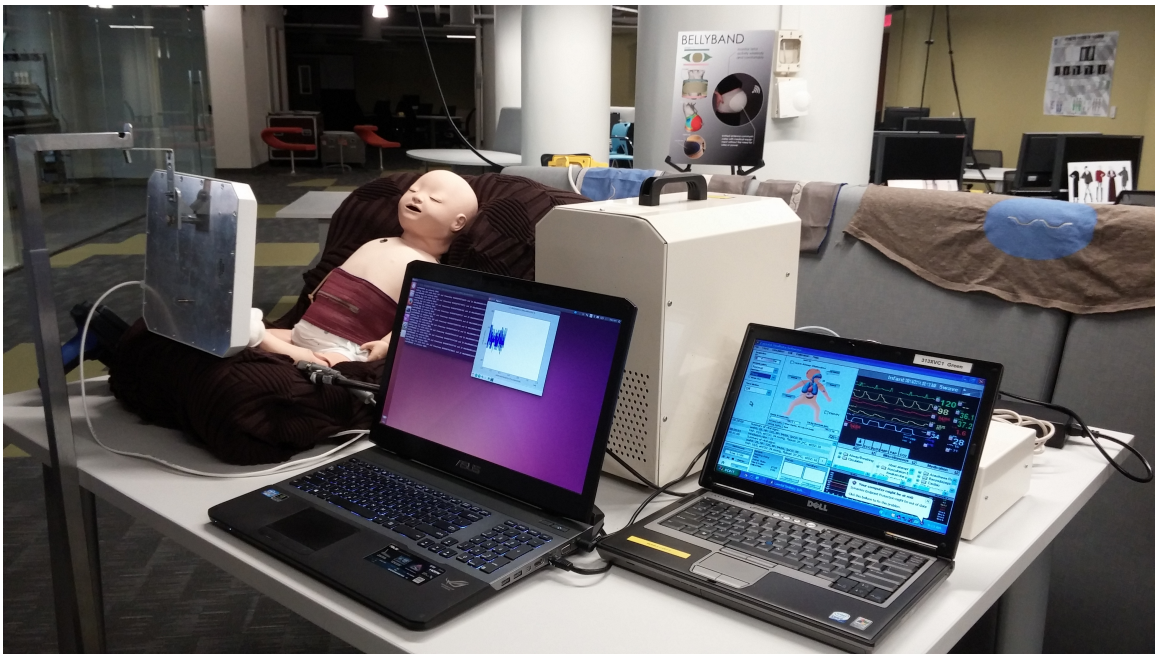

Figure 11.2: The SimBaby control software (right) configuring the automated respiratory patterns of the SimBaby mannequin (left), with RFID monitoring visualization depicted (center)

Simulated uterine experimentation consisted of applying a smart-fabric Bellyband to the abdom-

inal area of an inflatable mannequin as well as applying a pressure transducer of a tocodynamometer to the same area, and manually inflating a mannequin about the abdominal area at some approximately periodic rate. Data collection modules for the Bellyband and for the tocodynamometer, developed as described in Section 4.3, allowed for simultaneous data collection, storage, and visualization of both sensors.

Both the tocodynamometer and the RFID RSSI measurement are reported as 8-bit integers. Because the values are filtered according to a desired filtering strategy, such as a multi-pass moving average like a Gaussian filter, or by a Kalman filter, they may present with decimal resolution. An example dataset collected from the pregnant mannequin using both the Tocodynamometer and the RFID-based Bellyband is shown in Figure 11.3

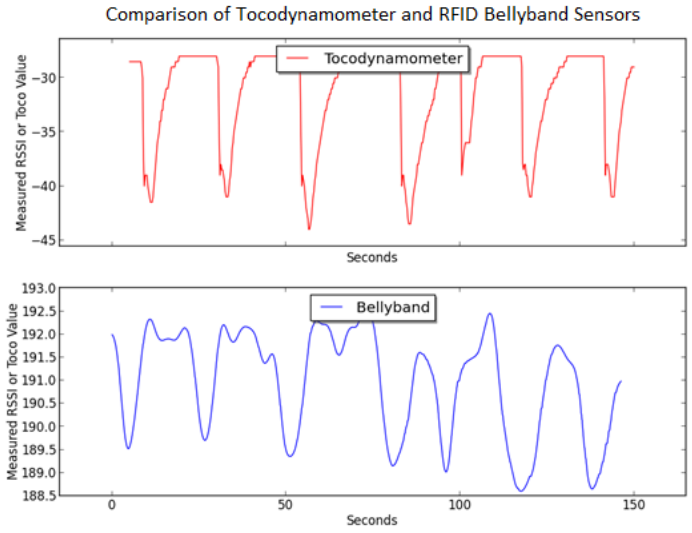

Figure 11.3: Tocodynamometer data collected from a programmable mannequin, at a range of approximately 3 feet (top), and Bellyband data simultaneously collected and filtered (bottom)

An example respiratory rate estimation was performed using the programmable mannequin via the spectral centroid method (Equation 7.1). This was calculated on both the tocodynamometer data and the simultaneously collected RFID Bellyband data. The mannequin was actuated at a rate of 30 stretches per minute, at a distance of approximately 3 feet from the interrogator. Although the spectral magnitude was spread out across multiple frequency bins in the case of the RFID Bellyband after low-pass filtering, the magnitudes themselves were preserved by the reconstruction

of their respective spectral centroids. An example trial is shown in Figure 7.1.

11.1 Human Subjects Experimentation

We programmed the Laerdal SimBaby with biologically infeasible scripts to obtain known ground truth for simulated experimentation. For example, we programmed the SimBaby to breathe at a known rate for a specified period of time, followed by a specified and prolonged duration of apnea. This scripted mannequin experimentation provided several benefits for this study: first, biologically infeasible or experimentally unethical scenarios such as prolonged apnea periods can be readily simulated to measure classification accuracy. Second, experimental runs can be repeated in a consistent way, allowing for verification of experimental results across several data collections and varying environmental conditions. However, the mannequin movements do not purely mimic natural biological movements, and the internals of the SimBaby are mostly metal components that do not approximate the water-based composition of the human body. As a result, the RF measurements, and even the ground-truth pressure measurements of the tocodynamometer, are not a prefect recreation of biological phenomena.

Institutional Review Board (IRB) protocols were submitted and accepted in support of this effort, and an additional protocol was submitted in indirect support of this project[1]. The two primary protocols deal with infant and uterine monitoring using the Bellyband, and the third protocol addresses heart and respiratory combined monitoring of adults (and, eventually, infants). The infant and uterine monitoring protocols are to be carried out at Hahnemann Hospital at the Drexel University College of Medicine campus, and the heart and respiratory protocol can be implemented in the research lab on the Drexel University main campus.

We utilize the REDCap secure research data repository for housing data collected from human subjects. REDCap is typically intended for manual input of observed or survey data, so a software module was integrated with the data collection system described in Section 4.2 to provide real-time data input to REDCap via its RESTful API. Additionally, REDCap was configured to accept Comma Separated Value (CSV) files containing batch database exports post-capture.

[1]The Drexel University IRB protocol numbers related to this effort are 1504003601, 1504003602, and 1604004440, and their approvals and consent information can be found in Appendix C.

For human data collection, it is recommended to keep the interrogator antenna and human subject separated by 50 cm, which results in a maximum peak Specific Absorption Rate (SAR) of 0.25 W/kg, within the maximum SAR of 0.8 W/kg[146]. Additionally, the FCC publishes the Maximum Permissible Exposure (MPE)[147] of $1500^{-1}f\ mW/cm^2$ for frequency f in MHz on average over a period of 30 minutes. In the 902-928 MHz band, this equates to a minimum MPE of 0.6 mW/cm^2. An RF power meter confirmed that RF power P received was well within these limits at all times at distances of 50 cm to 100 cm, near the theoretical value of 0.03 mW/cm^2, computed via Equation 11.1.

$$\begin{aligned} P_{received} &= P_{transmitted}[(4\pi r^2)^{-1}] \\ P_{received} &= (1\ W)[(4\pi \times (0.5^2\ m^2))^{-1}] \\ P_{received} &= 0.3\ W/m^2 = 0.03\ mW/cm^2 \end{aligned} \tag{11.1}$$

For adults and infants, the Bellyband is deployed about the abdominal area, and the RFID interrogator antenna is positioned via a rotating tripod mount to orient toward the knit antenna and at a distance of at least 50 cm from any part of the body. This distance is measured and the RF power measured with an RF power meter to ensure it is within the MPE maximum for the frequency band in use. Data are collected in 3 minute intervals for up to a period of 15 minutes. Ground truth data is extracted depending on what is available at the facility, either via a commercial respiratory monitor, hospital ventilator, and/or manually using a metronome and taking note of physical phenomena.

Part IV

Results

Chapter 12: Experimental Results

12.1 Ambient Motion and Re-Training Detection

In Chapter 6, we discussed an approach for classifying band stretch state as well as baseline "training" (in Section 6.6) parameters from which those classifications are made. We also note that ambient motion artifacts (in Section 6.5) can alter these baseline conditions, causing them to be recalculated or entry into a brief semi-unsupervised state.

MCMC simulation is particularly useful for uterine monitoring since it is entirely unsupervised, and because the anomaly class is the motion-based class, which hinders our semi-unsupervised approach that assumes that oscillatory motion is taking place like in respiratory. It is also applicable to the detection of stretching movements involved with respiratory activity, and in detecting shifts in baseline training conditions. This is because the MCMC simulation was structured to separate short windows of data into one or two probabilistic distributions. If the solution failed to converge, this is a potential indicator that stretching motion is not taking place during that window, because the band is left stationary in whatever position it held previously. If the simulation does converge upon two candidate distributions, those two distributions can be evaluated for their properties and their evolution over time, including the mean, standard deviation, and relative number of data points fit to that distribution from the data window under inspection.

Example MCMC simulation results are given in the plots of Figures 12.1 through 12.2, in which a human subject made abdominal motions at a given rate (10, 15, 20, and varied, respectively) for 30 seconds, followed by a period of non-movement, and repeating. The periods of non-motion are found at time periods 30-60 seconds and 90-120 seconds.

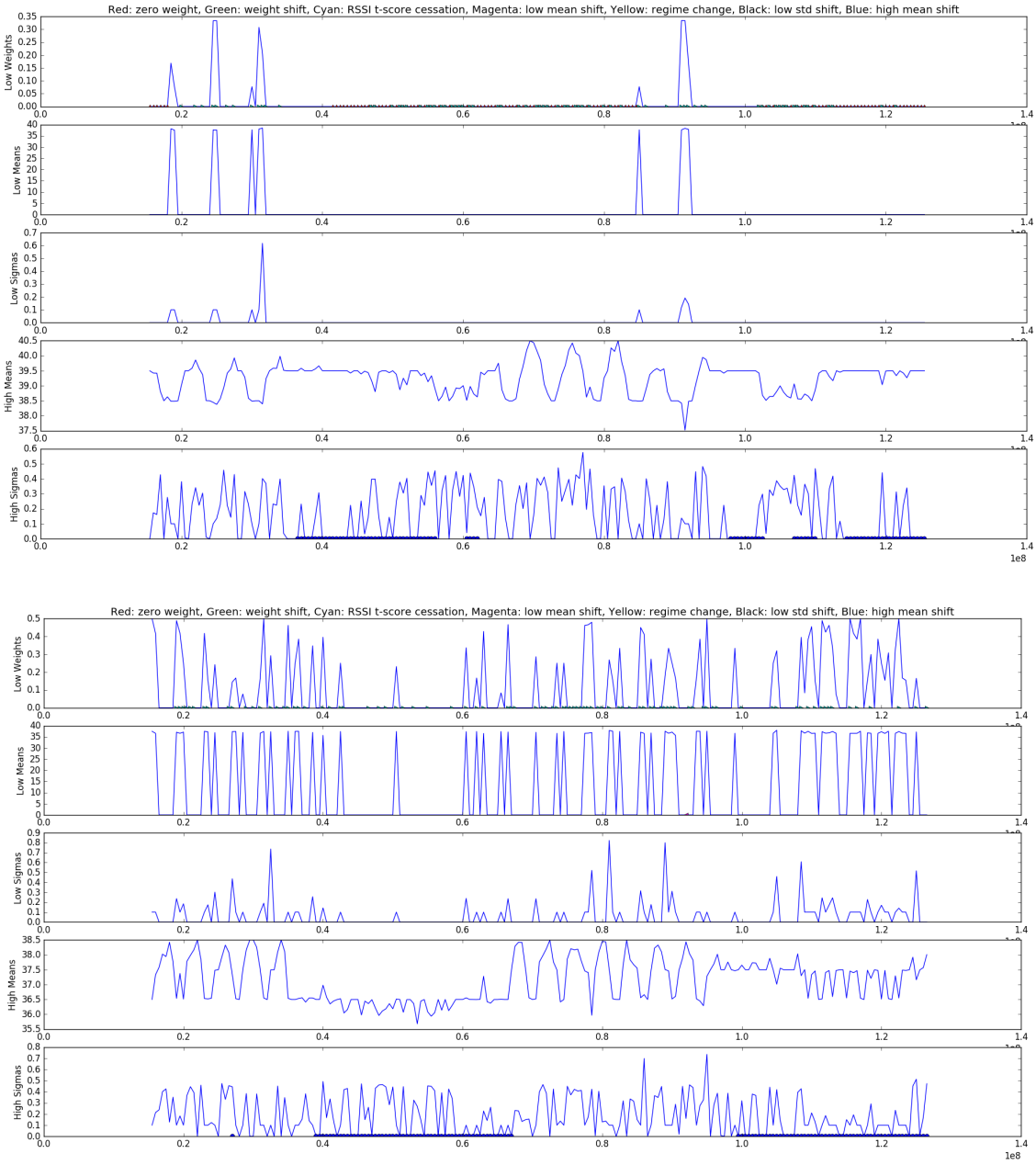

Figure 12.1: MCMC simulations for a human trial at a respiratory rate of 10 per minute (top) and 15 per minute (bottom) with brief cessations: each plot depicts the "low magnitude" distribution percentage of data or "weight," mean and standard deviation, and the residual "high magnitude" distribution mean and standard deviation; respiration is used to simulate band stretching activity in lieu of testing in active labor and delivery, and stationary classes are denoted in blue across the bottom of the lower plot

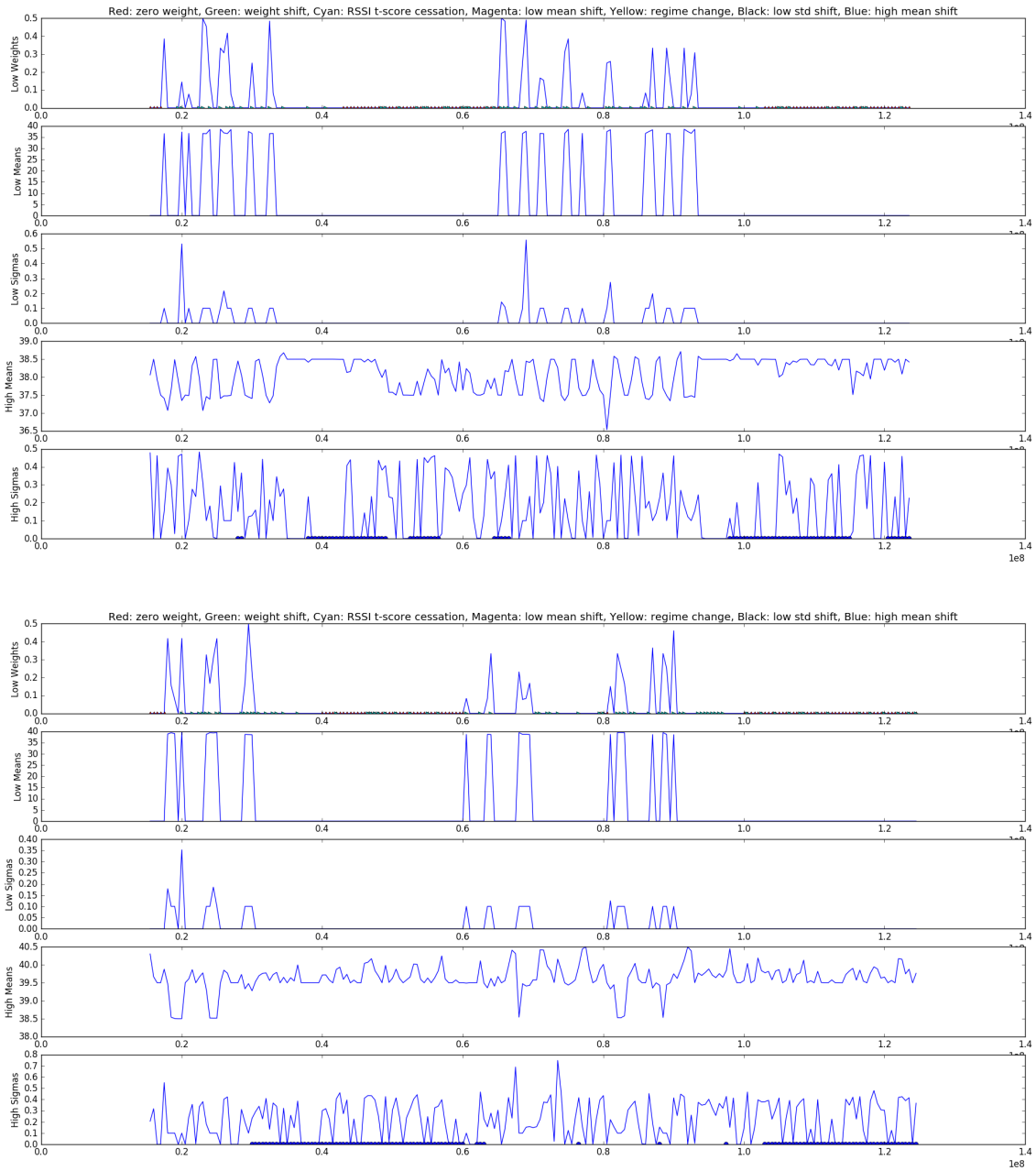

Figure 12.2: MCMC simulations for a human trial at a respiratory rate of 20 per minute (top) and a varying rate (bottom) with brief cessations: each plot depicts the "low magnitude" distribution percentage of data or "weight," mean and standard deviation, and the residual "high magnitude" distribution mean and standard deviation; respiration is used to simulate band stretching activity in lieu of testing in active labor and delivery, and stationary classes are denoted in blue across the bottom of the lower plot

The "weight" measurement indicates the number of data points classified into each of the two distributions, if the MCMC simulation converged upon a fit of data into those distributions. If no

such solution was identified, the weight is returned as 0 and the entire distribution is assumed to be a noise component. This is generally identified in between breath periods and during periods of apnea, as shown in the plots of Figures 12.1 through 12.2. However, shallow breaths, noise, ambient motion artifacts, and the probabilistic nature of the MCMC simulation may result in a false classification of only a single distribution. This problem is exacerbated by the small duration of the data window used (*i.e.,* 0.5 seconds), in which a portion of a breath could be conflated as outlier points fitting into a single Gaussian distribution. This simulation will be useful in informing the band state classification algorithm results, described in Section 12.2.

Additional MCMC simulation results are utilized to detect retraining conditions. If the mean or standard deviation of the simulated distributions shifts over time, this could indicate a fundamental change in the band state, such as ambient movement or a change in environmental conditions, requiring a new semi-unsupervised baseline training. Training is semi-unsupervised in the sense that an expert is not needed to monitor data collection from the band; however, it is assumed that a brief period (*i.e.*, 30 seconds) of "normal" activity is collected at the start of the observation to support the establishment of baseline environmental conditions.

Retraining conditions are also indicated by the introduction of ambient motion artifacts. In this case, the retraining state could be in the form of automated adjustment of baseline parameters, or calibration of collected data to remove the artifact. Using LOESS filtering, we are able to track shifts in the mean of oscillatory patterns such as respiration due to ambient motion artifacts, such as the sudden movement during breathing seen in Figure 12.3. By tracking the change in this filtered mean and its inflection points over short windows, we are able to detect the onset, duration, and magnitude of such shifts. This approach improved ambient motion classification from ≈86% using an Artificial Neural Network to 95% using LOWESS filtering with a slope threshold, and eliminated the need to train the neural network.

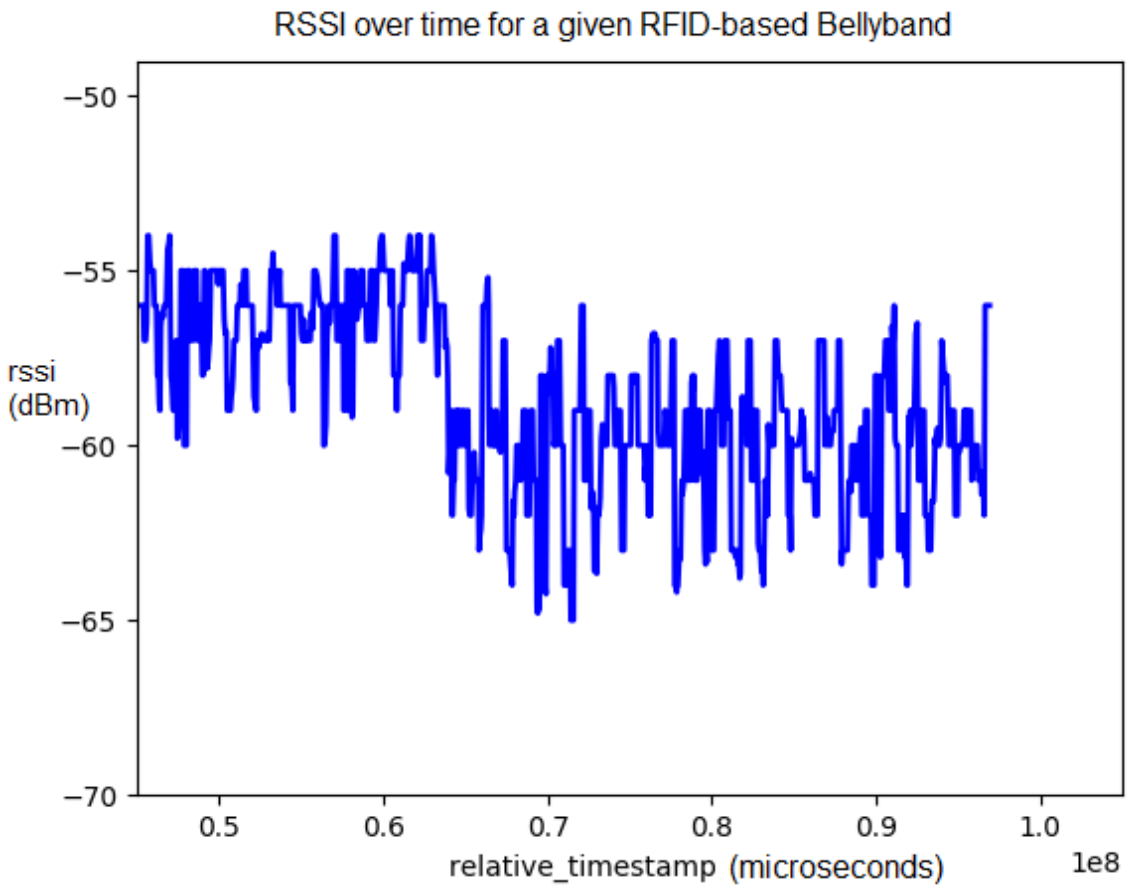

Figure 12.3: Sample respiratory data from a programmable mannequin indicating a sudden ambient movement after 60 seconds

12.2 Activity State Classification

Classification of respiratory activity is computed using the features extracted in Chapter 5, according to the algorithms described in Chapter 6, in order to determine the instantaneous band state (*i.e.*, stretching or not), as well as to determine the frequency with which the band state changes and the duration of classified stretching activity. As such, our classification detects cessation of stretching activity for prolonged periods of time (*i.e.*, apnea), or the resumption of stretching activity (*i.e.*, a uterine contraction, or limb motion), or for brief changes in band state (*i.e.*, the pause between respiratory inspiration and expiration, or the duration of the breath between inspiration and expiration). The latter is fused with other band measurements to estimate band stretching rate (*i.e.*, respiratory rate) in Section 12.3.

For classification of prolonged periods within a specific state, such as apnea, we perform classification of short-time windows according to the approach in Chapter 6. SVM classification using synthetically generated anomaly (non-breathing) class data and Platt Scaling (see Section 6.3) resulted in a 0.94 AUC precision-recall curve and ROC curve, as shown in Figure 12.4, an improvement from ≈0.70 using a One-Class SVM without Platt Scaling[16;17].

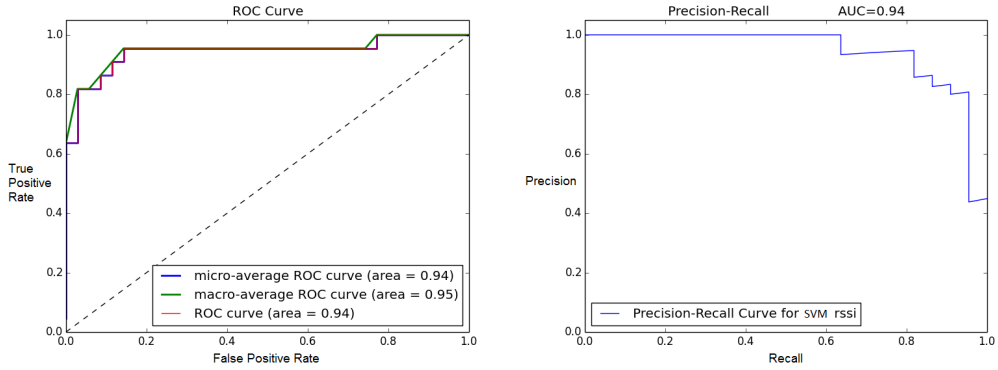

Figure 12.4: Receiver Operating Characteristic (ROC, left) and Precision-Recall curve (right) for the SVM classifier with area-under-the-curve (AUC) of 0.94 in each figure.

In addition to perceptron-based classification such as the SVM, we also utilize feature-based classification using hypothesis testing on the maximum spectral magnitude observed in a Short-Time Fourier Transform of windows of RFID data. Again, because short time windows are utilized, durations between inspiration and expiration may be classified as interruptions in respiratory activity. Apnea is defined as a reduction of respiratory activity for 10 seconds[136], so we establish a threshold of 4-10 seconds before determining that an interruption of respiratory activity has taken place[1]. Using the programmable SimBaby, respiratory cessation was detected with no false positives nor false negatives after 4 seconds, when iterating between respiratory activity at a known rate and at a rate of 0 (cessation), as shown in the programmable rate plots in Figure 12.9. This approach was also used on RFID data collected from an infant subject, in which each of two apnea periods were classified.

An Analysis of Variance (ANOVA) test was utilized to quantify the identification of respiratory

[1] A shorter alerting threshold, *i.e.*, 4 seconds, is used for experimental purposes.

artifacts. It was observed that the SimBaby mannequin completes an inspiration or expiration in ≈0.5 seconds, with an average of 0.56 seconds per event. By correlating the first order difference of the RFID received backscatter power ζ and the tag velocity v[25], we observed an RMS error from the 0.56 second ground truth of 0.57 seconds, an improvement over using ζ alone ($p = 0.0001$). Although this improvement is attributable to the tag velocity, using ζ also improved significantly over using the tag velocity for counting the number of respirations in a given period, which improved upon respiratory rate estimation (discussed in Section 12.3). Tag velocity is used primarily in measuring the duration of a classified band stretch action, with correlation utilized for denoising purposes.

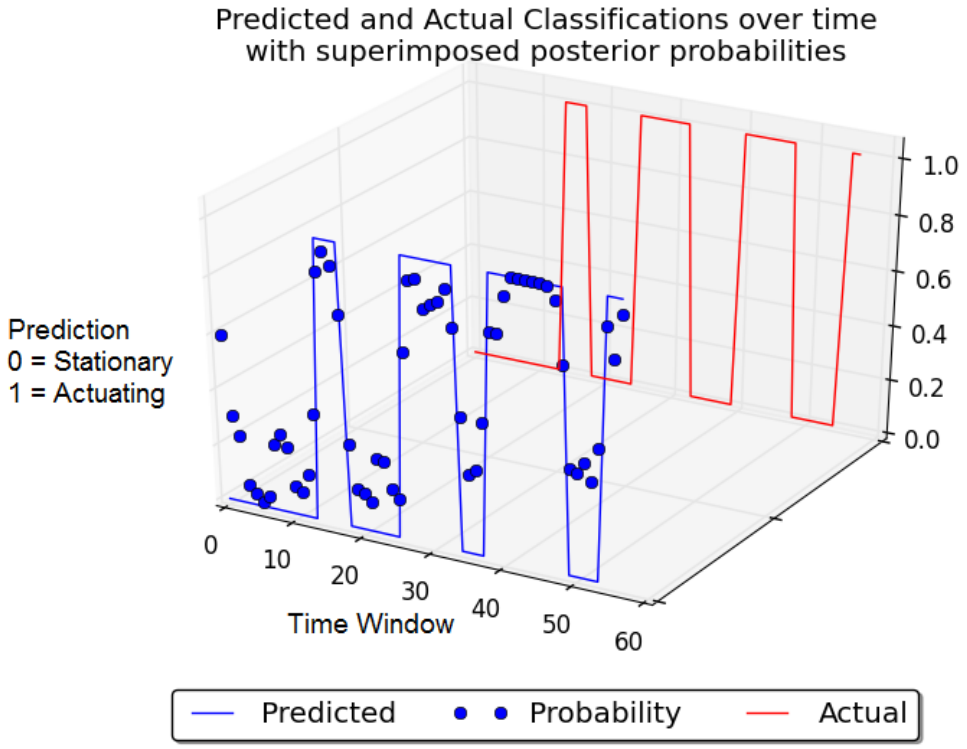

Figure 12.5: Predictive classification between breathing and non-breathing classes using the posterior probability trend with a programmable mannequin SimBaby alternating between breathing at an arbitrary rate and apnea (true class depicted in red); using probabilistic classification, changes in posterior probability are correctly classified before the linear perceptron alone would have

The hypothesis test classifier was tested with an adult human subject breathing at various rates

as shown in with brief cessations at 30 and 90 seconds. Although the RSSI dynamic range was only 1-3 dBm of RSSI, the classifier alerted all apnea conditions (one was reported rather late), with a false positive in each trial (except for one) at the transition point between respiration and cessation lasting ≈1 and ≈3 seconds each, and no false negatives. These results are summarized in Table 12.1.

Table 12.1: Classification alerting times for respiratory cessation of an adult human subject

	First Apnea (t=30 sec) Detection Time	**Second Apnea (t=90 sec) Detection Time**	**False Positive(s)**
Rate 10	37 sec	91 sec (brief), 97 sec	65-68 sec
Rate 15	45 sec	103 sec	71 sec
Rate 20	37 sec	98 sec	64 sec
Rate 30	34 sec	98 sec	None
Rate Varied	39 sec	113 sec	29 sec

As discussed in Section 12.1, the MCMC simulation data distributions reveal shifts in the RFID data properties that can indicate cessation of band stretching activity or the need for retraining baseline conditions. Either is an event that must be alerted. To determine these conditions, a vote among binary classifications summarized in Section 12.1 is taken. These measure a shift in the low MCMC distribution (noise) mean or standard deviation by three times the standard deviation of the baseline mean, a shift in the number of data points classified into the noise distribution, a change in the Hidden Markov Model classification using the Doppler, ζ, and tag velocity, and a reduction in the t-score confidence of the hypothesis test classifier, or a shift in the high MCMC distribution (stretching activity) mean. If any of these measurements alert for a period of 4 consecutive seconds, an alert is made. In the event of disagreement among the measurements, a majority vote is taken. The result of these classifications is shown in the plots of Figures 12.6 through 12.8, and represent the same data as that collected from the adult human subject for the hypothesis test classifier summarized in Table 12.1. Two of the false positives are eliminated via MCMC simulation and voting over the t-test approach alone; particularly, the t-test t-score temporal trend suggested alertable

conditions when the binary classifier alone rejected the hypothesis when $p > 0.05$.

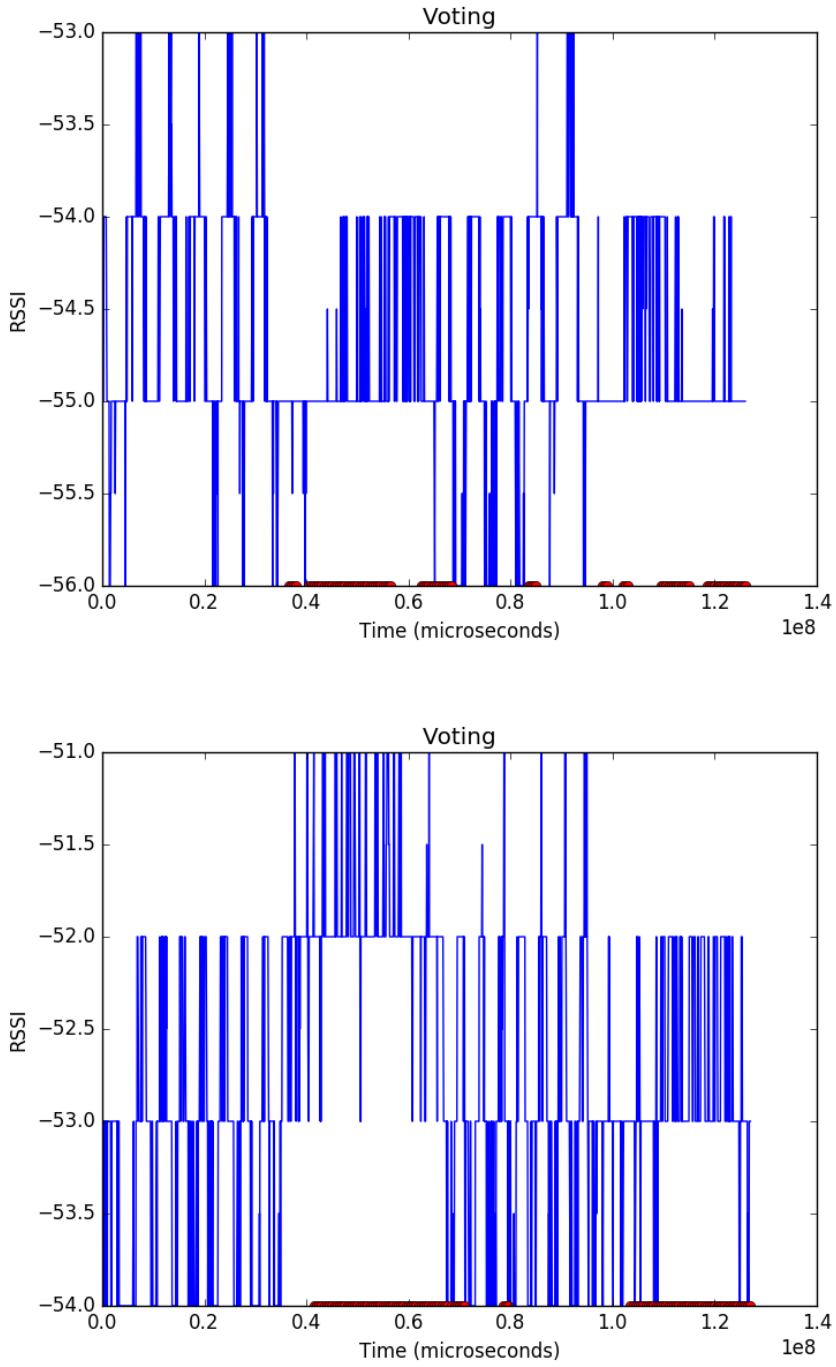

Figure 12.6: Classification using voting on multiple MCMC simulator measurements for a human trial at a respiratory rate of 10 per minute (top) and 15 per minute (bottom) with brief cessations: cessation periods are classified along the bottom of the plot

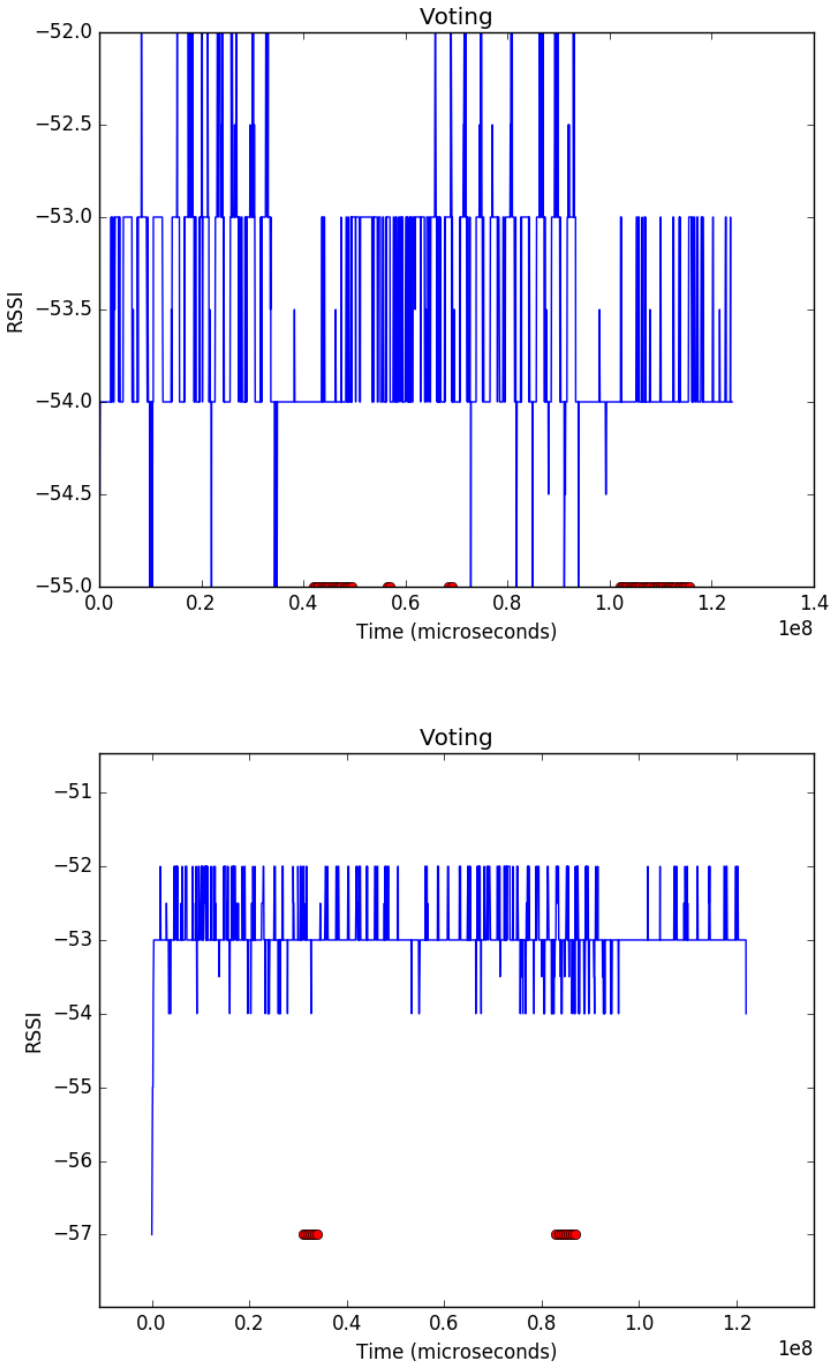

Figure 12.7: Classification using voting on multiple MCMC simulator measurements for a human trial at a respiratory rate of 20 per minute (top) and 30 per minute (bottom) with brief cessations: cessation periods are classified along the bottom of the plot

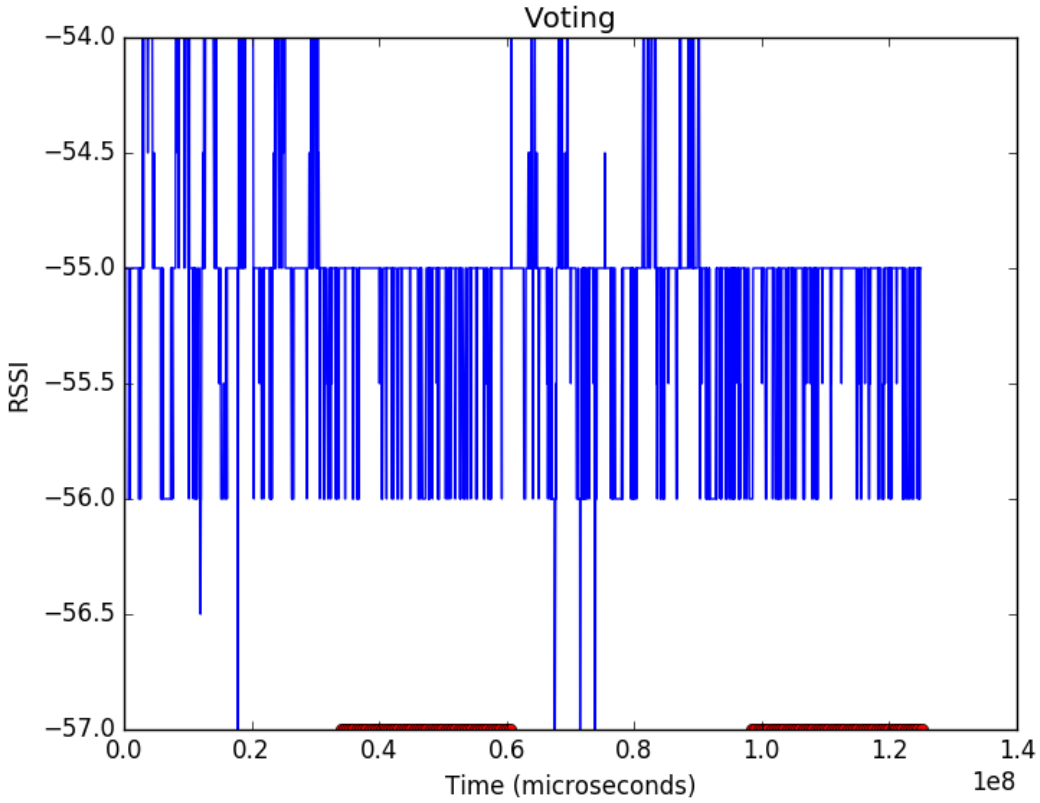

Figure 12.8: Classification using voting on multiple MCMC simulator measurements for a human trial at a varying respiratory rate with brief cessations: cessation periods are classified along the bottom of the plot

12.3 Activity Rate Estimation

Using the programmable mannequin SimBaby and the cessation t-test hypothesis testing described in Section 12.2, respiratory rates were calculated from the number of oscillations between classification states over a sliding window period of time. This resulted in an RMS error of ≈9 respirations per minute, and the calculated rates over time are illustrated in the plots of Figure 12.9[17].

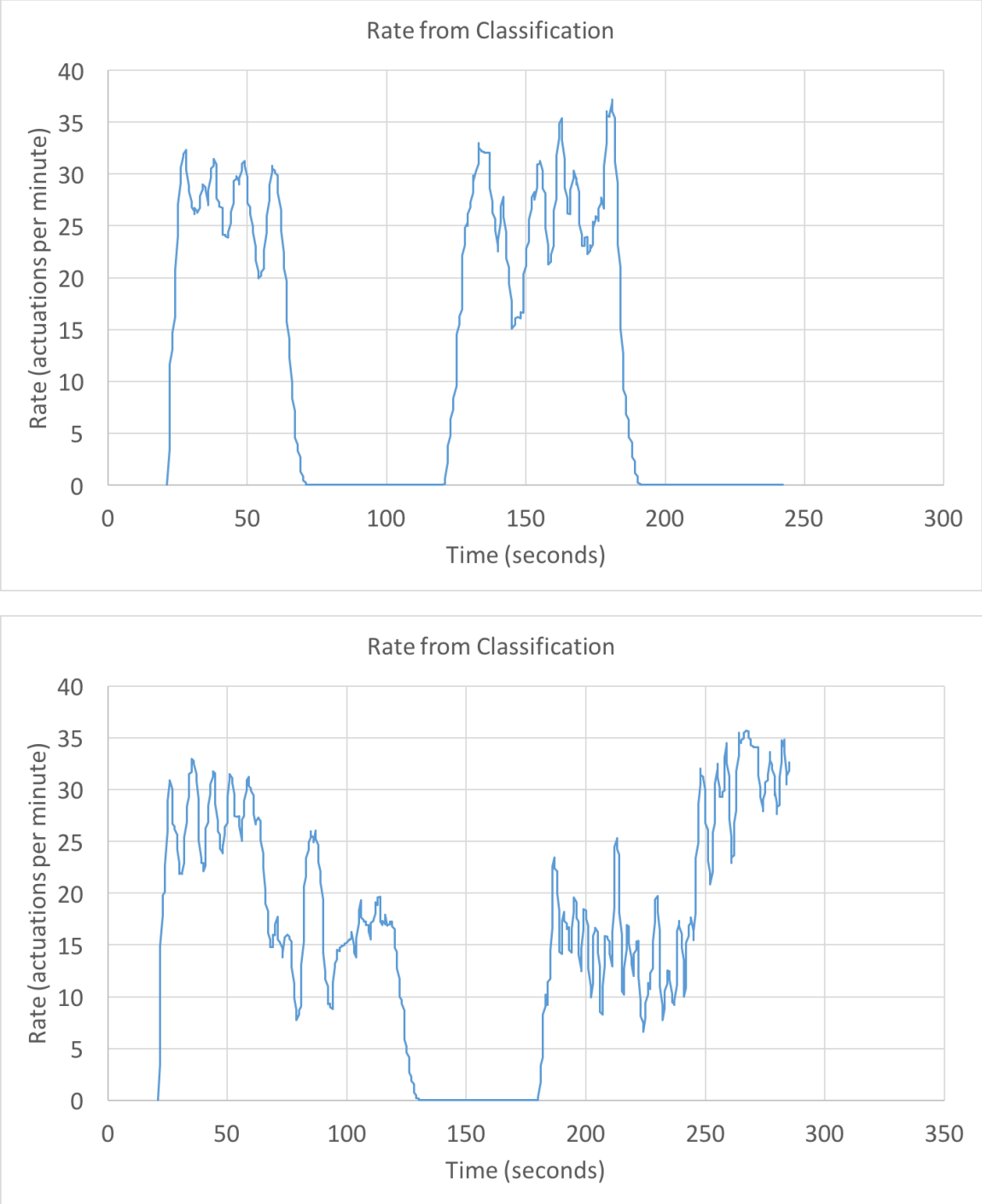

Figure 12.9: Rate estimation over time for two experimental runs (top: stretching at 31, 0, 31, 0 per minute for 1 minute each, and bottom: stretching at 31, 15, 0, 15, 31 per minute for 1 minute each), computed as the number of actuation classifications made in the past $k = 6$ seconds extrapolated to a one-minute rate. Note that the plots begin after 20 seconds because the first 20 seconds of data are reserved for training the classifier used in computing the rates.

Spectral centroid analysis and discrete peak detection approaches described in Chapter 7 were

fused together via a weighted average by the inverse of their respective recent variances, resulting in rate estimates as shown in Tables 12.2 and 12.3 for the same SimBaby data sets (also with $k = 6$ second rate estimate windows)[17].

Table 12.2: Estimated and actual rates from the FFT and Slope "tangent" sensors aggregated over each 30 seconds, in which the respiration rate oscillates between 31 and 0.

	Estimated Rate	**FFT Rate**	**Slope Rate**	**True Rate**
0-30 sec	26.6	23.1	31.4	31
30-60 sec	29.4	30.7	27.5	31
60-90 sec	2.4	3.2	0.6	0.0
90-120 sec	0.0	0.0	0.0	0.0
120-150 sec	24.7	34.7	17.5	31
150-180 sec	25.8	28.1	24.2	31
180-210 sec	1.1	0.6	0.4	0.0
210-240 sec	0.0	0.7	0.0	0.0

Table 12.3: Actuation rates aggregated over each 30 seconds, in which the respiration rate oscillates between 31, 15, and 0.

	Estimated Rate	**FFT Rate**	**Slope Rate**	**True Rate**
0-30 sec	28.6	29.6	29.5	31
30-60 sec	29.9	30.6	28.6	31
60-90 sec	18.9	20.8	18.1	15
90-120 sec	18.9	22.4	16	15
120-150 sec	0.3	0.0	0.0	0.0
150-180 sec	0.0	0.0	0.0	0.0
180-210 sec	15.9	27.5	9.2	15
210-240 sec	17.2	27.1	11.8	15
240-270 sec	22.6	21.8	27.6	31
270-300 sec	25.2	22.6	25.2	31

Spectral analysis was improved by reducing spectral leakage by FFT interpolation in the short-term, and by long-term tracking the observed spectral frequency over time using a Hidden Markov Model (HMM), as described in Chapter 7.

Indeed, the use of RFID power ζ improves upon respiratory rate estimation using the tag velocity alone, as an overall RMS error of 1.7 respirations per minute compared to 4.2 respirations per minute with the velocity alone; this was a significant improvement (ANOVA test $p = 0.001$)[25].

On another adult human subject, we observed an RMS error of 11 for the Fast Fourier Transform (FFT) alone, 78 with Quinn interpolation, 6 with HMM tracking, and 7 with Quinn interpolation and HMM tracking[25]. Although Quinn interpolation was not informative in this study, we have found it to be helpful in other measurements at best, and not destructive in the worst case due to multisensor fusion via a Gaussian Mixture Model (GMM) as discussed in Chapter 9; moreover, we observed that the interpolation errors introduced here were corrected via HMM tracking. GMM fusion generally selected the best performing measure by RMS and by variance[25]. This selection enables the measurement of RFID physical properties using multiple approaches that may be hindered by certain environmental conditions (for example, spectral analysis being perturbed by ambient motion), because the GMM tracks the measurements into a fused estimate, which is subsequently tracked by a Kalman filter. An example fusion result is shown in Figure 12.10.

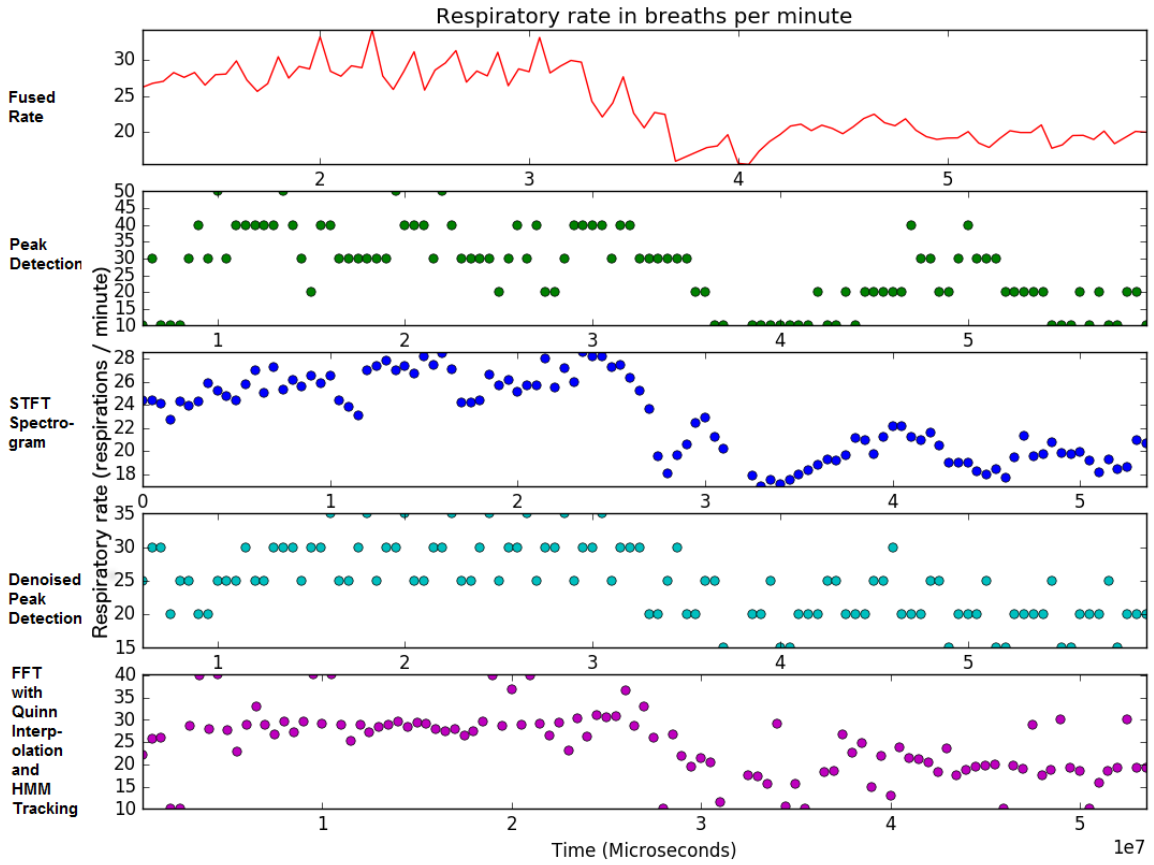

Figure 12.10: Individual observed measures and their fusion over time for a human subject breathing at a rate of approximately 30 for 30 seconds, and at a rate of approximately 15 for 30 seconds

Chapter 13: Conclusion

In this effort, we developed a software system to securely collect and store RFID physical properties in real-time for biomedical processing. Through a generalizable RESTful web services interface, this framework has supported all of the experimentation in this thesis as well as other RFID data processing projects. Our framework overcame challenges throughout the wearable IoT sensor networks application stack, including communications latency, known-ciphertext vulnerabilities, and integration of heterogeneous sensor/interrogator pairs.

Using this framework, we monitored non-intrusive, passive, battery-free wearable sensors that reflect wireless signals without requiring the user's attention. We detected the changes in the physical properties of these wireless reflections as the body moves or stretches while wearing the garment, to indicate biofeedback such as respiratory events like sleep apnea, uterine activity of a mother during labor and delivery, or movement of the limbs to prevent the onset of a blood clot. Because wireless signal properties do not translate directly into physiological activity due to a number of factors including noise and mechanical artifacts, we performed unsupervised machine learning to enable ubiquitous and non-stationary physiological monitoring via these physical properties of an ambient wireless signal. We enabled this unsupervised learning approach by dynamically generating synthetic anomaly data according to a distribution constructed from aggregate environmental observations and noise.

In addition to environmental noise, FCC standard multifrequency spectrum utilization, and mechanical interference from moving objects in the field, RFID data collected from the Bellyband is subject to variations in wearer fit. Further, stretching the band is intended to degrade the impedance match to the 900 MHz RFID signal; however, the interrogation frequency changes over time, and so the antenna may not be tuned exactly to the interrogation frequency at a specific time. Further, wearer fit may change the optimal frequency, such that band stretch actually improves the impedance match briefly before degrading it as expected, as the antenna tunes to a frequency closer

to the interrogation frequency in use at that moment. Respiratory artifacts may also move the tag closer to the interrogator in space, resulting in a strengthening of the backscatter response concurrent to the signal degredation due to stretching. It is difficult to obtain a consistent backscatter signal pattern for a given individual and environment, in practice. Nevertheless, by denoising and correcting measurements of individual RFID physical properties, and by fusing those measure estimates using Expectation Maximization on a Gaussian Mixture Model into unified classifications and estimates of band state, we classified human activity state from powerless, wireless, wearable RFID-based smart garment devices. Classification was done in an unsupervised or semi-unsupervised manner, and a novel approach to simulating multi-class training data enabled this semi-unsupervised classification with a high degree of precision and recall accuracy. The development of these algorithms culminated in a successful laboratory human subjects trial, and clinical trials of infants and pregnant women are in preparation. We conducted and evaluated live human classification like apnea events, uterine contractions, and other strain gauge movements using these features, estimated the rate of actuation and state change, and, finally, made adaptive predictions about the onset of the next event based upon this information.

These novel algorithms support a range of biomedical projects beyond those considered directly in this work. Further, the RFID data collection software framework has been demonstrated to be useful in multiple RF studies, including those on human subjects.

13.1 Future Work

As future work, we are expanding the multisensor fusion model to multiple interrogators to enable wearer "handoffs" from interrogator to interrogator as they move freely about the area. We are studying multisensor fusion of multiple worn RFID tags for signal denoising in order to enable mobile interrogation of an ambulatory subject outside of this hospital setting.

Additionally, Vora[12;148] describes a passive wearable solution for monitoring heart rate activity in human subjects. This circuit works by disabling the RFID backscatter response for a brief period when an ECG R-wave spike is observed. The intervals between successive RFID response "gaps" are used to estimate the subject heart rate. An example deployment of the RFID heart monitor using a

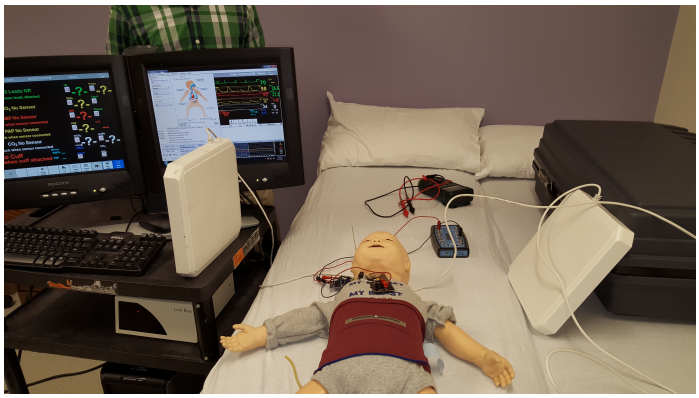

Figure 13.1: An example physical deployment according to Figure 4.3 in which the Bellyband is actuated by the SimBaby simulator while an RFID-based heart monitor is simultaneously interrogated by the client driver described in Figure 4.8

simulated EKG signal, and RFID Bellyband respiratory monitor deployed on a SimBaby exhibiting simulated respiratory activity, is shown in Figure 13.1. The RFID software and data framework of Section 4.2 was used to collect RFID-based ECG data for analysis.

A correlation exists between heart beat interval and respiratory activity due to respiratory sinus arrhythmia (RSA)[149]. Respiratory activity can be inferred by modulating according to R-wave and T-wave activity[149] activity; however, the low-power RFID-based heart monitor provides information on subject R-wave activity. During inhalation, the interval between successive R-wave peaks decreases, and subsequently rises during expiration. It is known that respiratory analysis can be performed using R-wave periods only, if noise reduction on the period data is performed[150].

Because the time-series data provided by the RFID-based heart monitor is different from a traditional ECG trace, simulated heart and respiratory data were simultaneously collected from a mannequin SimBaby as shown in Figure 13.1. Significant gaps in the RFID response from the heart RFID tag were used to identify peaks in the R-wave trace of the unknown ECG, and the R-R wave peaks formed a modulating time-series from which respiration activity is inferred.

R-R intervals are inferred from RFID "outages" by measuring the duration between RFID interrogations. There is a natural gap between interrogations, typically 0.05 seconds or lower depending upon the interrogation rate. Because we assume at least an interrogation rate of 20 Hz, an interroga-

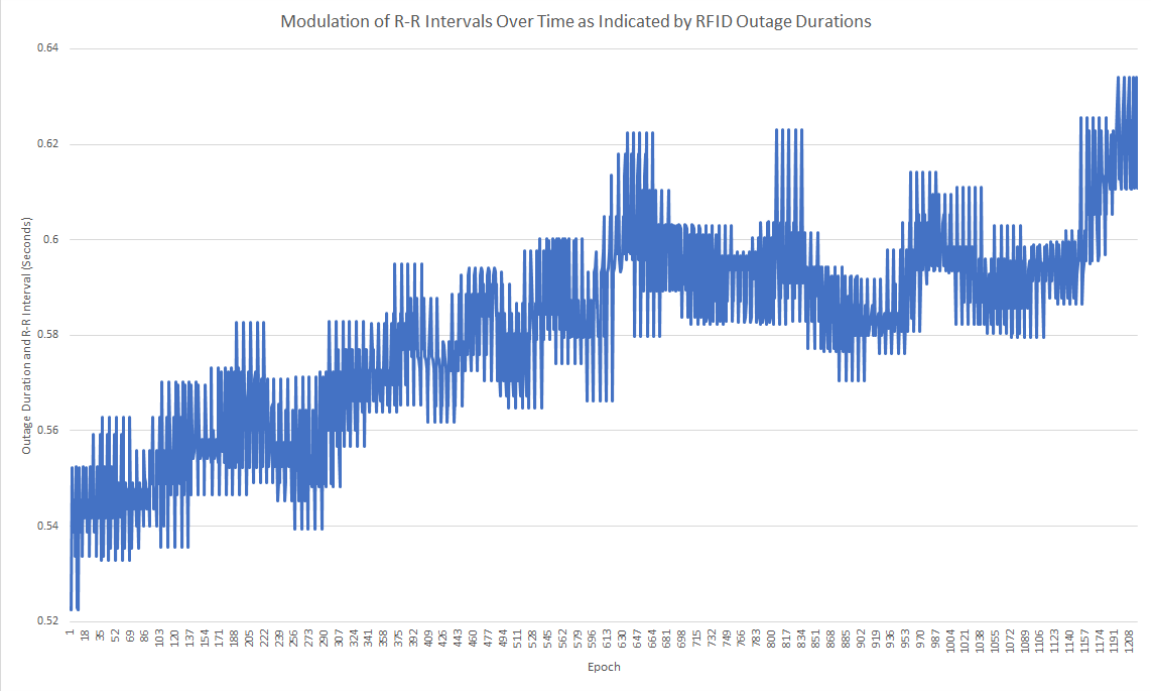

Figure 13.2: An example modulation of R-R intervals as inferred from RFID outages during heart monitoring over a 145 second period, correlating to respiratory activity

tion period of 0.05 seconds is reasonable. Therefore, the RFID tag outage time when a heart R-wave peak is detected is set to be at least 0.1 seconds. Outages of at least this duration are labeled in the data, and the period between these labeled data elements is recorded as the R-R interval. The resulting R-R interval is depicted in Figure 13.2.

As future work, we will collect simultaneous R-R heart data and respiratory data using RFID wearable devices, as well as a ground-truth respiratory monitor, to infer respiratory detection through an RFID heart sensor. We have obtained IRB approval for this study on human subjects and work is ongoing at the time of this writing.

Bibliography

[1] Gordon H. Guyatt, Elie A. Akl, Mark Crowther, David D. Gutterman, Holger J. Schuünemann, for the American College of Chest Physicians Antithrombotic Therapy Panel, and Prevention of Thrombosis. Executive Summary: Antithrombotic Therapy and Prevention of Thrombosis, 9th ed: American College of Chest Physicians Evidence-Based Clinical Practice Guidelines. *Chest*, 141(2 Suppl):7S–47S, Feb 2012. ISSN 0012-3692. doi: 10.1378/chest.1412S3. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3278060/`. 22315257[pmid].

[2] Qazi Iqbal, Mir M. Younus, Asif Ahmed, Ikhlas Ahmad, Javed Iqbal, Bashir A. Charoo, and S. Wajid Ali. Neonatal Mechanical Ventilation: Indications and Outcome. *Indian J Crit Care Med*, 19(9):523–527, Sep 2015. ISSN 0972-5229. doi: 10.4103/0972-5229.164800. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4578196/`. 26430338[pmid].

[3] Tammy Y. Euliano, Minh Tam Nguyen, Shalom Darmanjian, Susan P. McGorray, Neil Euliano, Allison Onkala, and Anthony R. Gregg. Monitoring Uterine Activity during Labor: A Comparison of Three Methods. *Am J Obstet Gynecol*, 208(1):66.e1–66.e6, Jan 2013. ISSN 0002-9378. doi: 10.1016/j.ajog.2012.10.873. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3529844/`. 23122926[pmid].

[4] T B Waggener, I D Frantz, A R Stark, and R E Kronauer. Oscillatory Breathing Patterns Leading to Apneic Spells in Infants. *Journal of applied physiology: respiratory, environmental and exercise physiology*, 52(5):1288–95, 1982. ISSN 0161-7567. URL `http://www.ncbi.nlm.nih.gov/pubmed/7096153`.

[5] Xavier Navarro, Fabienne Porée, Alain Beuchée, and Guy Carrault. Respiration Signal as a Promising Diagnostic Tool for Late Onset Sepsis in Premature Newborns. In *Computing in Cardiology*, pages 947 – 950, September 2010.

[6] Guang Yang, Cora Sau, Wan Lai, Joseph Cichon, and Wei Li. An Evidence-based Quality Improvement Project to Improve Deep Vein Thrombosis Prophylaxis with Mechanical Modalities in the Surgical Intensive Care Unit. 344(6188):1173–1178, 2015. ISSN 1527-5418. doi: 10.1126/science.1249098.Sleep.

[7] Anthony J. Comerota, Mira L. Katz, and John V. White. Why does Prophylaxis with External Pneumatic Compression for Deep Vein thrombosis Fail? *The American Journal of Surgery*, 164(3):265–268, Sep 1992. ISSN 0002-9610. doi: 10.1016/S0002-9610(05)81083-9. URL `https://doi.org/10.1016/S0002-9610(05)81083-9`.

[8] M J G Dunn, C L Gwinnutt, and A J Gray. Critical Care in the Emergency Department: Patient Transfer. *Emergency Medicine Journal*, 24(1):40–44, 2007. ISSN 1472-0205. doi: 10.1136/emj.2006.042044. URL `http://emj.bmj.com/cgi/doi/10.1136/emj.2006.042044`.

[9] Ji Young Hwang, Ki Young Kim, and Kang Hyun Lee. Factors That Influence the Acceptance of Telemetry by Emergency Medical Technicians in Ambulances: An Application of the Extended Technology Acceptance Model. *Telemed J E Health*, 20(12):1127–1134, Dec 2014. ISSN 1530-5627. doi: 10.1089/tmj.2013.0345[PII]. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4270140/`. 25531202[pmid].

[10] W. Mongan, E. Anday, G. Dion, A. Fontecchio, K. Joyce, T. Kurzweg, Y. Liu, O. Montgomery, I. Rasheed, C. Sahin, S. Vora, and K. Dandekar. A Multi-Disciplinary Framework for Continuous Biomedical Monitoring Using Low-Power Passive RFID-Based Wireless Wearable Sensors. In *2016 IEEE International Conference on Smart Computing, SMARTCOMP 2016*, 2016. ISBN 9781509008988. doi: 10.1109/SMARTCOMP.2016.7501674.

[11] D. Patron, W. Mongan, T. P. Kurzweg, A. Fontecchio, G. Dion, E. K. Anday, and K. R. Dandekar. On the Use of Knitted Antennas and Inductively Coupled RFID Tags for Wearable Applications. *IEEE Transactions on Biomedical Circuits and Systems*, 10(6):1047–1057, Dec 2016. ISSN 1932-4545. doi: 10.1109/TBCAS.2016.2518871.

[12] Shrenik Vora, Kapil Dandekar, and Timothy Kurzweg. Passive RFID Tag Based Heart Rate Monitoring from an ECG Signal. In *37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4403–4406. PubMed, 2015.

[13] Xiao Fan, InChan Song, KyungHi Chang, Dong-Beom Shin, Heyung-Sub Lee, Cheol-Sig Pyo, and Jong-Suk Chae. Gen2-Based Tag Anti-collision Algorithms Using Chebyshev's Inequality and Adjustable Frame Size. *ERTI Journal*, 30(5):653–662, 10 2008. doi: http://dx.doi.org/10.4218/etrij.08.1308.0098.

[14] Wen-Tzu Chen and Wen-Bin Kao. A Novel Q-algorithm for EPC Global Class-1 Generation-2 Anti-collision Protocol. *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, 5(6):61 – 64, 2011. ISSN 1307-6892. URL `http://iastem.com/Publications?p=54`.

[15] GS1. EPC Tag Data Standard (TDS). September 2017. URL `https://www.gs1.org/sites/default/files/docs/epc/GS1_EPC_TDS_i1_11.pdf`.

[16] W. Mongan, K. Dandekar, G. Dion, T. Kurzweg, and A. Fontecchio. Statistical Analytics of Wearable Passive RFID-based Biomedical Textile Monitors for Real-time State Classification. In *2015 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pages 1–2, Dec 2015. doi: 10.1109/SPMB.2015.7405465.

[17] W.M. Mongan, I. Rasheed, K. Ved, A. Levitt, E. Anday, K. Dandekar, G. Dion, T. Kurzweg, and A. Fontecchio. Real-time Detection of Apnea via Signal Processing of Time-series Properties of RFID-based Smart Garments. In *2016 IEEE Signal Processing in Medicine and Biology Symposium, SPMB 2016 - Proceedings*, 2017. ISBN 9781509067138. doi: 10.1109/SPMB.2016.7846871.

[18] William M. Mongan, Ilhaan Rasheed, Khyati Ved, Shrenik Vora, Kapil Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam Fontecchio. On the Use of Radio Frequency Identification for Continuous Biomedical Monitoring. In *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation*, IoTDI '17, pages 197–202, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4966-6. doi: 10.1145/3054977.3055002. URL `http://doi.acm.org/10.1145/3054977.3055002`.

[19] Ziad Obermeyer, Jasmeet K Samra, and Sendhil Mullainathan. Individual Differences in Normal Body Temperature: Longitudinal Big Data Analysis of Patient Records. *BMJ*, 359, 2017. ISSN 0959-8138. doi: 10.1136/bmj.j5468. URL `https://www.bmj.com/content/359/bmj.j5468`.

[20] P Davies and I Maconochie. The Relationship between Body Temperature, Heart Rate and Respiratory Rate in Children. *Emergency Medicine Journal*, 26(9):641–643, 2009. ISSN 1472-0205. doi: 10.1136/emj.2008.061598. URL `https://emj.bmj.com/content/26/9/641`.

[21] Kamuran Turksoy, Colleen Monforti, Minsun Park, Garett Griffith, Laurie Quinn, and Ali Cinar. Use of Wearable Sensors and Biometric Variables in an Artificial Pancreas System. *Sensors*, 17(3), 2017. ISSN 1424-8220. doi: 10.3390/s17030532. URL `http://www.mdpi.com/1424-8220/17/3/532`.

[22] EPCglobal. Low Level Reader Protocol (LLRP) Ratified Standard, Version 1.1, October 2010. http://www.gs1.org/gsmp/kc/epcglobal/llrp/llrp_1_1-standard-20101013.pdf.

[23] S.A. Vora, W.M. Mongan, E.K. Anday, K.R. Dandekar, G. Dion, A.K. Fontecchio, and T.P. Kurzweg. On Implementing an Unconventional Infant Vital Signs Monitor with Passive RFID Tags. In *2017 IEEE International Conference on RFID, RFID 2017*, 2017. ISBN 9781509045761. doi: 10.1109/RFID.2017.7945586.

[24] Shrenik Vora, William Mongan, Kapil Dandekar, Adam Fontecchio, and Timothy Kurzweg. Wireless Heart and Respiration Monitoring for Infants using Passive RFID Tags. In *Proceedings of BHI-2016 International Conference on Biomedical and Health Informatics*, February 2016.

[25] William M. Mongan, Robert Ross, Ilhaan Rasheed, Yuqiao Liu, Khyati Ved, Endla Anday, Kapil R. Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam K. Fontecchio. Data Fusion of Single-Tag RFID Measurements for Respiratory Rate Monitoring. In *IEEE Signal Processing in Medicine and Biology (SPMB)*, pages 1–6, 2017.

[26] Daphney-Stavroula Zois, Marco Levorato, and Urbashi Mitra. Energy-Efficient, Heterogeneous Sensor Selection for Physical Activity Detection in Wireless Body Area Networks. *IEEE Transactions on Signal Processing*, 61(7):1581–1594, 2013. URL `http://dblp.uni-trier.de/db/journals/tsp/tsp61.html#ZoisLM13`.

[27] S. Park and S. Jayaraman. Smart Textile-Based Wearable Biomedical Systems: A Transition Plan for Research to Reality. *IEEE Transactions on Information Technology in Biomedicine*, 14(1):86–92, Jan 2010. ISSN 1089-7771. doi: 10.1109/TITB.2009.2025817.

[28] Sima Ajami and Fotooheh Teimouri. Features and Application of Wearable Biosensors in Medical Care. *J Res Med Sci*, 20(12):1208–1215, Dec 2015. ISSN 1735-1995. doi: 10.4103/1735-1995.172991. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4766830/`. 26958058[pmid].

[29] A. Syberfeldt, O. Danielsson, and P. Gustavsson. Augmented Reality Smart Glasses in the Smart Factory: Product Evaluation Guidelines and Review of Available Products. *IEEE Access*, 5:9118–9130, 2017. ISSN 2169-3536. doi: 10.1109/ACCESS.2017.2703952.

[30] Mostafa Haghi, Kerstin Thurow, and Regina Stoll. Wearable Devices in Medical Internet of Things: Scientific Research and Commercially Available Devices. *Healthc Inform Res*, 23(1):4–15, Jan 2017. ISSN 2093-3681. doi: 10.4258/hir.2017.23.1.4. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC5334130/`. 28261526[pmid].

[31] John G. Webster, editor. *The Physiological Measurement Handbook*. CRC Press, 2015. ISBN 978-1-4398-0847-4.

[32] *MAGICSTRAP Technical Data Sheet*. Murata, 11 2012.

[33] *Impinj Monza X-2K Dura Datasheet*. Impinj, 3 2014. Rev. 1.51.

[34] Impinj. Impinj Speedway Revolution, . http://www.impinj.com/products/readers/speedway-revolution/.

[35] *Circular Polarity RFID Panel Antenna*. RFMAX.

[36] Rodolfo Giometti. LLRPyC LLRP Python Protocol Client Implementation. http://wiki.enneenne.com/index.php/LLRPyC.

[37] Fosstrak. Fosstrak: Open Source RFID Software Platform. August 2018. URL `http://fosstrak.github.io/`.

[38] Laerdal. Laerdal SimBaby and Linkbox. http://www.laerdal.com/us/item/245-02001.

[39] D. Arnitz, U. Muehlmann, and K. Witrisal. Characterization and Modeling of UHF RFID Channels for Ranging and Localization. *IEEE Transactions on Antennas and Propagation*, 60(5):2491–2501, May 2012. ISSN 0018-926X. doi: 10.1109/TAP.2012.2189705.

[40] EPCGlobal. EPC Radio-Frequency Identity Protocols Generation-2 UHF RFID. April 2015. URL https://www.gs1.org/sites/default/files/docs/epc/Gen2_Protocol_Standard.pdf.

[41] U.S. Government Publishing Office. Electronic Code of Federal Regulations, Title 47, Chapter I, Subchapter A, Part 15.247, 2018. http://www.ecfr.gov/cgi-bin/text-idx?node=pt47.1.15&rgn=div5#se47.1.15_1247.

[42] Carole A. Winterhalter, Justyna Teverovsky, Patricia Wilson, Jeremiah Slade, Wendy Horowitz, Edward Tierney, and Vikram Sharma. Development of Electronic Textiles to Support Networks, Communications, and Medical Applications in Future U.S. Military Protective Clothing Systems. *IEEE Transactions on Information Technology in Biomedicine*, 9(3):402–406, 2005. ISSN 10897771. doi: 10.1109/TITB.2005.854508.

[43] Ling Xu and Joshua Le Wei Li. A Dualband Microstrip Antenna for Wearable Application. *2012 10th International Symposium on Antennas, Propagation and EM Theory, ISAPE 2012*, pages 109–112, 2012. doi: 10.1109/ISAPE.2012.6408720.

[44] Matteo Stoppa and Alessandro Chiolerio. Wearable Electronics and Smart Textiles: A Critical Review. *Sensors (Switzerland)*, 14(7):11957–11992, 2014. ISSN 14248220. doi: 10.3390/s140711957.

[45] Ivo Locher, Maciej Klemm, Tünde Kirstein, and Gerhard Tröster. Design and Characterization of Purely Textile Patch Antennas. *IEEE Transactions on Advanced Packaging*, 29(4):777–788, 2006. ISSN 15213323. doi: 10.1109/TADVP.2006.884780.

[46] Y. Liu, A. Levitt, C. Kara, C. Sahin, G. Dion, and K. R. Dandekar. An Improved Design of Wearable Strain Sensor Based on Knitted RFID Technology. In *2016 IEEE Conference on Antenna Measurements Applications (CAMA)*, pages 1–4, Oct 2016. doi: 10.1109/CAMA.2016.7815769.

[47] Marcel Mlynczak, Ewa Migacz, Maciej Migacz, and Wojciech Kukwa. Detecting Breathing and Snoring Episodes using a Wireless Tracheal Sensor - a Feasibility Study. *IEEE Journal of Biomedical and Health Informatics*, 21(6):1–1, 2016. ISSN 2168-2194. doi: 10.1109/JBHI.2016.2632976. URL http://ieeexplore.ieee.org/document/7762101/.

[48] Philippe Guay, Stepan Gorgutsa, Sophie Larochelle, and Younes Messaddeq. Wearable Contactless Respiration Sensor Based on Multi-material Fibers Integrated into Textile. *Sensors (Switzerland)*, 17(5), 2017. ISSN 14248220. doi: 10.3390/s17051050.

[49] Ruth Ravichandran, Elliot Saba, Ke-yu Chen, Mayank Goel, Sidhant Gupta, and Shwetak N Patel. WiBreathe : Estimating Respiration Rate Using Wireless Signals in Natural Settings in the Home. 16:131–139, 2015.

[50] Chen Chen, Student Senior Member, Yi Han, Yan Chen, Student Senior Member, Hung-quoc Lai, Feng Zhang, Student Senior Member, Beibei Wang, Student Senior Member, and K J Ray Liu. TR-BREATH : Time-Reversal Breathing Rate Estimation and Detection. 9294(c):1–14, 2017. doi: 10.1109/TBME.2017.2699422.

[51] Fadel Adib, Hongzi Mao, Zachary Kabelac, Dina Katabi, and Robert C Miller. Smart Homes that Monitor Breathing and Heart Rate. *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*, pages 837–846, 2015. doi: 10.1145/2702123.2702200. URL http://dl.acm.org/citation.cfm?doid=2702123.2702200.

[52] Laura Mason. Signal Processing Methods for Non-Invasive Respiration Monitoring. *Philosophy*, page 175, 2002. URL http://www.ibme.ox.ac.uk/bsp/publications/pdf-files/laura-thesis.pdf.

[53] L. Tarassenko, N. Townsend, G. Clifford, L. Mason, J. Burton, and J. Price. Medical Signal Processing using the Software Monitor. In *A DERA/IEE Workshop on Intelligent Sensor Processing (Ref. No. 2001/050)*, pages 3/1–3/4, Feb 2001. doi: 10.1049/ic:20010098.

[54] Han Ding, Min Xi, Zhe Li, Jizhong Zhao, Jinsong Han, Wei Xi, Kun Zhao, and Zhiping Jiang. Device-Free Object Tracking Using Passive Tags. *International Journal of Distributed Sensor Networks*, 2015:1–3, 2014. ISSN 15501477. doi: 10.1155/2015/870204. URL `http://link.springer.com/10.1007/978-3-319-12646-3`.

[55] Jinsong Han, Han Ding, Chen Qian, Dan Ma, Wei Xi, Zhi Wang, Zhiping Jiang, and Longfei Shangguan. CBID: A Customer Behavior Identification System Using Passive Tags. *Proceedings - International Conference on Network Protocols, ICNP*, pages 47–58, 2014. ISSN 10921648. doi: 10.1109/ICNP.2014.26.

[56] C. Zhou and J. D. Griffin. Accurate Phase-Based Ranging Measurements for Backscatter RFID Tags. *IEEE Antennas and Wireless Propagation Letters*, 11:152–155, 2012. ISSN 1536-1225. doi: 10.1109/LAWP.2012.2186110.

[57] Xiaonan Hui and Edwin C. Kan. Monitoring Vital Signs over Multiplexed Radio by Near-field Coherent Sensing. *Nature Electronics*, 2017. ISSN 2520-1131. doi: 10.1038/s41928-017-0001-0. URL `http://www.nature.com/articles/s41928-017-0001-0`.

[58] Reza S. Dilmaghani, Hossein Bobarshad, M. Ghavami, Sabrieh Choobkar, and Charles Wolfe. Wireless Sensor Networks for Monitoring Physiological Signals of Multiple Patients. *IEEE Transactions on Biomedical Circuits and Systems*, 5(4):347–356, 2011. ISSN 19324545. doi: 10.1109/TBCAS.2011.2114661.

[59] Tia Gao, Christopher Pesto, Leo Selavo, Yin Chen, Jeonggil Ko, JongHyun Lim, Andreas Terzis, Andrew Watt, James Jeng, Bor-rong Rong Chen, Konrad Lorincz, and Matt Welsh. Wireless Medical Sensor Networks in Emergency Response: Implementation and Pilot Results. *2008 IEEE International Conference on Technologies for Homeland Security, HST'08*, pages 187–192, 2008. doi: 10.1109/THS.2008.4534447.

[60] Xiang Zhang, Helei Wu, and Marcin Uradzinski. Usefulness of Nonlinear Interpolation and Particle Filter in Zigbee Indoor Positioning. 63, 12 2014.

[61] Can Uysal and Tansu Filik. Contactless Respiration Rate Estimation Using MUSIC Algorithm. pages 606–610.

[62] Bin Zhou, Chao Hu, Hai Bin Wang, Ruiwen Guo, and Max Q.H. Meng. A Wireless Sensor Network for Pervasive Medical Supervision. *IEEE ICIT 2007 - 2007 IEEE International Conference on Integration Technology*, pages 740–744, 2007. doi: 10.1109/ICITECHNOLOGY.2007.4290419.

[63] Hanchuan Li, Can Ye, and Alanson P. Sample. IDSense: A Human Object Interaction Detection System Based on Passive UHF RFID. *Chi*, (c):2555–2564, 2015. doi: 10.1145/2702123.2702178. URL `http://dx.doi.org/10.1145/2702123.2702178`.

[64] Han Ding, Chen Qian, Jinsong Han, Ge Wang, Wei Xi, Kun Zhao, and Jizhong Zhao. RFIPad: Enabling Cost-Efficient and Device-Free In-air Handwriting Using Passive Tags. *Proceedings - International Conference on Distributed Computing Systems*, pages 447–457, 2017. doi: 10.1109/ICDCS.2017.141. URL `http://ieeexplore.ieee.org/document/7979990/`.

[65] Swadhin Pradhan, Eugene Chai, Karthik Sundaresan, Sampath Rangarajan, and Lili Qiu. Konark: A RFID Based System for Enhancing In-store Shopping Experience. *Proceedings of the 4th International on Workshop on Physical Analytics*, pages 19–24, 2017. doi: 10.1145/3092305.3092307. URL `http://doi.acm.org/10.1145/3092305.3092307`.

[66] R. Bhattacharyya, C. Floerkemeier, and S. Sarma. Towards Tag Antenna Based Sensing - An RFID Displacement Sensor. In *2009 IEEE International Conference on RFID*, pages 95–102, April 2009. doi: 10.1109/RFID.2009.4911195.

[67] T. Volk, S. Gorbey, M. Bhattacharyya, W. Gruenwald, B. Lemmer, L. M. Reindl, T. Stieglitz, and D. Jansen. RFID Technology for Continuous Monitoring of Physiological Signals in Small Animals. *IEEE Transactions on Biomedical Engineering*, 62(2):618–626, Feb 2015. ISSN 0018-9294. doi: 10.1109/TBME.2014.2361856.

[68] Stephen A Weis. RFID (Radio Frequency Identification): Principles and Applications. *System*, 2(3), 2007.

[69] Xuanlong Nguyen, Michael I Jordan, and Bruno Sinopoli. A Kernel-based Learning Approach to ad hoc Sensor Network Localization. *Transactions on Sensor Networks (TOSN*, 1 (1):134–152, 2005. ISSN 15504859. doi: 10.1145/1077391.1077397. URL `http://dl.acm.org/citation.cfm?id=1077397`.

[70] Kentaro Yamanoi, Kanji Tanaka, Mitsuru Hirayma, Eiji Kondo, and Yoshihiko Kimuro. Self-localization of Mobile Robots with RFID System by using Support Vector Machine. *Iros*, pages 3756–3761, 2004. doi: 10.1109/IROS.2004.1389999.

[71] Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. *Machine Learning Journal*, 20(3):273–297, 9 1995. ISSN 0885-6125. doi: 10.1023/A:1022627411411. URL `http://dx.doi.org/10.1023/A:1022627411411`.

[72] C Philipp Schloter. Wireless Symbolic Positioning using Support Vector Machines. *Electrical Engineering*, pages 1141–1146, 2006.

[73] Changzhi Wang, Zhicai Shi, and Fei Wu. Intelligent RFID indoor localization system using a Gaussian filtering based extreme learning machine. *Symmetry*, 9(3), 2017. ISSN 20738994. doi: 10.3390/sym9030030.

[74] Xinyu Li, Yanyi Zhang, Ivan Marsic, Aleksandra Sarcevic, and Randall S. Burd. Deep Learning for RFID-Based Activity Recognition. *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM - SenSys '16*, pages 164–175, 2016. doi: 10.1145/2994551.2994569. URL `http://dl.acm.org/citation.cfm?doid=2994551.2994569`.

[75] S Amendola, R Lodato, and S Manzari. RFID Technology for IoT-based Personal Healthcare in Smart Spaces. *Internet of Things*, 1(2):144–152, 2014. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6780609`.

[76] Ali Montaser and Osama Moselhi. RFID Indoor Location Identification for Construction Projects. *Automation in Construction*, 39:167 – 179, 2014. ISSN 0926-5805. doi: https://doi.org/10.1016/j.autcon.2013.06.012. URL `http://www.sciencedirect.com/science/article/pii/S092658051300109X`.

[77] Yuri Álvarez López, Mª Elena de Cos Gómez, José Lorenzo Álvarez, and Fernando Las-Heras Andrés. Evaluation of an RSS-based Indoor Location System. *Sensors and Actuators A: Physical*, 167(1):110 – 116, 2011. ISSN 0924-4247. doi: https://doi.org/10.1016/j.sna.2011.02.037. URL `http://www.sciencedirect.com/science/article/pii/S0924424711000999`.

[78] Tamer Dag and Taner Arsan. Received Signal Strength Based Least Squares Lateration Algorithm for Indoor Localization. *Computers and Electrical Engineering*, 66:114 – 126, 2018. ISSN 0045-7906. doi: https://doi.org/10.1016/j.compeleceng.2017.08.014. URL `http://www.sciencedirect.com/science/article/pii/S0045790617308509`.

[79] Martijn Truijens, Xunzhang Wang, Hans de Graaf, J. J. Liu, and Changzhi Wu. Evaluating the Performance of Absolute RSSI Positioning Algorithm-Based Microzoning and RFID in Construction Materials Tracking. 2014.

[80] W. D. Boyer. A Diplex, Doppler Phase Comparison Radar. *IEEE Transactions on Aerospace and Navigational Electronics*, ANE-10(1):27–33, March 1963. ISSN 0096-1957. doi: 10.1109/TANE.1963.4502075.

[81] Lanxin Qiu, Xiaoxuan Liang, and Zhangqin Huang. PATL: A RFID Tag Localization Based on Phased Array Antenna. *Scientific Reports*, 7:1–12, 2017. ISSN 20452322. doi: 10.1038/srep44183. URL `http://dx.doi.org/10.1038/srep44183`.

[82] Yuri Álvarez López, María Elena de Cos Gómez, and Fernando Las-Heras Andrés. A Received Signal Strength RFID-based Indoor Location System. *Sensors and Actuators A: Physical*, 255:118 – 133, 2017. ISSN 0924-4247. doi: https://doi.org/10.1016/j.sna.2017.01.007. URL `http://www.sciencedirect.com/science/article/pii/S0924424716309153`.

[83] Waleed Alsalih, Abdallah Alma'aitah, and Wadha Alkhater. RFID Localization Using Angle of Arrival Cluster Forming. *International Journal of Distributed Sensor Networks*, 10(3):269596, 2014. doi: 10.1155/2014/269596. URL `https://doi.org/10.1155/2014/269596`.

[84] Gülben Ç Ali, Burcin Becerik-Gerber, Burak Göktepe, Shuai Li, and Nan Li. Analysis of the variability of RSSI values for active RFID-based indoor applications. 37:186–210, 07 2013.

[85] S Li, N Li, Gulben Calis, Burcin Becerik-Gerber, Ph D Student, and A M Asce. Impact of Ambient Temperature, Tag/Antenna Orientation and Distance on the Performance of Radio Frequency Identification in Construction Industry. 06 2011.

[86] Clarke Robert H., Twede Diana, Tazelaar Jeffrey R., and Boyer Kenneth K. Radio Frequency Identification (RFID) Performance: the Effect of Tag Orientation and Package Contents. *Packaging Technology and Science*, 19(1):45–54. doi: 10.1002/pts.714. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/pts.714`.

[87] J. Huiting, H. Flisijn, A. B. J. Kokkeler, and G. J. M. Smit. Exploiting Phase Measurements of EPC Gen2 RFID Tags. In *2013 IEEE International Conference on RFID-Technologies and Applications (RFID-TA)*, pages 1–6, Sept 2013. doi: 10.1109/RFID-TA.2013.6694503.

[88] Lanxin Qiu, Zhangqin Huang, Shaohua Zhang, Cheng Jing, Hao Li, and Shuyao Li. Multi-frequency Phase Difference of Arrival Range Measurement: Principle, Implementation, and Evaluation. *Int. J. Distrib. Sen. Netw.*, 2015:9:9–9:9, January 2016. ISSN 1550-1329. doi: 10.1155/2015/715307. URL `https://doi.org/10.1155/2015/715307`.

[89] G. M. Moltshan, V. F. Pissarenko, and N. A. Smirnova. Some Statistical Methods of Detecting Signals in Noise. *Geophysical Journal of the Royal Astronomical Society*, 8(3):319–323, 1964. ISSN 1365246X. doi: 10.1111/j.1365-246X.1964.tb06298.x.

[90] Aravind Kailas and Mary Ann Ingram. Wireless Communications Technology in Telehealth Systems. *Proceedings of the 2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace and Electronic Systems Technology, Wireless VITAE 2009*, pages 926–930, 2009. doi: 10.1109/WIRELESSVITAE.2009.5172574.

[91] Leonardo A. Amaral, Fabiano P. Hessel, Eduardo A. Bezerra, Jernimo C. Corrêa, Oliver B. Longhi, Thiago F O Dias, Leonardo A Amaral Ã, Fabiano P. Hessel, Eduardo A. Bezerra, Oliver B. Longhi, Thiago F O Dias, Leonardo A. Amaral, Fabiano P. Hessel, Eduardo A. Bezerra, Jernimo C. Corrêa, Oliver B. Longhi, and Thiago F O Dias. eCloudRFID - A Mobile Software Framework Architecture for Pervasive RFID-based Applications. *Journal of Network and Computer Applications*, 34(3):972–979, 2011. ISSN 10848045. doi: 10.1016/j.jnca.2010.04.005.

[92] P. Romero and K. Dighe. Fast and Unsupervised Classification of Radio Frequency Data Sets Utilizing Machine Learning Algorithms. In *Proceedings of The Fourth International Conference on Data Analytics*, pages 146–152, July 2015.

[93] M. Valkama, A. Springer, and G. Hueber. Digital Signal Processing for Reducing the Effects of RF Imperfections in Radio Devices — An Overview. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pages 813–816, May 2010. doi: 10.1109/ISCAS.2010.5537444.

[94] V. Syrjala and M. Valkama. Sampling Jitter Cancellation in Direct-Sampling Radio. In *2010 IEEE Wireless Communication and Networking Conference*, pages 1–6, April 2010. doi: 10.1109/WCNC.2010.5506638.

[95] K. Youssef, L.-S. Bouchard, K. Z. Haigh, H. Krovi, J. Silovsky, and C. P. Vander Valk. Machine Learning Approach to RF Transmitter Identification. *ArXiv e-prints*, November 2017.

[96] Merima Kulin, Tarik Kazaz, Ingrid Moerman, and Eli De Poorter. End-to-end Learning from Spectrum Data: A Deep Learning approach for Wireless Signal Identification in Spectrum Monitoring applications. *CoRR*, abs/1712.03987, 2017. URL `http://arxiv.org/abs/1712.03987`.

[97] Zhuo Su, Shing Chi Cheung, and Koon Ting Chu. Investigation of Radio Link Budget for UHF RFID Systems. *Proceedings of 2010 IEEE International Conference on RFID-Technology and Applications, RFID-TA 2010*, (June):164–169, 2010. doi: 10.1109/RFID-TA.2010.5529938.

[98] Impinj. Application Note - Low Level User Data Support. `https://support.impinj.com/hc/en-us/article_attachments/200774268/SR_AN_IPJ_Speedway_Rev_Low_Level_Data_Support_20130911.pdf`, .

[99] P. V. Nikitin, R. Martinez, S. Ramamurthy, H. Leland, G. Spiess, and K. V. S. Rao. Phase Based Spatial Identification of UHF RFID Tags. In *2010 IEEE International Conference on RFID (IEEE RFID 2010)*, pages 102–109, April 2010. doi: 10.1109/RFID.2010.5467253.

[100] A. Povalac. Spatial Identification Methods and Systems for RFID Tags. 2012.

[101] Theodore S Rappaport and Claire D McGillem. UHF Fading in Factories. *IEEE Journal on Selected Areas in Communications*, 7(1):40–48, 1989.

[102] Zheng Yang, Zimu Zhou, and Yunhao Liu. From RSSI to CSI : Indoor Localization via Channel Response. *ACM Computing Surveys (CSUR)*, 46(2):25, 2013. ISSN 03600300.

[103] A. Whitney, J. Fessler, J. Parker, and N. Jacobs. Received Signal Strength Indication Signature for Passive UHF Tags. In *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 1183–1187, July 2014. doi: 10.1109/AIM.2014.6878242.

[104] Jason L. Brchan, Lianlin Zhao, Jiaqing Wu, Robert E. Williams, and Lance C. Pérez. A Real-time RFID Localization Experiment using Propagation Models. *2012 IEEE International Conference on RFID, RFID 2012*, pages 141–148, 2012. doi: 10.1109/RFID.2012.6193042.

[105] Christoph Angerer. Design and Exploration of Radio Frequency Identification Systems. 2010.

[106] R. A. Fisher. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 7(7):179–188, 1936.

[107] Sam Kachigan. *Multivariate Statistical Analysis*. Radius Press, 1991.

[108] Impinj. Impinj LTK Programmers Guide. `https://support.impinj.com/hc/en-us/article_attachments/205361988/Impinj_LTK_Programmers_Guide.pdf`. Accessed: 2016-10-08.

[109] Ben Ransford. sllurp LLRP Python Protocol Client Implementation. https://github.com/ransford/sllurp.

[110] Intermec. IP30 Handheld RFID Reader User Guide. http://epsfiles.intermec.com/eps_files/eps_man/937-000.pdf. Access: 2016-10-10.

[111] GuoChen Peng and M.F. Bocko. Non-Contact ECG Sensing Employing Gradiometer Electrodes. *Biomedical Engineering, IEEE Transactions on*, 60(1):179–183, Jan 2013. ISSN 0018-9294. doi: 10.1109/TBME.2012.2219531.

[112] Philips. Series 50 Fetal Monitors, Design Interface Protocol Specifications: Programmer's Guide. http://www.frankshospitalworkshop.com/equipment/documents/ecg/service_manuals/Philips_Series_50_-_Programmers_guide.pdf. Accessed: 2016-10-10.

[113] J. Nagle. Congestion Control in IP/TCP Internetworks. RFC 896, RFC Editor, January 1984.

[114] Frederic P. Miller, Agnes F. Vandome, and John McBrewster. *Advanced Encryption Standard*. Alpha Press, 2009. ISBN 6130268297, 9786130268299.

[115] Paul A. Harris, Robert Taylor, Robert Thielke, Jonathon Payne, Nathaniel Gonzalez, and Jose G. Conde. Research Electronic Data Capture (REDCap)-A Metadata-driven Methodology and Workflow Process for Providing Translational Research Informatics Support. *J. of Biomedical Informatics*, 42(2):377–381, April 2009. ISSN 1532-0464. doi: 10.1016/j.jbi.2008.08.010. URL http://dx.doi.org/10.1016/j.jbi.2008.08.010.

[116] Roy Thomas Fielding. *REST: Architectural Styles and the Design of Network-based Software Architectures*. Doctoral dissertation, University of California, Irvine, 2000. URL http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm.

[117] Mountain View, Johan Rydell, Mingliang Pei, and Salah Machani. TOTP: Time-Based One-Time Password Algorithm. RFC 6238, May 2011. URL https://rfc-editor.org/rfc/rfc6238.txt.

[118] Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin. *Learning from Data*. AML-Book, 2012. ISBN 1600490069, 9781600490064.

[119] Wassily Hoeffding. Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association*, 58(301):13–30, March 1963. URL http://www.jstor.org/stable/2282952?

[120] Vladimir Vapnik, Esther Levin, and Yann Le Cun. Measuring the VC-dimension of a Learning Machine. *Neural Computation*, 6:851–876, 1994.

[121] D.R. Hush and B.G. Horne. Progress in Supervised Neural Networks. *Signal Processing Magazine, IEEE*, 10:8 – 39, 1993. ISSN 1053-5888.

[122] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *J. Artif. Int. Res.*, 16(1):321–357, June 2002. ISSN 1076-9757. URL http://dl.acm.org/citation.cfm?id=1622407.1622416.

[123] Nico Görnitz, Marius Kloft, Konrad Rieck, and Ulf Brefeld. Toward Supervised Anomaly Detection. *J. Artif. Int. Res.*, 46(1):235–262, January 2013. ISSN 1076-9757. URL http://dl.acm.org/citation.cfm?id=2512538.2512545.

[124] V. Vapnik. *Statistical Learning Theory*. Wiley, September 1998.

[125] L. M. Manevitz, M. Yousef, N. Cristianini, J. Shawe-Taylor, and B. Williamson. One-Class SVMs for Document Classification. *Journal of Machine Learning Research*, 2:139–154, 2001.

[126] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[127] William S. Cleveland. LOWESS: A Program for Smoothing Scatterplots by Robust Locally Weighted Regression. *The American Statistician*, 35(1), 1981. ISSN 00031305. doi: 10.2307/2683591. URL http://dx.doi.org/10.2307/2683591.

[128] Abraham Savitzky and M. J. E. Golay. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Anal. Chem.*, 36(8):1627–1639, July 1964. doi: 10.1021/ac60214a047. URL http://dx.doi.org/10.1021/ac60214a047.

[129] Ronald W. Schafer. On the Rrequency-domain Properties of Savitzky-Golay Filters. *2011 Digital Signal Processing and Signal Processing Education Meeting, DSP/SPE 2011 - Proceedings*, pages 54–59, 2011. doi: 10.1109/DSP-SPE.2011.5739186.

[130] Nicholas Metropolis and S. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341, 1949. ISSN 01621459. URL http://www.jstor.org/stable/2280232.

[131] Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of Markov Chain Monte Carlo*. CRC press, 2011.

[132] J.B. MacQueen. Some Methods for Classification and Analysis of Multivariate Observations. *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, 1967.

[133] Philip L Witt and Peter McGrain. Comparing Two Sample Means t Tests. *Physical Therapy*, 65(11):1730–1733, 1985. ISSN 0031-9023. URL http://ptjournal.apta.org/content/65/11/1730.

[134] Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern Recognition*. Academic Press, 4th edition, 2008. ISBN 1597492728, 9781597492720.

[135] J.A. Quinn and M. Sugiyama. A Least-squares Approach to Anomaly Detection in Static and Sequential Data. *Pattern Recognition Letters*, 40:36–40, 2014.

[136] R. Begg and M. Palaniswami. *Computational Intelligence for Movement Sciences: Neural Networks and other Emerging Technologies*. Idea Group Publishing, 2006.

[137] J. C. Platt. Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pages 61–74. MIT Press, 1999.

[138] Warren S. McCulloch and Walter Pitts. Neurocomputing: Foundations of Research. chapter A Logical Calculus of the Ideas Immanent in Nervous Activity, pages 15–27. MIT Press, Cambridge, MA, USA, 1988. ISBN 0-262-01097-6. URL http://dl.acm.org/citation.cfm?id=65669.104377.

[139] B. G. Quinn. Estimating Frequency by Interpolation using Fourier Coefficients. *IEEE Transactions on Signal Processing*, 42(5):1264–1268, May 1994. ISSN 1053-587X. doi: 10.1109/78.295186.

[140] B.G. Quinn. Estimation of Frequency, Amplitude, and Phase from the DFT of a Time Series. *IEEE Transactions on Signal Processing*, 45(3):814–817, 1997. ISSN 1053587X. doi: 10.1109/78.558515. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=558515.

[141] RL Streit and RF Barrett. Frequency Line Tracking Using Hidden Markov Models. *IEEE Trans. Acoustics. Speech. and Singal Processing*, 38(4):486—-598, 1990. URL http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA263437%5Cnhttp://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=52700.

[142] T. Gunes and N. Erdol. HMM Based Spectral Frequency Line Tracking: Improvements and New Results. *2006 IEEE International Conference on Acoustics Speed and Signal Processing Proceedings*, pages II–673–II–676, 2006. ISSN 15206149. doi: 10.1109/ICASSP.2006.1660432. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1660432`.

[143] Premananda Indic, David Paydarfar, and Riccardo Barbieri. Point Process Modeling of Interbreath Interval: A New Approach for the Assessment of Instability of Breathing in Neonates. *IEEE Transactions on Biomedical Engineering*, 60(10):2858–2866, 2013. ISSN 00189294. doi: 10.1109/TBME.2013.2264162.

[144] Urs Frey, M Silverman, a L Barabási, and Béla Suki. Irregularities and Power Law Distributions in the Breathing Pattern in Preterm and Term Infants. *Journal of Applied Physiology*, pages 789–797, 1998.

[145] Kazuhiko Ozeki. *Theory of Affine Projection Algorithms for Adaptive Filtering*. Springer Publishing Company, Incorporated, 1st edition, 2015. ISBN 4431557377, 9784431557371.

[146] Serena Fiocchi, Ioannis A. Markakis, Paolo Ravazzani, and Theodoros Samaras. SAR exposure from UHF RFID Reader in Adult, Child, Pregnant Woman, and Fetus Anatomical Models. *Bioelectromagnetics*, 34(6):443–452, 2013. ISSN 1521-186X. doi: 10.1002/bem.21789. URL `http://dx.doi.org/10.1002/bem.21789`.

[147] U.S. Government Publishing Office. FCC OET Bulletin 65, 2018. `https://www.fcc.gov/Bureaus/Engineering_Technology/Documents/bulletins/oet65/oet65.pdf`.

[148] Shrenik Vora. *Wireless and Battery-Free Biosignal Monitoring using Passive RFID Tags*. PhD thesis, Drexel University, Philadelphia, PA, 3 2017.

[149] G. B. Moody, R. G. Mark, A. Zoccola, and S. Mantero. Derivation of Respiratory Signals from Multi-lead ECGs. *Computers in Cardiology 1985*, (November):113–116, 1985. ISSN 9780769535616. doi: 10.1109/FBIE.2008.41. URL `http://www.physionet.org/physiotools/edr/cic85/`.

[150] Axel Schäfer and Karl W. Kratky. Estimation of Breathing Rate from Respiratory Sinus Arrhythmia: Comparison of Various Methods. *Annals of Biomedical Engineering*, 36(3):476–485, 2008. ISSN 00906964. doi: 10.1007/s10439-007-9428-1.

Part V

Appendix

Appendix A: Frequency Tracking via a Hidden Markov Chain of Interpolated Discrete Fourier Transform Time-Series Estimates

Data: Short-window time series vector v of RSSI time series data, resampled to a known constant sampling frequency for spectral analysis
Result: Most likely biologically feasible fundamental oscillatory frequency
Function `QuinnInterpolate(`M, F, x`)`**:**

`/* Using Quinn interpolation, compute the resulting frequency` $\hat{x}$ `from frequency bin` $F[x]$ `given frequencies` F `and magnitudes` M `*/`

$\delta_+ \leftarrow (\Re M[x+1] \times \Re M[x] + \Im M[x+1] \times \Im M[x]) \times (\Re M[x]^2 + \Im M[x]^2)^{-1}$;
$\delta_+ \leftarrow -\delta_+ \times (1-\delta_+)^{-1}$;
$\delta_- \leftarrow (\Re M[x-1] \times \Re M[x] + \Im M[x-1] \times \Im M[x]) \times (\Re M[x]^2 + \Im M[x]^2)^{-1}$;
$\delta_- \leftarrow \delta_- \times (1-\delta_-)^{-1}$;
$\mathcal{K}_+ \leftarrow \frac{1}{4}\log(3\delta_+^4 + 6\delta_+^2 + 1) - \frac{\sqrt{6}}{24} \times \log((\delta_+^2 + 1 - \sqrt{\frac{2}{3}}) \times (\delta_+^2 + 1 + \sqrt{\frac{2}{3}})^{-1})$;
$\mathcal{K}_- \leftarrow \frac{1}{4}\log(3\delta_-^4 + 6\delta_-^2 + 1) - \frac{\sqrt{6}}{24} \times \log((\delta_-^2 + 1 - \sqrt{\frac{2}{3}}) \times (\delta_-^2 + 1 + \sqrt{\frac{2}{3}})^{-1})$;
$\hat{x} \leftarrow \frac{1}{2}(\delta_+ + \delta_-) + \mathcal{K}_+ - \mathcal{K}_-$;
return $F[\hat{x}]$;
`/*` $F[\hat{x}]$ `is computed by taking a weighted average of the two adjacent frequency bins, weighted by` $\hat{x}$ `and` $1-\hat{x}$ `*/`

$M, F \leftarrow \mathrm{FFT}(\mathrm{v})$;
`/* M, F are the FFT magnitudes and frequencies */`
$L \leftarrow$ find_biofeasible_localmax_idx(M);
$Q \leftarrow []$;
for $k \in L$ **do**

`/* Find the local maxima FFT frequencies, interpolating the real (`$\Re$`) and imaginary (`$\Im$`) magnitude components of each maximum frequency by Quinn's algorithm */`
$Q_i \leftarrow Q_i \cup$ `QuinnInterpolate(`M, F, k`)`;

end

`/* Eliminate frequencies that were not present in the previous time iteration, within a magnitude threshold` σ `and frequency bin distance` ν `*/`

$Q_{i,k} \leftarrow 0 \colon \nexists Q_{i-1,l} \colon |Q_{i,k} - Q_{i-1,l}| \leq \sigma; l = \{k-\nu, ...k+\nu\}$;

`/* Using the Forward-Backward algorithm of Baum and Welch, compute the most likely frequency from the current locally-maximal FFT magnitudes and the corresponding locally-maximal FFT magnitudes from prior time steps, using the resulting FFT magnitude corresponding to the most likely frequency as an HMM observation */`

$B_i \leftarrow |M_k| \colon Q_{i,k} \neq 0$;
$B_{i,j} \leftarrow \frac{B_{i,j}}{\sum_k B_{i,k}}$;
$A_i \leftarrow \{(2\pi\sigma^2)^{-1} e^{\frac{(k-l)^2}{2\sigma^2}}\}|_{k,l=[0,|Q_i|],k\neq l}$;
$A_{i,j} \leftarrow \frac{A_{i,j}}{\sum_k A_{i,k}}$;
$\alpha_i \leftarrow B_{i-1,k} \sum_j (A_{j,i}\alpha_{i-1,j})$;
$\beta_i \leftarrow \sum_j (|B_{i-1,j}| A_{i,j}\beta_{i-1,j})$;
$\gamma \leftarrow \alpha_{i-1,j}\beta_{i-1,j}(\sum_k \alpha_{i-1,k}\beta_{i-1,k})^{-1}$;
$x \leftarrow \mathrm{argmax}\ \gamma$

`/* Using Quinn interpolation, compute the resulting frequency` z `from most likely frequency bin` $F[x]$ `and corresponding magnitude` $M[x]$ `*/`

$z \leftarrow$ `QuinnInterpolate(`M, F, x`)`;

Algorithm 1: An algorithm to compute z, the most likely biologically-feasible oscillatory frequency from a time-series window of frequency-normalized RFID RSSI strain gauge data

Appendix B: Modifications to *sllrp* Library to Enable Physical Layer Impinj Extensions and Message Queues

The following physical layer library revisions enable selection of antenna and frequency (where geographically available) for Impinj interrogators. Additionally, we enable the use of message queues according to the Producer-Consumer pattern to decrease the delay induced at the network layer due to communications overhead in transmitting each received tag message. Instead, tags are placed into a queue immediately upon receipt, and transmitted for processing in small batches at regular intervals (*i.e.*, every 0.5 seconds). Nagle's Algorithm is disabled at the transport layer as a result of the Producer-Consumer pattern and batch transmission manually implemented here, so that transmission and delay is better controlled at the application layer. Finally, Impinj extensions are enabled on initial connection to the interrogator to enable the measurement of Doppler shift and phase angle for each received backscatter response.

B.1 Modifications to *llrp.diff* in the *sllrp* Library

llrp.diff

```
Index: llrp.py
===================================================================
--- llrp.py    (revision 411)
+++ llrp.py    (revision 1620)
@@ -15,6 +15,8 @@
 from twisted.internet.protocol import ClientFactory
 from twisted.protocols.basic import LineReceiver
 from twisted.internet.error import ReactorAlreadyRunning, ReactorNotRunning
+import threading # mongan 8/22/16
+import Queue # mongan 8/22/16

 LLRP_PORT = 5084

@@ -72,16 +74,23 @@
                   data[:self.full_hdr_len])
         ver = (msgtype >> 10) & BITMASK(3)
         msgtype = msgtype & BITMASK(10)
+        name = '' # mongan 11/10/16
         try:
             name = Message_Type2Name[msgtype]
             logger.debug('deserializing {} command'.format(name))
             decoder = Message_struct[name]['decode']
         except KeyError:
```

```
-            raise LLRPError('Cannot find decoder for message type '
-                    '{}'.format(msgtype))
+            raise LLRPError('Cannot find decoder for message name {}, message type '
+                    '{}'.format(name, msgtype)) # mongan 8/17/15
         body = data[self.full_hdr_len:length]
         try:
+            if name == "ENABLE_IMPINJ_EXTENSIONS_RESPONSE": # mongan 8/18/15
             self.msgdict = {
+                    name: dict()
+                }
+                self.msgdict[name]['Result'] = decoder(body)[0]
+            else:
+                self.msgdict = {
                name: dict(decoder(body))
             }
             self.msgdict[name]['Ver'] = ver
@@ -90,8 +99,8 @@
             logger.debug('done deserializing {} command'.format(name))
         except LLRPError as e:
             logger.warning('Problem with {} message format: {}'.format(name, e))
+            return
             return ''
-        return ''

     def isSuccess (self):
         if not self.msgdict:
@@ -106,6 +115,8 @@
                     ['ConnectionAttemptEvent']['Status'] == 'Success'
             elif 'LLRPStatus' in md:
                 return md['LLRPStatus']['StatusCode'] == 'Success'
+            elif 'Result' in md:
+                return md['Result']['LLRPStatus']['StatusCode'] == 'Success' # mongan 8/18/15
         except KeyError as KE:
             logger.error('failed to parse status from {}: {}'.format(msgName,
                     KE))
@@ -136,6 +147,7 @@
     STATE_SENT_GET_CAPABILITIES = 9
     STATE_PAUSING = 10
     STATE_PAUSED = 11
+    STATE_SENT_IMPINJ_EXTENSIONS = 12

     @classmethod
     def getStates (_):
@@ -157,7 +169,7 @@
             start_inventory=True, reset_on_connect=True,
             disconnect_when_done=True,
             tag_content_selector={},
-            session=2, tag_population=4):
+            session=2, tag_population=4, mode_index=-1, channellist=[], periodictrigger=100):
         self.factory = factory
         self.setRawMode()
```

```
75          self.state = LLRPClient.STATE_DISCONNECTED
76  @@ -181,6 +193,9 @@
77          self.tag_content_selector = tag_content_selector
78          if self.start_inventory:
79              logger.info('will start inventory on connect')
80  +       self.mode_index = mode_index # mongan 8/7/15
81  +       self.channellist = channellist # mongan 6/6/2016
82  +       self.periodictrigger = periodictrigger # mongan 7/19/2016
83
84          logger.info('using antennas: {}'.format(self.antennas))
85
86  @@ -202,6 +217,12 @@
87
88          self.disconnecting = False
89          self.rospec = None
90  +
91  +       # mongan 8/22/16
92  +       self.dataqueue = Queue.Queue()
93  +       self.exiting = False
94  +       self.datathread = threading.Thread(target=self.datahandler, args=())
95  +       self.datathread.start()
96
97      def addStateCallback (self, state, cb):
98          self._state_callbacks[state].append(cb)
99  @@ -211,6 +232,7 @@
100
101     def connectionMade (self):
102         self.transport.setTcpKeepAlive(True)
103 +       self.transport.setTcpNoDelay(True) # mongan 8/12/16 disable Nagle to help
    ↪  reduce lag
104         self.peername = self.transport.getHandle().getpeername()
105         logger.info('connected to {}'.format(self.peername))
106         self.factory.protocols.add(self)
107 @@ -313,6 +335,8 @@
108         logger.debug('starting message callbacks for {}'.format(msgName))
109         for fn in self._message_callbacks[msgName]:
110             fn(lmsg)
111 +           #t = threading.Thread(target=fn, args=(lmsg,)) # mongan 8/22/16
112 +           #t.start()
113         logger.debug('done with message callbacks for {}'.format(msgName))
114
115         # keepalives can occur at any time
116 @@ -392,6 +416,33 @@
117             elif self.start_inventory:
118                 self.startInventory()
119
120 +       # mongan 8/13/15
121 +       elif self.state == LLRPClient.STATE_SENT_IMPINJ_EXTENSIONS:
122 +           if msgName != 'ENABLE_IMPINJ_EXTENSIONS_RESPONSE':
123 +               logger.error('unexpected response {} ' \
124 +                       ' when receiving Impinj Extensions
    ↪  Response'.format(msgName))
125 +               return
126 +
```

```
127 +            if not lmsg.isSuccess():
128 +                logger.fatal('Error sending Impinj Extensions:
    ↪  {}'.format(lmsg.msgdict[msgName]))
129 +                return
130 +
131 +            self.processDeferreds(msgName, lmsg.isSuccess())
132 +
133 +            rospec = self.getROSpec()['ROSpec'] # mongan 8/18/15
134 +
135 +
136 +            started = defer.Deferred()
137 +            started.addCallback(self._setState_wrapper,
138 +                LLRPClient.STATE_INVENTORYING)
139 +            started.addErrback(self.panic, 'ENABLE_ROSPEC failed')
140 +
141 +            d = defer.Deferred() # mongan 8/18/15
142 +            d.addCallback(self.send_ENABLE_ROSPEC, rospec, onCompletion=started)
143 +            d.addErrback(self.panic, 'ADD_ROSPEC failed')
144 +
145 +            self.send_ADD_ROSPEC(rospec, onCompletion=d) # mongan 8/15/15; add
    ↪  rospec after enabling extensions
146 +
147          # in state SENT_ADD_ROSPEC, expect only ADD_ROSPEC_RESPONSE; respond to
148          # favorable ADD_ROSPEC_RESPONSE by enabling the added ROSpec and
149          # advancing to state SENT_ENABLE_ROSPEC.
150 @@ -491,21 +542,10 @@
151 +    def datahandler(self):
152 +        while not self.exiting:
153 +            data = self.dataqueue.get(block=True)
154 -    def rawDataReceived (self, data):
155 -        logger.debug('got {} bytes from reader: {}'.format(len(data),
156 -                    data.encode('hex')))
157 -        if self.expectingRemainingBytes:
158 -            if len(data) >= self.expectingRemainingBytes:
159 -                data = self.partialData + data
160 -                self.partialData = ''
161 -                self.expectingRemainingBytes -= len(data)
162 -            else:
163 -                # still not enough; wait until next time
164 -                self.partialData += data
165 -                self.expectingRemainingBytes -= len(data)
166 -                return
167 -
168 @@ -539,7 +579,24 @@
169 +
170 +    def rawDataReceived (self, data):
171 +        logger.debug('got {} bytes from reader: {}'.format(len(data),
172 +                    data.encode('hex')))
173
174 +        if self.expectingRemainingBytes:
175 +            if len(data) >= self.expectingRemainingBytes:
176 +                data = self.partialData + data
177 +                self.partialData = ''
178 +                self.expectingRemainingBytes -= len(data)
```

```
179 +            else:
180 +                # still not enough; wait until next time
181 +                self.partialData += data
182 +                self.expectingRemainingBytes -= len(data)
183 +                return
184 +
185 +        self.dataqueue.put(data)
186 +
187      def panic (self, failure, *args):
188          logger.error('panic(): {}'.format(args))
189          logger.error(failure.getErrorMessage())
190 @@ -564,6 +621,17 @@
191          self.setState(LLRPClient.STATE_SENT_GET_CAPABILITIES)
192          self._deferreds['GET_READER_CAPABILITIES_RESPONSE'].append(onCompletion)
193
194 +    # mongan 8/13/15
195 +    def send_ENABLE_IMPINJ_EXTENSIONS (self, onCompletion):
196 +        self.sendLLRPMessage(LLRPMessage(msgdict={
197 +            'ImpinjEnableExtensions': {
198 +                'Ver':  1,
199 +                'Type': 1023,
200 +                'ID': 0,
201 +            }}))
202 +        self.setState(LLRPClient.STATE_SENT_IMPINJ_EXTENSIONS)
203 +        self._deferreds['ENABLE_IMPINJ_EXTENSIONS_RESPONSE'].append(onCompletion)
204 +
205      def send_ADD_ROSPEC (self, rospec, onCompletion):
206          self.sendLLRPMessage(LLRPMessage(msgdict={
207              'ADD_ROSPEC': {
208 @@ -689,23 +757,25 @@
209              logger.warn('ignoring startInventory() while already inventorying')
210              return None
211
212 -        rospec = self.getROSpec()['ROSpec']
213 +        rospec = self.getROSpec()['ROSpec']
214
215          logger.info('starting inventory')
216
217 -        started = defer.Deferred()
218 -        started.addCallback(self._setState_wrapper,
219 -                LLRPClient.STATE_INVENTORYING)
220 -        started.addErrback(self.panic, 'ENABLE_ROSPEC failed')
221 +        # mongan 8/18/15
222 +        #started = defer.Deferred()
223 +        #started.addCallback(self._setState_wrapper,
224 +        #        LLRPClient.STATE_INVENTORYING)
225 +        #started.addErrback(self.panic, 'ENABLE_ROSPEC failed')
226
227          if self.duration:
228              task.deferLater(reactor, self.duration, self.stopPolitely, True)
229
230          d = defer.Deferred()
231 -        d.addCallback(self.send_ENABLE_ROSPEC, rospec, onCompletion=started)
232 -        d.addErrback(self.panic, 'ADD_ROSPEC failed')
```

```
+        #d.addCallback(self.send_ENABLE_ROSPEC, rospec, onCompletion=started) # mongan 8/18/15
+        d.addErrback(self.panic, 'ENABLE_IMPINJ_EXTENSIONS failed') # mongan 8/18/15 was ADD_ROSPEC

-        self.send_ADD_ROSPEC(rospec, onCompletion=d)
+        self.send_ENABLE_IMPINJ_EXTENSIONS(onCompletion=d) # mongan 8/13/15
+        #self.send_ADD_ROSPEC(rospec, onCompletion=d) # mongan 8/15/15

     def getROSpec (self):
         if self.rospec:
@@ -717,7 +787,9 @@
                                tx_power=self.tx_power, antennas=self.antennas,
                                tag_content_selector=self.tag_content_selector,
                                session=self.session,
-                               tag_population=self.tag_population)
+                               tag_population=self.tag_population,
+                               modeidx=self.mode_index,
+                               channellist=self.channellist,
+                               periodictrigger=self.periodictrigger) #mongan 6/6/2016 (channellist) 8/15/15 (modeidx), 7/19/16 (periodictrigger)
         logger.debug('ROSpec: {}'.format(self.rospec))
         return self.rospec

@@ -827,6 +899,8 @@
         self._message_callbacks = defaultdict(list)

         self.protocols = set()
+
+        self.proto = None

     def startedConnecting(self, connector):
         logger.info('connecting...')
@@ -839,19 +913,22 @@
         self._message_callbacks['RO_ACCESS_REPORT'].append(cb)

     def buildProtocol(self, _):
-        proto = LLRPClient(factory=self, **self.client_args)
+        if 'ntari' in self.client_args: # mongan 5/3/17
+            del self.client_args['ntari']

+        self.proto = LLRPClient(factory=self, **self.client_args)
+
         # register state-change callbacks with new client
         for state, cbs in self._state_callbacks.items():
             for cb in cbs:
-                proto.addStateCallback(state, cb)
+                self.proto.addStateCallback(state, cb)

         # register message callbacks with new client
         for msg_type, cbs in self._message_callbacks.items():
             for cb in cbs:
-                proto.addMessageCallback(msg_type, cb)
+                self.proto.addMessageCallback(msg_type, cb)
```

```
283
284 -         return proto
285 +         return self.proto
286
287       def clientConnectionLost(self, connector, reason):
288           logger.info('lost connection: {}'.format(reason.getErrorMessage()))
```

B.2 Modifications to *llrp_proto.diff* in the *sllrp* Library

llrp_proto.diff

```
1  Index: llrp_proto.py
2  ===================================================================
3  --- llrp_proto.py    (revision 411)
4  +++ llrp_proto.py    (revision 1620)
5  @@ -1843,21 +1846,50 @@
6  +# mongan 7/19/16 16.2.4.1.1.1.1 PeriodicTriggerValue Parameter
7  +def encode_PeriodicTriggerValue(par):
8  +    msgtype = Message_struct['PeriodicTriggerValue']['type']
9  +
10 +    msg_header = '!HHII'
11 +    msg_header_len = struct.calcsize(msg_header)
12 +
13 +    offset = 0
14 +
15 +    # utc timestamp parameter after the periodic trigger value is optional and so
   ↪  is omitted; so that the first time is the first message receipt plus the
   ↪  offset which we assume to be 0 so that the first message is delivered right
   ↪  away to start
16 +    data = ''
17 +    data = struct.pack(msg_header, msgtype,
18 +            len(data) + msg_header_len, offset, par['PeriodicTriggerValue']) +
   ↪  data
19 +
20 +    return data
21 +
22 +Message_struct['PeriodicTriggerValue'] = {
23 +    'type': 180,
24 +    'fields': [
25 +        'Type',
26 +        'Length',
27 +        'Offset',
28 +        'Period',
29 +        'UTCTimestamp'
30 +
31 +    ],
32 +    'encode': encode_PeriodicTriggerValue
33 +}
34 +
35 @@ -2047,7 +2079,10 @@
36      if 'C1G2SingulationControl' in par:
37          data += encode('C1G2SingulationControl')(par['C1G2SingulationControl'])
38      # XXX custom parameters
39 -
```

```
+    # mongan 6/6/2016
+    if 'ImpinjFixedFrequencyMode' in par:
+        data +=
 ↪  encode('ImpinjFixedFrequencyMode')(par['ImpinjFixedFrequencyMode'])
+
     data = struct.pack(msg_header, msgtype,
             len(data) + struct.calcsize(msg_header)) + data
     return data
@@ -2058,8 +2093,9 @@
         'TagInventoryStateAware',
         'C1G2Filter',
         'C1G2RFControl',
-        'C1G2SingulationControl'
+        'C1G2SingulationControl',
         # XXX custom parameters
+        'ImpinjFixedFrequencyMode'
     ],
     'encode': encode_C1G2InventoryCommand
 }
@@ -2116,21 +2152,77 @@
     'encode': encode_C1G2SingulationControl
 }

+# mongan 6/6/2016 Impinj antenna select
+def encode_ImpinjFixedFrequencyMode(par):
+    msgtype = Message_struct['ImpinjFixedFrequencyMode']['type']
+
+    msg_header = '!HH'
+
+    data = struct.pack('!I',
 ↪  Message_struct['ImpinjFixedFrequencyMode']['vendorid'])
+    data += struct.pack('!I',
 ↪  Message_struct['ImpinjFixedFrequencyMode']['subtype'])
+
+    if 'FixedFrequencyMode' in par and par['FixedFrequencyMode']:
+        data += struct.pack('!H', par['FixedFrequencyMode'])
+    data += struct.pack('!H', 0) # reserved
+    if 'ChannelListCount' in par and par['ChannelListCount']:
+        data += struct.pack('!H', par['ChannelListCount'])
+    if 'ChannelsList' in par and par['ChannelsList']:
+        for c in par['ChannelsList']:
+            data += struct.pack('!H', c)
+
+    data = struct.pack(msg_header, msgtype,
+            len(data) + struct.calcsize(msg_header)) + data
+
+    return data
+
+Message_struct['ImpinjFixedFrequencyMode'] = {
+    'type': 1023, # Custom Parameter
+    'subtype': 26, # FixedFrequencyList
+    'vendorid': 25882, # Impinj
+    'id': 0,
+    'fields': [
```

```
+        'FixedFrequencyMode', #bits 0-16, skip 16-32 as reserved, then
+        'ChannelListCount', # bits 0-16, followed by list of 16-bit channels
+        'ChannelsList' # array of channels to append as 16-bit channels list right after ChannelListCount
+    ],
+    'encode': encode_ImpinjFixedFrequencyMode
+}
+
+# mongan 8/13/2015 Impinj enable extensions; must call before adding ROSpec if using Impinj extensions
+def encode_ImpinjEnableExtensions (par):
+    data = struct.pack('!I', Message_struct['ImpinjEnableExtensions']['vendorid'])
+    data += struct.pack('!B', Message_struct['ImpinjEnableExtensions']['subtype'])
+    data += struct.pack('!I', 0) # reserved space for future use in protocol
+
+    return data
+
+Message_struct['ImpinjEnableExtensions'] = {
+    'type': 1023,
+    'vendorid': 25882,
+    'subtype': 21,
+    'fields': [
+        'Ver', 'Type', 'ID',
+    ],
+    'encode': encode_ImpinjEnableExtensions
+}
+


+    if 'ImpinjTagReportContentSelector' in par and par['ImpinjTagReportContentSelector']:    # mongan 8/13/15 if impinj tag selector group given, encode it
+        data += encode('ImpinjTagReportContentSelector')(par['ImpinjTagReportContentSelector'])
+
     data = struct.pack(msg_header, msgtype,
             len(data) + msg_header_len,
             roReportTrigger, n) + data

     return data
@@ -2138,17 +2230,137 @@
     'fields': [
         'N',
         'ROReportTrigger',
-        'TagReportContentSelector'
+        'TagReportContentSelector',
+        'ImpinjTagReportContentSelector', # mongan 8/8/2015
     ],
     'encode': encode_ROReportSpec
 }

+# mongan 8/8/2015
+def encode_ImpinjTagReportContentSelector (par):
```

```
138 +    msgtype = Message_struct['ImpinjTagReportContentSelector']['type']
139 +
140 +    msg_header = '!HH'
141 +
142 +    data = struct.pack('!I',
    ↪  Message_struct['ImpinjTagReportContentSelector']['vendorid'])
143 +    data += struct.pack('!I',
    ↪  Message_struct['ImpinjTagReportContentSelector']['subtype'])
144 +
145 +    if 'ImpinjEnableRFPhaseAngle' in par and par['ImpinjEnableRFPhaseAngle']:
146 +        data +=
    ↪  encode('ImpinjEnableRFPhaseAngle')(par['ImpinjEnableRFPhaseAngle'])
147 +    if 'ImpinjEnableRFDopplerFrequency' in par and
    ↪  par['ImpinjEnableRFDopplerFrequency']:
148 +        data +=
    ↪  encode('ImpinjEnableRFDopplerFrequency')(par['ImpinjEnableRFDopplerFrequency'])
149 +
150 +    data = struct.pack(msg_header, msgtype,
151 +            len(data) + struct.calcsize(msg_header)) + data
152 +
153 +    return data
154 +
155 +# mongan 8/8/2015
156 +Message_struct['ImpinjTagReportContentSelector'] = {
157 +    'type': 1023, # Custom Parameter
158 +    'subtype': 50, # Tag Report Content Selector
159 +    'vendorid': 25882, # Impinj
160 +    'id': 0,
161 +    'fields': [
162 +        'ImpinjEnableRFPhaseAngle',
163 +        'ImpinjEnableRFDopplerFrequency'
164 +    ],
165 +    'encode': encode_ImpinjTagReportContentSelector
166 +}
167 +
168 +# mongan 8/17/2015
169 +# the ENABLE_IMPINJ_EXTENSIONS_RESPONSE type is 1023, a custom parameter, as is
    ↪  the subtypes for doppler and phase, so we'll handle these here as well
170 +# this is called by llrp deserialize, with ver, etc. already extracted
171 +def decode_ImpinjCustomParameter (data):
172 +    if len(data) == 0:
173 +        return None, data
174 +
175 +    msg = {}
176 +    logger.debug('%s' % func())
177 +
178 +    # Decode parameters
179 +    msg['vendor'] = struct.unpack('!I', data[:struct.calcsize('!I')])[0]
180 +    data = data[struct.calcsize('!I'):]
181 +    msg['subtype'] = struct.unpack('!B', data[:struct.calcsize('!B')])[0]
182 +    data = data[struct.calcsize('!B'):]
183 +
184 +    # subtype 22 is EXTENSIONS_RESPONSE
185 +    ret, data = decode_LLRPStatus(data)
```

```
186 +    msg['LLRPStatus'] = ret
187 +
188 +    return msg, data
189 +
190 +# mongan 8/17/2015
191 +Message_struct['ENABLE_IMPINJ_EXTENSIONS_RESPONSE'] = {
192 +    'type': 1023, # Custom Parameter
193 +    'fields': [
194 +        'Ver', 'Type', 'ID',
195 +        'vendor', 'subtype',
196 +        'LLRPStatus'
197 +    ],
198 +    'decode': decode_ImpinjCustomParameter
199 +}
200 +
201 +# mongan 8/8/2015
202 +def encode_ImpinjEnableRFPhaseAngle (par):
203 +    msgtype = Message_struct['ImpinjEnableRFPhaseAngle']['type']
204 +
205 +    msg_header = '!HH'
206 +
207 +    data = struct.pack('!I',
    ↪  Message_struct['ImpinjEnableRFPhaseAngle']['vendorid'])
208 +    data += struct.pack('!I',
    ↪  Message_struct['ImpinjEnableRFPhaseAngle']['subtype'])
209 +
210 +    data += struct.pack('!H', 1)
211 +
212 +    data = struct.pack(msg_header, msgtype,
213 +            len(data) + struct.calcsize(msg_header)) + data
214 +
215 +    return data
216 +
217 +# mongan 8/8/2015
218 +Message_struct['ImpinjEnableRFPhaseAngle'] = {
219 +    'type': 1023, # Custom Parameter
220 +    'subtype': 52, # Enable RF Phase Angle
221 +    'vendorid': 25882, # Impinj
222 +    'fields': [
223 +
224 +    ],
225 +    'encode': encode_ImpinjEnableRFPhaseAngle
226 +}
227 +
228 +# mongan 8/8/2015
229 +def encode_ImpinjEnableRFDopplerFrequency (par):
230 +    msgtype = Message_struct['ImpinjEnableRFDopplerFrequency']['type']
231 +
232 +    msg_header = '!HH'
233 +
234 +    data = struct.pack('!I',
    ↪  Message_struct['ImpinjEnableRFDopplerFrequency']['vendorid'])
235 +    data += struct.pack('!I',
    ↪  Message_struct['ImpinjEnableRFDopplerFrequency']['subtype'])
```

```
+    data += struct.pack('!H', 1)
+
+    data = struct.pack(msg_header, msgtype,
+            len(data) + struct.calcsize(msg_header)) + data
+
+    return data
+
+# mongan 8/8/2015
+Message_struct['ImpinjEnableRFDopplerFrequency'] = {
+    'type': 1023, # Custom Parameter
+    'subtype': 67, # Enable RF Doppler Frequency
+    'vendorid': 25882, # Impinj
+    'fields': [
+
+    ],
+    'encode': encode_ImpinjEnableRFDopplerFrequency
+}
+
@@ -2216,11 +2428,36 @@
+
+    # mongan 8/18/15
+    # decode TLV parameters like custom values
+    while body:
+        tlv_ver_type = struct.unpack('!H', body[:struct.calcsize('!H')])[0]
+        if tlv_ver_type == 1023: # CustomParameter (Impinj)
+            body = body[struct.calcsize('!H'):]
+            tlv_length = struct.unpack('!H', body[:struct.calcsize('!H')])[0]
+            body = body[struct.calcsize('!H'):]
+            tlv_vendor = struct.unpack('!I', body[:struct.calcsize('!I')])[0]
+            body = body[struct.calcsize('!I'):]
+            tlv_subtype = struct.unpack('!I', body[:struct.calcsize('!I')])[0]
+            body = body[struct.calcsize('!I'):]
+            tlv_value = struct.unpack('!H', body[:struct.calcsize('!H')])[0]
+            body = body[struct.calcsize('!H'):]
+
+            # 56 is phase angle, 68 is doppler
+            if tlv_subtype == 56:
+                par['RFPhaseAngle'] = tlv_value
+            elif tlv_subtype == 68:
+                par['Doppler'] = tlv_value
+        else:
+            break
+
+        #print tlv_ver_type, tlv_length, tlv_vendor, tlv_subtype, tlv_value
+
@@ -2244,6 +2481,8 @@
         'AirProtocolTagData',
         'AccessSpecID',
         'OpSpecResult',
+        'RFPhaseAngle', # mongan 8/18/15
+        'Doppler', # mongan 8/18/15
     ],
     'decode': decode_TagReportData
 }
```

```
@@ -2776,11 +3015,14 @@
                     str += __llrp_data2xml(e, p, level + 1)
             else:
                 str += tabs + '\t<%s>%s</%s>\n' % (p, sub, p)
-
+
         str += tabs + '</%s>\n' % name
-
+
         return str

+    if msg == None: # mongan 6/23/2016
+        return ''
+
     ans = ''
     for p in msg:
         ans += __llrp_data2xml(msg[p], p)
@@ -2790,7 +3032,7 @@
     def __init__(self, llrpcli, msgid, priority=0, state='Disabled',
             antennas=(1,), tx_power=91, duration_sec=None,
             report_every_n_tags=None, tag_content_selector={},
-            session=2, tag_population=4):
+            session=2, tag_population=4, modeidx=-1, channellist=[],
  ↪  periodictrigger=100, ntari=-1): #mongan 6/6/2016 to add channellist, added
 ↪  modeidx 8/7/15, periodictrigger 7/19/16, ntari 9/28/16
         # Sanity checks
         if msgid <= 0:
             raise LLRPError('invalid ROSpec message ID {} (need >0)'.format(\
@@ -2801,26 +3043,33 @@
         if not state in ROSpecState_Name2Type:
             raise LLRPError('invalid ROSpec state {} (need [{}])'.format(\
                     state, ','.join(ROSpecState_Name2Type.keys())))
-
+
         rmode = llrpcli.reader_mode
+        if modeidx == -1:
             mode_index = rmode['ModeIdentifier']
+        else:
+            mode_index = modeidx # mongan 8/7/15
+
+        if ntari == -1:
             tari = rmode['MaxTari']
-
+        else:
+            tari = ntari
+
         tagReportContentSelector = {
-            'EnableROSpecID': False,
-            'EnableSpecIndex': False,
-            'EnableInventoryParameterSpecID': False,
+            'EnableROSpecID': True,
+            'EnableSpecIndex': True,
+            'EnableInventoryParameterSpecID': True,
             'EnableAntennaID': True,
```

```
342 -             'EnableChannelIndex': False,
343 +             'EnableChannelIndex': True,
344               'EnablePeakRRSI': True,
345 -             'EnableFirstSeenTimestamp': False,
346 +             'EnableFirstSeenTimestamp': True,
347               'EnableLastSeenTimestamp': True,
348               'EnableTagSeenCount': True,
349 -             'EnableAccessSpecID': False,
350 +             'EnableAccessSpecID': True,
351           }
352           if tag_content_selector:
353               tagReportContentSelector.update(tag_content_selector)
354 -
355 +
356           self['ROSpec'] = {
357               'ROSpecID': msgid,
358               'Priority': priority,
359 @@ -2827,7 +3076,8 @@
360               'CurrentState': state,
361               'ROBoundarySpec': {
362                   'ROSpecStartTrigger': {
363 -                     'ROSpecStartTriggerType': 'Immediate',
364 +                     'ROSpecStartTriggerType': 'Immediate', # was Immediate, can
    ↪  be Periodic in which case periodic trigger will be used; mongan 7/19/16
365 +                     'PeriodicTriggerValue': periodictrigger
366                   },
367                   'ROSpecStopTrigger': {
368                       'ROSpecStopTriggerType': 'Null',
369 @@ -2837,7 +3087,7 @@
370               'AISpec': {
371                   'AntennaIDs': ' '.join(map(str, antennas)),
372                   'AISpecStopTrigger': {
373 -                     'AISpecStopTriggerType': 'Duration',
374 +                     'AISpecStopTriggerType': 'Null', # was Duration mongan
    ↪  7/20/16
375                       'DurationTriggerValue': 500,
376                   },
377                   'InventoryParameterSpec': {
378 @@ -2850,10 +3100,14 @@
379                   'ROReportTrigger': 'Upon_N_Tags_Or_End_Of_AISpec',
380                   'N': 1,
381                   'TagReportContentSelector': tagReportContentSelector,
382 +                 'ImpinjTagReportContentSelector': { # mongan 8/8/2015
383 +                     'ImpinjEnableRFPhaseAngle': True, # append this so that if
    ↪  using Impinj, this custom parameter will cause it to output doppler and phase
    ↪  by default
384 +                     'ImpinjEnableRFDopplerFrequency': True
385                   },
386 +             },
387           }
388 @@ -2877,18 +3131,27 @@
389 +
390 +               # mongan 6/6/2016 - parameterize to select frequencies; FCC does
    ↪  not allow a single frequency continuously / without delay
```

```
391 +            if len(channellist) > 0:
392 +                for antconfig in
    ↪ self['ROSpec']['AISpec']['InventoryParameterSpec']['AntennaConfiguration']:
393 +                    antconfig['C1G2InventoryCommand']['ImpinjFixedFrequencyMode']
    ↪ = {
394 +                        'FixedFrequencyMode': 2,
395 +                        'ChannelListCount': len(channellist),
396 +                        'ChannelsList': channellist
397 +                    }
398 +
```

Appendix C: Institutional Review Board Approval Letters for Relevant Human Studies Protocols

C.1 IRB Approval for Infant Respiratory Monitoring

Drexel UNIVERSITY
Office of Research

APPROVAL OF PROTOCOL

February 13, 2018

Kapil Dandekar, M.D.

Dear Dr. Dandekar:

On 02/13/2018 the IRB reviewed the following protocol:

Type of Review:	Continuation Expedited
Title:	Smart Fabric Bellyband to Monitor Infant Respiratory Activity
Investigator:	Kapil Dandekar, M.D.
IRB ID:	1504003601-R002
Funding:	None
Grant Title:	None
Grant ID:	None
IND, IDE or HDE:	None
Documents Reviewed:	HRP 212 Continuing Review Progress Report, Previous Action, HRP 502 Informed Consent, Confirmed Trainings

The IRB approved the protocol from 02/19/2018 to 02/18/2019 inclusive.

On 02/13/2018 – According to 45 CFR 46.109(e) this study has been Approved Expedited Renewal. Renewal Approval Includes Enrollment of Infants from Hahnemann University Hospital Neonatal Intensive Care Unit (NICU) located at 230 N Broad St, Philadelphia, PA 19102 to wear the Smart Fabric Bellyband to monitor infant respiratory activity: 10 Approved; 0 Enrolled; 10 Remaining to Complete the Study; HRP 502 Informed Consent Version February 2016.
According to 45 CFR 46.110, this study is Approved Expedited Categories 1B and 7.
According to 21 CFR 812.2(b) Non-Significant Risk Device
Risks to Children: 45 CFR 46.404 Minimal Risk.
Permission of one parent is sufficient even if the other parent is alive, known, competent, reasonably available, and shares legal responsibility for the care and custody of the child.

Before 02/13/2019, which is 45 days prior to study closure, you are to submit a completed "FORM: Continuing Review Progress Report (HRP-212)" and required attachments to request continuing approval or closure.

If continuing review approval is not granted before the expiration date of 02/18/2019 approval of this protocol expires on that date.

Attached are stamped approved consent documents. Use copies of these documents to document consent.

In conducting this protocol you are required to follow the requirements listed in the INVESTIGATOR MANUAL (HRP-103).
Sincerely,

Barbara Ferrigno
Member, Medical IRB # 1

Drexel University
Hahnemann University Hospital
Permission to Take Part
In a Research Study

1. Title of research study:

"SMART FABRIC BELLYBAND" to monitor infant respiratory activity.

2. Researcher: Kapil Dandekar, Genevieve Dion, Adam Fontecchio, Timothy Kurzweg, William Mongan, Owen Montgomery, Endla Anday

3. Why you are being invited to take part in a research study

We invite you to take part in a research study because you have an infant, and we are developing a passive, wireless device to monitor motion and breathing activity.

4. What you should know about a research study

- Someone will explain this research study to you.
- Whether or not you take part is up to you.
- You can choose not to take part.
- You can agree to take part now and change your mind later.
- If you decide to not be a part of this research no one will hold it against you.
- Feel free to ask all the questions you want before you decide.

5. Who can you talk to about this research study?

If you have questions, concerns, or complaints, or think the research has hurt you, talk to the research team at William Mongan: 215-895-0286

This research has been reviewed and approved by an Institutional Review Board (IRB). An IRB reviews research projects so that steps are taken to protect the rights and welfare of human subjects taking part in research. You may talk to them at (215) 762-3944 or email HRPP@drexel.edu for any of the following:

- Your questions, concerns, or complaints are not being answered by the research team.
- You cannot reach the research team.
- You want to talk to someone besides the research team.
- You have questions about your rights as a research subject.
- You want to get information or provide input about this research.

6. Why are we doing this research?

We are currently conducting a research study to compare the Smart Fabric Bellyband, a medical device developed in the Drexel Wireless Systems Laboratory (DWSL). The goal of this study is automated detection of abnormalities in the control of breathing.

APPROVED
Human Research Protection
Protocol # ***1504003601-R002***
Approval Date: ***02/19/2018***
Expiration Date: ***02/18/2019***

7. How long will the research last?

Your infant will be monitored with the BellyBand for up to 24 hours, in addition to the standard monitoring the infant will be placed on when he/she is admitted to the Neonatal Intensive Care Uniit.

8. How many people will be studied?

We expect about 20 infants will be enrolled in this research study.

9. What happens if I say yes, I want to be in this research?

1. The baby will wear the Smart Fabric Bellyband around the abdominal area for the duration of the study.
2. The baby will have two devices placed during monitoring: the Smart Fabric Bellyband and traditional infant monitoring devices.
3. This is a one time study, so no follow up will be needed. The baby will be monitored with the Bellyband on for a predetermined length of time not to exceed 24 hours.
4. We would like to take a photo of the baby's abdominal area wearing the band. The photo will only be of the abdomen. The baby will not be able to be identified form the photo.

10. What are my Responsibilities If I Take Part in this Research?

If you take part in this research, it is very important that you:

- Follow the physician's or researcher's instructions.

11. What happens if I do not want to be in this research?

You may decide not to take part in the research and it will not be held against you.

12. What happens if I say yes, but I change my mind later?

You agree to take part in the research now and may stop at any time. It will not be held against you.

13. Do I have to pay for anything while I am on this study?

There is no cost to you for participating in this study.

14. What happens to the information we collect?

Efforts will be made to limit your personal information, including research study and medical records, to people who have a need to review this information. Organizations that may inspect and copy your information include the IRB and other representatives of this organization. The information we collect will not be personally identifiable, and may be used for commercialization, FDA clearance, and academic publications

18. What else do I need to know?

This research is being funded by the National Science Foundation conducted by Drexel University.

Authorization to Use and Disclose Protected Health Information

Federal law provides additional protections of your personal information that are described here.

A. Individually Identifiable Health Information That Will Be Collected

The following personal health information about you will be collected and used during the research study and may be given out to others:

- Your name, address, telephone number, date of birth;
- Information in medical records located in your doctor's office or at other medical facilities you may have received treatment.

B. Who Will See and Use Your Health Information within Drexel University

The researcher and other authorized individuals involved in the research study at Drexel University will see your health information during and may give out your health information during the research study. These include the researcher and the research staff, the institutional review board and their staff, legal counsel, research office and compliance staff, officers of the organization and other people who need to see the information in order to conduct the research study or make sure it is being done properly. Your health information may be disclosed or transmitted electronically.

C. Who Else May See and Use your Health Information

Other persons and organizations outside of Drexel University may see and use your health information during this research study. These include:

- Governmental entities that have the right to see or review your health information, such as The Office for Human Research Protections.
- Doctors and staff at the hospital where this research study will take place.
- The sponsor of this research study and persons that the sponsor may hire to work on the research study. The name of the sponsor is William H. Coulter Endowment.

If your health information is given to someone not required by law to keep it confidential, then that information may no longer be protected, and may be used or given out without your permission.

D. Why your health information will be used and given out

Your health information will be used and given out to carry out the research study and to evaluate the results of the study. Your information may also be used to meet the reporting requirements of governmental agencies.

E. If you do not want to give authorization to use your health information

You do not have to give your authorization to use or give out your health information. However, if you do not give authorization, you cannot participate in this research study.

F. How to cancel your authorization

At any time you may cancel your authorization to allow your health information to be used or given out by sending a written notice to Human Research Protection at 1505 Race Street, 7th floor, Bellet Building, Mail Stop 444, Philadelphia, Pennsylvania, 19102. If you leave this research study, no new

health information about you will be gathered after you leave. However, information gathered before that date may be used or given out if it is needed for the research study or any follow-up.

G. When your authorization ends

Your authorization to use and give out health information will continue until you withdraw or cancel your authorization.

After the research study is finished, your health information will be maintained in a research database. Drexel University shall not re-use or re-disclose the health information in this database for other purposes unless you give written authorization to do so. However, the Drexel University Institutional Review Board may permit other researchers to see and use your health information under adequate privacy safeguards.

H. Your right to inspect your medical and research records

You will not be able to look at your research records while you are taking part in this research study. Your personal information will be made available in an emergency if doctors need this information to treat you. You can have access to your medical record and any research study information when the study is over. However, the researcher does not have to release research information to you if it is not part of your medical record.

Signature Block for Capable Adult

Your signature documents your permission to take part in this research.

DO NOT SIGN THIS FORM AFTER THIS DATE → February 18, 2019

Signature of subject	Date
Printed name of subject	
Signature of person obtaining consent	Date
Printed name of person obtaining consent	Form Date

C.2 IRB Approval for Maternal Uterine Monitoring during Labor and Delivery

Drexel UNIVERSITY
Office of Research

APPROVAL OF PROTOCOL

June 29, 2018

Kapil Dandekar, PhD
Electrical and Computer Engineering
Drexel University

Dear Dr. Dandekar:

On June 28, 2018, the IRB reviewed the following protocol:

Type of Review:	Continuing Review
Title:	Maternity Smart Fabric Bellyband to Monitor Uterine Activity
Investigator:	Dandekar, Kapil R
IRB ID:	1504003602R003
Funding:	Electrical & Computer Engineering (3665) / Wallace H. Coulter Endowment
Grant Title:	Maternity "smart fabric bellyband" to monitor uterine activity and assess fetal well being
Grant ID:	12121489
IND, IDE or HDE:	None
Documents Reviewed:	HRP-212 Continuing Review Progress Report, HRP-503 Protocol, HRP-502 Consent

On June 28, 2018, the IRB approved the protocol renewal. This protocol was reviewed following the Pre-2018 Common Rule. Approval includes enrollment of 20 pregnant women to wear the Smart Fabric Bellyband while in active labor. 0 subjects have been enrolled since the initial approval; 20 remaining to complete the study.

According to 45 CFR 46.110, this study is approved Expedited Categories 1(b) and 4. Risk Level of Device: Non-Significant Risk, According to 21 CFR 812.2(b). According to 45 CFR 46.204, and 45 CFR 46.205, this study involves research not greater than minimal risk.

The IRB approved the protocol from June 29, 2018 to June 28, 2019, inclusive. Before May 15, 2019, which is 45 days prior to study closure, you are to submit a continuing approval or closure request, and required attachments in COEUS.

If continuing review approval is not granted before the expiration date of June 28, 2019, approval of this protocol expires on that date.

Attached are stamped approved consent documents. Use copies of these documents to document consent.

In conducting this protocol you are required to follow the requirements listed in the INVESTIGATOR MANUAL (HRP-103).

Sincerely,

Zachary Hathaway
Member, Institutional Review Board #1

Drexel UNIVERSITY **Consent to Take Part in Human Research Study**

Consent to Take Part
In a Research Study

1. Title of research study: Maternity "SMART FABRIC BELLYBAND" to monitor uterine activity

2. Researchers: Kapil Dandekar, Genevieve Dion, Adam Fontecchio, Timothy Kurzweg, William Mongan, Owen Montgomery

3. Why you are being invited to take part in a research study

We invite you to take part in a research study because you are pregnant, and we want to monitor the uterine contraction.

4. What you should know about a research study

- Someone will explain this research study to you.
- Whether or not you take part is up to you.
- You can choose not to take part.
- You can agree to take part now and change your mind later.
- If you decide to not be a part of this research no one will hold it against you.
- Feel free to ask all the questions you want before you decide.

5. Who can you talk to about this research study?

If you have questions, concerns, or complaints, or think the research has hurt you, talk to the research team at **215-895-0286 (William Mongan).**

This research has been reviewed and approved by an Institutional Review Board (IRB). An IRB reviews research projects so that steps are taken to protect the rights and welfare of human subjects taking part in research. You may talk to them at (215) 762-3944 or email HRPP@drexel.edu for any of the following:

- Your questions, concerns, or complaints are not being answered by the research team.
- You cannot reach the research team.
- You want to talk to someone besides the research team.
- You have questions about your rights as a research subject.
- You want to get information or provide input about this research.

6. Why are we doing this research?

We are currently conducting a research study to compare the Smart Fabric Bellyband, a medical device developed in the Drexel Wireless Systems Laboratory (DWSL), to compare with the Tocodynamometer, which is what doctors use as standard care for fetal monitoring. We want to see if the Smart Fabric Bellyband is able to obtain the same level of information as the Tocodynamometer (Toco).

7. How long will the research last?

We expect that you will be in this research study for about 1 hour.

8. How many people will be studied?

We expect about 20 people here will be in this research study.

9. What happens if I say yes, I want to be in this research?

1. You will still be monitored by the nurses and physicians, and obtaining the standard care. But in addition to that, you will be wearing the Smart Fabric Bellyband around your lower abdominal area for an hour.
2. You will have two devices placed during active labor: the Smart Fabric Bellyband and the Tocodynamometer. The Tocodynamometer is the medical device for the standard of care. It means that without participating in this research, you will still need the 'toco' for fetal monitoring. We will measure your belly to ensure that the correct bellyband size is used.
3. We would like to take a photo of your belly wearing the band. The photo will only be of your belly. You will not be able to be identified from the photo.
4. This is a one time study, so no follow up will be needed. You will have the Bellyband on for one hour.

10. What happens if I do not want to be in this research?

You may decide not to take part in the research and it will not be held against you.

11. What happens if I say yes, but I change my mind later?

You agree to take part in the research now and stop at any time, and it will not be held against you.

12. Do I have to pay for anything while I am on this study?

There is no cost to you for participating in this study.

13. What happens to the information we collect?

Efforts will be made to limit your personal information, including research study and medical records, to people who have a need to review this information. We cannot promise complete secrecy. Organizations that may inspect and copy your information include the IRB and other representatives of this organization. The information we collect will not be personally identifiable, and may be used for commercialization, FDA clearance, and academic publications.

18. What else do I need to know?

This research is being funded by the National Science Foundation.
This research study is being done by Drexel University.
You will receive a $50 gift card for participating in this study.

Signature Block for Capable Adult

Your signature documents your permission to take part in this research.

DO NOT SIGN THIS FORM AFTER THIS DATE → 06/28/2019

Signature of subject	Date
Printed name of subject	
Signature of person obtaining consent	Date
Printed name of person obtaining consent	

Subject's Initials_____

C.3 IRB Approval for Cardiorespiratory Monitoring of Adults

Drexel UNIVERSITY
Office of Research

APPROVAL OF PROTOCOL

May 21, 2018

Timothy Kurzweg, PhD

Dear Dr. Kurzweg:

On 05/21/2018, the IRB reviewed the following protocol:

Type of Review:	Continuing
Title:	Passive RFID Based Heart Rate Monitor to Wirelessly Monitor Heart Rate
Investigator:	Timothy Kurzweg, PhD
IRB ID:	1604004440-R002
Funding:	PFI: BIC Wearable Smart Textiles – (Coeus 5431) (Grant #2359163665)
Grant Title:	None
Grant ID:	None
IND, IDE or HDE:	None
Documents Reviewed:	HRP 212 Continuing Review Form, HRP 503 Protocol, HRP 502 Adult Informed Consent/Permission Consent, and Recruitment Advertisement

On 05/21/2018 – Approved Renewal. Renewal Includes: 25 Approved; 6 enrolled; 19 remaining to complete the study.

The IRB approved the protocol from 06/21/2018 to 06/21/2019 inclusive.

Risks to Children: 45 CFR 46.404 Minimal Risk

Permission of one parent is sufficient even if the other parent is alive, known, competent, reasonably available, and shares legal responsibility for the care and custody of the child. Assent for children not obtain: The capability of these children is so limited that they cannot reasonably be consulted.

Before 05/06/2019, which is 45 days prior to study closure, you are to submit a completed "FORM: Continuing Review Progress Report (HRP-212)" and required attachments to request continuing approval or closure.

Drexel UNIVERSITY
Office of Research

If continuing review approval is not granted before the expiration date of 06/20/2019 approval of this protocol expires on that date.

Attached is a stamped approved consent document. Use copies of this document to document consent.

In conducting this protocol you are required to follow the requirements listed in the INVESTIGATOR MANUAL (HRP-103).

Sincerely,

Adithi Asuri
Member, IRB #1

Drexel University
Consent to Take Part
In a Research Study
At Hahnemann Hospital

If you are a parent or legal guardian of a child who may take part in this study, permission from you is required. The assent (agreement) of your child may be required. When we say "you" in this consent form, we mean you or your child; when we say "we" in this consent form, we mean Drexel University.

1. Title of research study: "PASSIVE RFID BASED HEART RATE MONITOR" to wirelessly monitor heart rate. (1604004440)

2. Researcher: Timothy Kurzweg

3.*Concise Summary of Key Informtion*:

Globally, cardiovascular diseases continue to remain the leading cause of death; killing more people than any other disease. This alone makes a strong case for development of convenient continuous cardiac monitors. Additionally, heart rate monitors have also started finding applications in diverse areas from optimizing athletic performance to monitoring health for those employed in professions where real time health monitoring may help mitigate risks (e.g. firefighters). Several technologies have been utilized for heart rate monitoring. These include, but are not limited to, ballistocardiography, pulse oximetry, phonocardiogram and electrocardiogram.

Over the last decade, there has been a big push towards the development of wearable health monitoring devices. The underlying belief is that wearable devices will allow for affordable round-the-clock monitoring which will enable early detection and prevention of many diseases. Heart rate monitors and electrocardiogram (ECG) devices are a group of devices that have received wide attention from the wearable devices community. However, research on ECG monitors has been focused on making better wearable sensors and integrating them with a minimal profile. To wirelessly transmit the cardiac information, researchers have largely continued using battery-powered methods like bluetooth and cellular technologies or prototype RF transmitters. Batteries add to the size and weight of wearable systems, making them obtrusive and cumbersome. A wearable system benefits from being battery-free as the absence of batteries reduces the system down time for battery replacement and charging.

Passive RFID (Radio Frequency Identification) tags work on wireless power harvested from a RFID reader and thus do not require batteries for operation. RFID tags have been conventionally used as product identification tools similar to barcodes. For these tags to be used in sensor networks, they have to be capable of transmitting sensor data along with the tag ID. In its simplest form, an RFID tag can be used as a one bit transmission device by turning the RFID tag on/off and having an RFID reader detect the tag's state. For instance, the reader detects a `1' when the tag is on and a `0' when the tag is off. A similar device that uses two RFID tags has been proposed in literature. Contemporary RFID tags are capable of transmitting multiple bits of optional data along with the default tag ID. However, additional circuitry like analog to digital converters (ADCs) and microcontrollers are required to digitize and embed the sensor data with the tag ID.

These additional components not only increase the size of the system but also add to its power requirements. Another drawback of such systems is the requirement of significant transmitted data redundancy to achieve a degree of reliability. For example, the electroencephalogram (EEG) system proposed in literature requires 92% data overhead and has a range of only 0.8m. The requirement for data overhead adds to the system power consumption. If this power is harvested wirelessly, higher power demand significantly degrades the system's range. The proposed RFID heart monitoring device has been designed by leveraging battery-free transmission capabilities of RFID tags while using minimal additional components and not requiring any data overheads. This device can potentially be used to wirelessly track a person's heart rate and location while he is mobile and could potentially limit the need for a patient to be tethered to a cardiac monitor on the hospital bed.

This research is being performed in order to develop a less invasive wearable heart/respiratory monitoring system that requires no external electrical power to operate. Consent is being sought for this research is voluntary. If you consent to the study, you will participate for approximately 30 minutes, during which a pair of ECG electrodes will be attached to the subject's chest to monitor heart rate via the wireless heart rate monitor. A smart-fabric Bellyband will be worn about the abdominal area for respiratory rate monitoring. Normal respiratory and heart activities will be monitored for approximately 30 minutes, and then the electrodes and Bellyband will be removed. There is no direct benefit from this study as traditional monitors will be used as needed outside of this research study; however, the data collected will be used to develop wireless, wearable monitoring technologies. In order to mitigate the radio-frequency exposure of the subject, and to comply with the FCC standard, we will utilize an off-the-shelf RFID interrogator installed at a proper reading distance. That is, the distance between the RFID interrogator and the bellyband will be at least 25 cm. In this way, we will be able to read the bellyband data and, at the same time, to reduce the SAR (Specific Absorption Rate). This minimum distance eliminates any radio-frequency exposure risk and complies with the FCC requirements for RFID technologies. There is no cost to participate in this study, but you will receive a $25 gift card in return for participation.

4. Why you are being invited to take part in a research study

We invite you to take part in a research study because you are a healthy adult willing to place standard ECG leads on your torso and/or a Bellyband wearable fabric.

5. What you should know about a research study

- Someone will explain this research study to you.
- Whether or not you take part is up to you.
- You can choose not to take part.
- You can agree to take part now and change your mind later.
- If you decide to not be a part of this research no one will hold it against you.
- Feel free to ask all the questions you want before you decide.

6. Who can you talk to about this research study?

If you have questions, concerns, or complaints, or think the research has hurt you, talk to the research team at ***William Mongan: 215-895-0286***

This research has been reviewed and approved by an Institutional Review Board (IRB). An IRB reviews research projects so that steps are taken to protect the rights and welfare of human subjects taking part in research. You may talk to them at (215) 762-3944 or email HRPP@drexel.edu for any of the following:

- Your questions, concerns, or complaints are not being answered by the research team.
- You cannot reach the research team.
- You want to talk to someone besides the research team.
- You have questions about your rights as a research subject.
- You want to get information or provide input about this research.

7. Why are we doing this research?

We are currently conducting a research study to evaluate the practical feasibility of a wireless and battery free heart rate and respiratory monitor based on RFID technology.

8. How long will the research last?

We expect that you will be in this research study for approximately 30 minutes for your heart and respiratory rate in addition to about fifteen minutes of set up time.

9. How many people will be studied?

We expect about 25 people here will be in this research study out of 25 people in the entire study nationally.

10. What happens if I say yes, I want to be in this research?

1. The research will be conducted at the Bossone Research Center of Drexel University.
2. You will have a pair of ECG electrodes attached to your chest to monitor your heart rate using the wireless heart rate monitor. In the hospital setting, these will be deployed by hospital staff.
3. A smart-fabric Bellyband will be worn about your abdominal area for respiratory rate monitoring. In the hospital setting, these will be deployed by hospital staff.
4. This is a one time study, so no follow up will be needed. You will be monitored for a predetermined time not exceeding 10 minutes.
5. We would like to take a photo of the truncal area with the monitor. The photo will only be of the truncal area, to include the ECG electrodes and Bellyband, and you will not be able to be identified form the photo.
6. Standard disposable commercial electrodes will be used to sense ECG signals for hygiene and to minimize the risk of skin irritation. The heart rate monitoring device is a very low power device (tens of microwatts) and thus poses no risk of electrical injury to the subject.
7. In order to mitigate the radio-frequency exposure of the subject, and to comply with the FCC standard, we will utilize an off-the-shelf RFID interrogator installed at a proper reading distance. That is, the distance between the RFID interrogator and the bellyband will be at least 25 cm. In this way, we will be able to read the bellyband data and, at the same time, to reduce the SAR (Specific Absorption Rate). This minimum distance eliminates any radio-frequency exposure risk and complies with the FCC requirements for RFID technologies.

11. What are my responsibilities if I take part in this research?

If you take part in this research, it is very important that you:

- Follow your physician's or researcher's instructions.
- Tell your study physician or researcher right away if you have a complication or injury.

12. What happens if I do not want to be in this research?

You may decide not to take part in the research and it will not be held against you.

13. What happens if I say yes, but I change my mind later?

You agree to take part in the research now and stop at any time it will not be held against you.

If you stop being in the research, already collected data may not be removed from the study database. You will be asked whether the researcher can collect data from your routine medical care. If you agree, this data will be handled the same as research data.

15. Do I have to pay for anything while I am on this study?

There is no cost to you for participating in this study.

16. Will being in this study help me any way?

There are no benefits to you from your taking part in this research. We cannot promise any benefits to others from your taking part in this research. However, possible benefits to others include a less invasive wearable continuous heart and respiratory monitor.

17. What happens to the information we collect?

Efforts will be made to limit your personal information, including research study and medical records, to people who have a need to review this information. We cannot promise complete secrecy. Organizations that may inspect and copy your information include the IRB and other representatives of this organization.

The information we collect will not be personally identifiable, and may be used for commercialization, FDA clearance, and academic publications.

19. What else do I need to know?

The sponsor National Science Foundation is paying Drexel University to conduct the study.

It is important for you to follow your physician's instructions including notifying your study physician as soon as you are able of any complication or injuries that you experienced.

If you agree to take part in this research study, we will pay you a $25 gift card for your time and effort.

Federal law provides additional protections of your personal information that are described here.

A. Individually Identifiable Health Information That Will Be Collected

The following personal health information about you will be collected and used during the research study and may be given out to others:

- Your name;
- Information from laboratory tests, blood and urine tests, x-rays, physical exams and other tests or procedures described in this consent form.

B. Who Will See and Use Your Health Information within Drexel University

The researcher and other authorized individuals involved in the research study at Drexel University will see your health information during and may give out your health information during the research study. These include the researcher and the research staff, the institutional review board and their staff, legal counsel, research office and compliance staff, officers of the organization and other people who need to see the information in order to conduct the research study or make sure it is being done properly. Your health information may be disclosed or transmitted electronically.

C. Who Else May See and Use your Health Information

Other persons and organizations outside of Drexel University may see and use your health information during this research study. These include:

- Governmental entities that have the right to see or review your health information, such as The Office for Human Research Protections, and the Food and Drug Administration
- Doctors and staff at the hospital where this research study will take place.
- Doctors and staff at other places that are participating in the research study.
- The sponsor of this research study and persons that the sponsor may hire to work on the research study. The name of the sponsor is National Science Foundation
- A data safety monitoring board.

If your health information is given to someone not required by law to keep it confidential, then that information may no longer be protected, and may be used or given out without your permission.

D. Why your health information will be used and given out

Your health information will be used and given out to carry out the research study and to evaluate the results of the study. Your health information will also be used so the sponsor may receive FDA approval for a new product or drug resulting from this research study.

Your information may also be used to meet the reporting requirements of governmental agencies.

E. If you do not want to give authorization to use your health information

You do not have to give your authorization to use or give out your health information. However, if you do not give authorization, you cannot participate in this research study.

F. How to cancel your authorization

At any time you may cancel your authorization to allow your health information to be used or given out by sending a written notice to Human Research Protection at 1505 Race Street, 7th floor, Bellet Building, Mail Stop 444, Philadelphia, Pennsylvania, 19102. If you leave this research study, no new health information about you will be gathered after you leave. However, information gathered before

that date may be used or given out if it is needed for the research study or any follow-up.

G. When your authorization ends

Your authorization to use and give out your health information will end when the research study is finished.

After the research study is finished, your health information will be maintained in a research database. Drexel University shall not re-use or re-disclose the health information in this database for other purposes unless you give written authorization to do so. However, the Drexel University Institutional Review Board may permit other researchers to see and use your health information under adequate privacy safeguards.

H. Your right to inspect your medical and research records

You have the right to look at your medical records at any time during this research study. However, the researcher does not have to release research information to you if it is not part of your medical record.

I. Information about Genetic Information Nondiscrimination Act (GINA)

The Federal law, called the Genetic Information Nondiscrimination Act (GINA), generally makes it illegal for health insurance companies, group health plans, and most employers to discriminate against you based on your genetic information. This law generally will protect you in the following ways:

- Health insurance companies and group health plans may not request your genetic information that we get from this research study.
- Health insurance companies and group health plans may not use your genetic information when making decisions regarding your eligibility or premiums.
- Employers with 15 or more employees may not use your genetic information that we get from this research study when making a decision to hire, promote, or fire you or when setting the terms of your employment.

However, GINA will not protect you if you already have a genetic disease or disorder and does not prohibit discrimination on the basis of an existing genetic disease or disorder. In addition, this new Federal law does not protect you against genetic discrimination by companies that sell life insurance, disability insurance, or long-term care insurance.

Signature Block for Capable Adult

Your signature documents your permission to take part in this research.

DO NOT SIGN THIS FORM AFTER THIS DATE → 06/20/2019

Signature of subject	Date
Printed name of subject	
Signature of person obtaining consent	Date
Printed name of person obtaining consent	

Signature Block for Adult Unable to Consent

DO NOT SIGN THIS FORM AFTER THIS DATE →

Printed name of subject

I am willing to serve as a legally authorized representative for the above named subject. The investigators have explained to me the role and responsibilities of a legally authorized representative. My signature documents my permission for the above named subject to take part in this research.

Signature of legally authorized representative | Date

Printed name of legally authorized representative

Address

City, State, ZIP

Phone | Email

Highest	The following individuals in descending order of priority are capable of serving as a legally authorized representative (LAR). Check the category that best describes the LAR's relationship to the subject.
☐	Health care agent appointed by the subject in a Power of Attorney;
☐	Court-appointed guardian authorized to consent to the subject's participation in the protocol in a current court order issued within the subject's jurisdiction;
☐	Spouse or domestic partner (unless an action for divorce is pending) and adult children of the subject who are not the children of the spouse or domestic partner;
☐	Adult child;
☐	Natural or adoptive parent;
☐	Adult brother or sister.
☐	Adult grandchild
☐ Lowest	Adult who has knowledge of the subject's preferences and values, including, but not limited to, religious and moral beliefs, to assess how the principal would make health care decisions. Unless related by blood, marriage, or adoption, the adult may not be the principal's attending physician or other health care provider nor an owner, operator or employee of a health care provider in which the principal receives care.

Signature of person obtaining consent | Date

Printed name of person obtaining consent

Drexel UNIVERSITY **Authorization to Use and Disclose Protected Health Information**

Drexel University
Consent to Take Part
In a Research Study
At Hahnemann Hospital

If you are a parent or legal guardian of a child who may take part in this study, permission from you is required. The assent (agreement) of your child may be required. When we say "you" in this consent form , we mean you or your child; when we say "we" in this consent form, we mean Drexel University.

1. Title of research study: "PASSIVE RFID BASED HEART RATE MONITOR" to wirelessly monitor heart rate. (1604004440)

2. Researcher: Timothy Kurzweg

3.*Concise Summary of Key Information*:

Globally, cardiovascular diseases continue to remain the leading cause of death; killing more people than any other disease. This alone makes a strong case for development of convenient continuous cardiac monitors. Additionally, heart rate monitors have also started finding applications in diverse areas from optimizing athletic performance to monitoring health for those employed in professions where real time health monitoring may help mitigate risks (e.g. firefighters). Several technologies have been utilized for heart rate monitoring. These include, but are not limited to, ballistocardiography, pulse oximetry, phonocardiogram and electrocardiogram.

Over the last decade, there has been a big push towards the development of wearable health monitoring devices. The underlying belief is that wearable devices will allow for affordable round-the-clock monitoring which will enable early detection and prevention of many diseases. Heart rate monitors and electrocardiogram (ECG) devices are a group of devices that have received wide attention from the wearable devices community. However, research on ECG monitors has been focused on making better wearable sensors and integrating them with a minimal profile. To wirelessly transmit the cardiac information, researchers have largely continued using battery-powered methods like bluetooth and cellular technologies or prototype RF transmitters. Batteries add to the size and weight of wearable systems, making them obtrusive and cumbersome. A wearable system benefits from being battery-free as the absence of batteries reduces the system down time for battery replacement and charging.

Passive RFID (Radio Frequency Identification) tags work on wireless power harvested from a RFID reader and thus do not require batteries for operation. RFID tags have been conventionally used as product identification tools similar to barcodes. For these tags to be used in sensor networks, they have to be capable of transmitting sensor data along

with the tag ID. In its simplest form, an RFID tag can be used as a one bit transmission device by turning the RFID tag on/off and having an RFID reader detect the tag's state. For instance, the reader detects a `1' when the tag is on and a `0' when the tag is off. A similar device that uses two RFID tags has been proposed in literature. Contemporary RFID tags are capable of transmitting multiple bits of optional data along with the default tag ID. However, additional circuitry like analog to digital converters (ADCs) and microcontrollers are required to digitize and embed the sensor data with the tag ID. These additional components not only increase the size of the system but also add to its power requirements. Another drawback of such systems is the requirement of significant transmitted data redundancy to achieve a degree of reliability. For example, the electroencephalogram (EEG) system proposed in literature requires 92% data overhead and has a range of only 0.8m. The requirement for data overhead adds to the system power consumption. If this power is harvested wirelessly, higher power demand significantly degrades the system's range. The proposed RFID heart monitoring device has been designed by leveraging battery-free transmission capabilities of RFID tags while using minimal additional components and not requiring any data overheads. This device can potentially be used to wirelessly track a person's heart rate and location while he is mobile and could potentially limit the need for a patient to be tethered to a cardiac monitor on the hospital bed.

This research is being performed in order to develop a less invasive wearable heart/respiratory monitoring system that requires no external electrical power to operate. Consent is being sought for this research is voluntary. If you consent to the study, you will participate for approximately 30 minutes, during which a pair of ECG electrodes will be attached to the subject's chest to monitor heart rate via the wireless heart rate monitor. A smart-fabric Bellyband will be worn about the abdominal area for respiratory rate monitoring. Normal respiratory and heart activities will be monitored for approximately 30 minutes, and then the electrodes and Bellyband will be removed. There is no direct benefit from this study as traditional monitors will be used as needed outside of this research study; however, the data collected will be used to develop wireless, wearable monitoring technologies. In order to mitigate the radio-frequency exposure of the subject, and to comply with the FCC standard, we will utilize an off-the-shelf RFID interrogator installed at a proper reading distance. That is, the distance between the RFID interrogator and the bellyband will be at least 25 cm. In this way, we will be able to read the bellyband data and, at the same time, to reduce the SAR (Specific Absorption Rate). This minimum distance eliminates any radio-frequency exposure risk and complies with the FCC requirements for RFID technologies. There is no cost to participate in this study, but you will receive a $25 gift card in return for participation.

4. Why you are being invited to take part in a research study

We invite you to take part in a research study because you have a healthy infant and are willing to place standard ECG leads on his/her torso and/or a Bellyband wearable fabric.

5. What you should know about a research study

- Someone will explain this research study to you.
- Whether or not you take part is up to you.
- You can choose not to take part.
- You can agree to take part now and change your mind later.
- If you decide to not be a part of this research no one will hold it against you.
- Feel free to ask all the questions you want before you decide.

6. Who can you talk to about this research study?

If you have questions, concerns, or complaints, or think the research has hurt you, talk to the research team at ***William Mongan: 215-895-0286***

This research has been reviewed and approved by an Institutional Review Board (IRB). An IRB reviews research projects so that steps are taken to protect the rights and welfare of human subjects taking part in research. You may talk to them at (215) 762-3944 or email HRPP@drexel.edu for any of the following:

- Your questions, concerns, or complaints are not being answered by the research team.
- You cannot reach the research team.
- You want to talk to someone besides the research team.
- You have questions about your rights as a research subject.
- You want to get information or provide input about this research.

7. Why are we doing this research?

We are currently conducting a research study to evaluate the practical feasibility of a wireless and battery free heart rate and respiratory monitor based on RFID technology.

8. How long will the research last?

We expect that you will be in this research study for approximately 30 minutes for your heart and respiratory rate in addition to about fifteen minutes of set up time.

9. How many people will be studied?

We expect about 25 people here will be in this research study out of 25 people in the entire study nationally.

10. What happens if I say yes, I want to be in this research?

1. The research will be conducted at Hahnemann Hospital.
2. You will have a pair of ECG electrodes attached to your chest to monitor your heart rate using the wireless heart rate monitor. In the hospital setting, these will be deployed by hospital staff.
3. A smart-fabric Bellyband will be worn about your abdominal area for respiratory rate monitoring. In the hospital setting, these will be deployed by hospital staff.
4. This is a one time study, so no follow up will be needed. You will be monitored for a predetermined time not exceeding 10 minutes.

5. We would like to take a photo of the truncal area with the monitor. The photo will only be of the truncal area, to include the ECG electrodes and Bellyband, and you will not be able to be identified form the photo.
6. Standard disposable commercial electrodes will be used to sense ECG signals for hygiene and to minimize the risk of skin irritation. The heart rate monitoring device is a very low power device (tens of microwatts) and thus poses no risk of electrical injury to the subject.
7. In order to mitigate the radio-frequency exposure of the subject, and to comply with the FCC standard, we will utilize an off-the-shelf RFID interrogator installed at a proper reading distance. That is, the distance between the RFID interrogator and the bellyband will be at least 25 cm. In this way, we will be able to read the bellyband data and, at the same time, to reduce the SAR (Specific Absorption Rate). This minimum distance eliminates any radio-frequency exposure risk and complies with the FCC requirements for RFID technologies.

11. What are my responsibilities if I take part in this research?

If you take part in this research, it is very important that you:

- Follow your physician's or researcher's instructions.
- Tell your study physician or researcher right away if you have a complication or injury.

12. What happens if I do not want to be in this research?

You may decide not to take part in the research and it will not be held against you.

13. What happens if I say yes, but I change my mind later?

You agree to take part in the research now and stop at any time it will not be held against you.

If you stop being in the research, already collected data may not be removed from the study database. You will be asked whether the researcher can collect data from your routine medical care. If you agree, this data will be handled the same as research data.

15. Do I have to pay for anything while I am on this study?

There is no cost to you for participating in this study.

16. Will being in this study help me any way?

There are no benefits to you from your taking part in this research. We cannot promise any benefits to others from your taking part in this research. However, possible benefits to others include a less invasive wearable continuous heart and respiratory monitor.

17. What happens to the information we collect?

Efforts will be made to limit your personal information, including research study and medical records, to people who have a need to review this information. We cannot promise complete secrecy. Organizations that may inspect and copy your information include the IRB and other representatives of this organization.

The information we collect will not be personally identifiable, and may be used for commercialization, FDA clearance, and academic publications.

19. What else do I need to know?

The sponsor National Science Foundation is paying Drexel University to conduct the study.

It is important for you to follow your physician's instructions including notifying your study physician as soon as you are able of any complication or injuries that you experienced.

If you agree to take part in this research study, we will pay you a $25 gift card for your time and effort.

Federal law provides additional protections of your personal information that are described here.

A. Individually Identifiable Health Information That Will Be Collected

The following personal health information about you will be collected and used during the research study and may be given out to others:

- Your name;
- Information from laboratory tests, blood and urine tests, x-rays, physical exams and other tests or procedures described in this consent form.

B. Who Will See and Use Your Health Information within Drexel University

The researcher and other authorized individuals involved in the research study at Drexel University will see your health information during and may give out your health information during the research study. These include the researcher and the research staff, the institutional review board and their staff, legal counsel, research office and compliance staff, officers of the organization and other people who need to see the information in order to conduct the research study or make sure it is being done properly. Your health information may be disclosed or transmitted electronically.

C. Who Else May See and Use your Health Information

Other persons and organizations outside of Drexel University may see and use your health information during this research study. These include:

- Governmental entities that have the right to see or review your health information, such as The Office for Human Research Protections, and the Food and Drug Administration
- Doctors and staff at the hospital where this research study will take place.
- Doctors and staff at other places that are participating in the research study.
- The sponsor of this research study and persons that the sponsor may hire to work on the research study. The name of the sponsor is National Science Foundation
- A data safety monitoring board.

If your health information is given to someone not required by law to keep it confidential, then that information may no longer be protected, and may be used or given out without your permission.

D. Why your health information will be used and given out

Your health information will be used and given out to carry out the research study and to evaluate the results of the study. Your health information will also be used so the sponsor may receive FDA approval for a new product or drug resulting from this research study.

Your information may also be used to meet the reporting requirements of governmental agencies.

E. If you do not want to give authorization to use your health information

You do not have to give your authorization to use or give out your health information. However, if you do not give authorization, you cannot participate in this research study.

F. How to cancel your authorization

At any time you may cancel your authorization to allow your health information to be used or given out by sending a written notice to Human Research Protection at 1505 Race Street, 7th floor, Bellet Building, Mail Stop 444, Philadelphia, Pennsylvania, 19102. If you leave this research study, no new health information about you will be gathered after you leave. However, information gathered before that date may be used or given out if it is needed for the research study or any follow-up.

G. When your authorization ends

Your authorization to use and give out your health information will end when the research study is finished.

After the research study is finished, your health information will be maintained in a research database. Drexel University shall not re-use or re-disclose the health information in this database for other purposes unless you give written authorization to do so. However, the Drexel University Institutional Review Board may permit other researchers to see and use your health information under adequate privacy safeguards.

H. Your right to inspect your medical and research records

You have the right to look at your medical records at any time during this research study. However, the researcher does not have to release research information to you if it is not part of your medical record.

I. Information about Genetic Information Nondiscrimination Act (GINA)

The Federal law, called the Genetic Information Nondiscrimination Act (GINA), generally makes it illegal for health insurance companies, group health plans, and most employers to discriminate against you based on your genetic information. This law generally will protect you in the following ways:

- Health insurance companies and group health plans may not request your genetic information that we get from this research study.
- Health insurance companies and group health plans may not use your genetic information when making decisions regarding your eligibility or premiums.
- Employers with 15 or more employees may not use your genetic information that we get from this research study when making a decision to hire, promote, or fire you or when setting the terms of your employment.

Authorization to Use and Disclose Protected Health Information

However, GINA will not protect you if you already have a genetic disease or disorder and does not prohibit discrimination on the basis of an existing genetic disease or disorder. In addition, this new Federal law does not protect you against genetic discrimination by companies that sell life insurance, disability insurance, or long-term care insurance.

Signature Block for Children

Your signature documents your permission for the named child to take part in this research.

DO NOT SIGN THIS FORM AFTER THIS → 06/20/2019

Printed name of child

Signature of parent or individual legally authorized to consent to the child's general medical care

Date

Printed name of parent or individual legally authorized to consent to the child's general medical care

- ❑ Parent
- ❑ Individual legally authorized to consent to the child's general medical care (See note below)

Note: Investigators are to ensure that individuals who are not parents can demonstrate their legal authority to consent to the child's general medical care. Contact legal counsel if any questions arise.

Signature of parent

Date

Printed name of parent

If signature of second parent not obtained, indicate why: (select one)

- ❑ The IRB determined that the permission of one parent is sufficient.
- ❑ Second parent is deceased
- ❑ Second parent is unknown
- ❑ Second parent is incompetent
- ❑ Second parent is not reasonably available
- ❑ Only one parent has legal responsibility for the care and custody of the child

Signature of person obtaining consent and assent

Date

Printed name of person obtaining consent

Drexel University
Recruiting Volunteers for a Research Study

Research Title
"PASSIVE RFID BASED HEART RATE MONITOR" to wirelessly monitor heart rate.

Research Objectives*

Wireless and battery-free heart rate monitors can allow for continuous uninterrupted monitoring for a variety of applications for clinical and fitness applications. Researchers at Drexel University Electrical and Computer Engineering Department are conducting a research study to evaluate the practical feasibility of a wireless and battery free heart rate, muscle contraction and respiratory rate monitor based on RFID technology. For each participant, the study involves collecting a simple electro-cardiogram (ECG) signal as well as wearing a wireless RFID-based respiratory monitoring device, and takes less than a total of 30 minutes to complete.

Information for Research Subjects Eligibility

You must be at least 18 years old to participate in this study.

Remuneration

You will be given a $25 gift card for participation in this study.

Location of the research and person to contact for further information

This research is approved by the Institutional review board.
If you are interested in participating in this study, please contact

William Mongan
Telephone number: **215-895-0286** Room: **University Crossings 100D**

(This research is conducted by a researcher who is a member of Drexel University or Drexel University College of Medicine).

APPROVED
Human Research Protection
Protocol # ***1604004440R002***
Approval Date: ***06/21/2018***
Expiration Date: ***06/20/2019***

Vita

William M. Mongan, Jr.

CONTACT INFORMATION

William Mongan
Philadelphia, Pennsylvania USA

Phone: (484) 442-0821
E-Mail: bill@billmongan.com
WWW: billmongan.com
LinkedIn: www.linkedin.com/in/billmongan

EDUCATION

Doctor of Philosophy — Drexel University
PhD in Electrical and Computer Engineering
Philadelphia, Pennsylvania — 2013-2018

- PhD Dissertation: *Predictive Analytics on Real-Time Biofeedback for Actionable Classification of Activity State*
- PhD Advisor: Dr. Adam Fontecchio
- Cumulative Graduate GPA: 3.91 on a 4.00 index

Master of Science — Drexel University
Computer Science, College of Engineering
Philadelphia, Pennsylvania — 2005-2008

- MS Thesis in Computer Science: *A Service-Based Web Portal for Integrated Reverse Engineering and Program Comprehension*
- MS Thesis Advisor: Dr. Spiros Mancoridis
- Graduate Research Assistant for the College of Engineering under the Applied Communications and Information Networking (ACIN) Project, and for the Department of Computer Science under the Software Engineering Research Group (SERG)
- GPA: 3.86 on a 4.00 index

Master of Science — Drexel University
Science of Instruction, School of Education
PA Instructional Certificate in Mathematics
Philadelphia, Pennsylvania — 2005-2008

- Earned a Secondary Mathematics (Pennsylvania In-State: Mathematics Grades 7-12) teaching certification in Pennsylvania
- GPA: 3.86 on a 4.00 index
- Served as a graduate fellow for the educational outreach program in the NSF GK-12 Program at the School District of Philadelphia at two schools and grades 5-8 (2006-2008)

Bachelor of Science, Magna Cum Laude — Drexel University
Computer Science, College of Engineering
Philadelphia, Pennsylvania — 2000-2005

- Undergraduate Research Assistant
- GPA: 3.79 on a 4.00 index
- Minor in Mathematics, College of Arts and Sciences
- Teaching Assistant for Systems Architecture I and II (CS281, CS282): 2003-2005

SELECTED FACULTY APPOINTMENTS AND TEACHING

Teaching Professor (full) — Drexel University
Computer Science
Philadelphia, Pennsylvania — 2017-Present

- Member of the Provost's DC Programming Summit, 2018-Present
- Delaware County Intermediate Unit (DCIU) STEM Education Council, 2017-Present
- Member of the Steinbright Faculty Advisory Board, 2017-Present
- Member of the CCI Undergraduate Data Science Curriculum Committee, 2016-Present
- Member of the Isaac L. Auerbach Innovation Award Committee, 2017-2018
- Member of the Faculty Data and Assessment Committee, Chair of the Survey Baseline Data Group Subcommittee 2017-2018

- Member of University Advising Implementation Committee Workgroup on Student Major Changes across the University, 2017.
- Member of the Faculty Senate Budget Planning and Development (BP&D) Committee, 2017-2018
- Cohort participant in Drexel Leaders 2020 professional development program, 2017
- Member of the University Program Alignment and Review (PAR) Committee, 2016-2019
- Member of the University Advising Implementation Committee (UAIC), 2016-Present
- Member of the Center for the Advancement of STEM Teaching and Learning Excellence (CASTLE) network at Drexel University, 2016-Present

Associate Department Head for Undergraduate Affairs Drexel University
Computer Science
Philadelphia, Pennsylvania 2015-Present

- Chair of the CCI Undergraduate Computing Curriculum Committee
- Chair of the CCI Program Alignment and Review (PAR) Committee 2015-2016
- Acting Department Head August, 2015-January, 2016
- Member of the CCI Faculty/Academic Council
- Member of the CCI College Curriculum Committee, co-Chair 2016-2017
- Member of CCI Computing Assessment and Accreditation Committee, Chair 2012-2016
- Program Assessment Coordinator for three programs within the Department of Computer Science and one program within the Software Engineering Department

Associate Teaching Professor Drexel University
Computer Science
Philadelphia, Pennsylvania 2012-2017

- Member of the Dean Search Committee for the College of Nursing and Health Professions (CNHP), representing the Faculty Senate, 2016-2017
- Member of the Philly Codefest Advisory Board, 2016-2017
- Member of the University Advising Steering Committee and Chair of Advising Infrastructure and Workflow Subcommittee, 2016-2017
- Director of Undergraduate Affairs for the Department of Computer Science, 2014-2015
- University Faculty Senator from CCI, 2014-2017
- Recording Secretary for the CCI Senate Caucus
- Member of the CCI Faculty Search Committee 2013-2015
- Member of the First-Year CCI Curriculum Committee
- Member of the CCI Computing Faculty Resources Committee, 2014-2016
- Member of the College of Engineering (COE) Assessment Executive Committee

Lecturer (Visiting) University of Pennsylvania
Computer and Information Science
School of Engineering and Applied Science 2012-2013

Assistant Teaching Professor Drexel University
Computer Science
Electrical and Computer Engineering
Philadelphia, Pennsylvania 2011-2012

- Chair of the CS Department Assessment Committee
- Member of the COE Assessment Committee

Instructor / Auxiliary Professor Drexel University
Computer Science
Electrical and Computer Engineering
Philadelphia, Pennsylvania 2008-2011

- Support external course projects in Circuit Design and Operating System Internals

- Chair of the CS Undergraduate Retention / Core Curriculum Committee
- Member of the CS Undergraduate Curriculum Committee
- Instructor for the Drexel University Computing Academy (DUCA) High School Summer Program: (formerly the Pennsylvania Governor's School in Information, Society and Technology)
- Developed a new track and curriculum for the CS Architecture Track concentration

Courses Taught

- Independent Study Advisor: RFID Multisensor Fusion (CSI499)
- Independent Study Advisor: Forecasting on Medical Data (CSI499, CSI199)
- Guest Lecturer for Collaborative Intelligent Radio Networks (CIRN) Design (ECE T680 Graduate Course)
- Independent Study Advisor: Biomedical Signal Processing (CSI499)
- Independent Study Advisor: Advanced Web Services and Mobile Application Development (CSI499)
- Independent Study co-Advisor: Social Media Analytics (CS498)
- Instructor and co-Author for Computing and Informatics Design Project (CI106, CS280)
- Instructor and co-Author for Computing and Informatics Design I (CI101)
- Independent Study Advisor: Computing and Informatics Design Experience (CS498)
- Independent Study Advisor: Big Data and Advanced Analytics (CS498)
- Instructor and Author for "Making Apps" Honors course for the university (HNRS202)
- Instructor and Author for Processor Architecture and Analysis (CS352, CS480)
- Independent Study Advisor: Advanced Topics in Operating Systems (CS498)
- Instructor for Computer Systems II: Digital Systems Organization and Design (CIT595 Graduate Course, Visiting at the University of Pennsylvania)
- Instructor and Author for Web and Mobile Application Development (CS275, CS280, CS480, CS690)
- Instructor for ESL097: the King Abdullah University of Science and Technology Introduction to C++ course for high school seniors for the Drexel University English Language Center (joint grant between Drexel University and the University of Pennsylvania)
- Guest Lecturer for Operating Systems (CS543 Graduate Course)
- Guest Lecturer for Computer Networks (CS544 Graduate Course)
- Instructor for Systems Architecture I and II (CS281, CS282, CS498)
- Instructor for Computer Organization (ECEC490)
- Instructor and Author for Systems Programming (CS283, ECEC353)
- Instructor for Concurrent Programming (CS361, CS498, ECEC490)
- Instructor for Processor Design (ECEC490)
- Instructor and Author for Operating Systems (CS370, ECEC421, ECEC490)
- Instructor for Computer Networks (CS472, ECEC357, ECEC490)
- Instructor for Computer Structures (ECEC355)
- Instructor and Author for Machine Organization (CS680 Graduate Course)
- Instructor for Internet Arch and Protocols (ECEC432)
- Instructor for Network Programming (ECEC433)
- Instructor and Author for Special Topics in Network Security (ECEC490)
- Instructor for Programming for Engineers (ECE203)
- Independent Study Advisor: Special Topics in Computer Architecture (CS498)
- Independent Study Advisor: Advanced Topics in CPU Design (CS498, ECEC497)

Selected Professional and Research Experience

Research Assistant — Drexel University
Philadelphia, Pennsylvania — 2001-2008

- Researched projects and tools for reverse engineering, software maintenance and program understanding with the Software Engineering Research Group (SERG)
- Developed REportal: A Web Based Reverse Engineering Portal[1] and re-engineered it into a maintainable, distributed Service Oriented Architecture
- Developed Abstract Software Cluster Visualization tool using XML

[1] `http://reportal.cs.drexel.edu`

Technical Writer and Research Assistant — Drexel University ACIN Program
Camden, New Jersey — 2005-2008

- Performed research, development and technical writing related to reverse engineering and software architecture with US Army CERDEC RDECOM C2D under the Applied Communications and Information Networking (ACIN) Project
- Served as a contributing member of the US Army Intelligent Agent (IA) sub-IPT group
- Researched tools and best practices to investigate software engineering and reverse engineering methods for distributed, multi-agent systems
- Co-authored the Agent Systems Reference Model (ASRM) and the Agent Systems Reference Architecture (ASRA)

Application Developer and IT Manager — David M. Banet and Associates
Exton, Pennsylvania — 2001-2005

- Developed customized software solutions to simplify business tasks, including:
  - Payroll integration system to compute payroll benefit deductions and integrate with client proprietary payroll systems
  - XML based electronic audit system
  - Specialized web-based services for clients
- Built and maintained network domain and all associated servers, workstations and integrated business tools

IT Consultant — Consulting Group, Inc.
Malvern, Pennsylvania — 2004

- Served as IT technician for the office during Open Enrollment period
- Assisted in the maintenance of network domain and associated workstations
- Provided technical support in troubleshooting workstations and servers

Volunteer — Chester-Wallingford Red Cross
Chester, Pennsylvania — 2003

- Replaced token-ring computer network with 10/100 ethernet and internet connectivity on a volunteer basis

Instructional Technology Consultant — Upper Darby School District
Upper Darby, Pennsylvania — 1998-2003

- Designed and managed web site for Upper Darby School District
- Created Upper Darby School District Instructional Intranet (RoyalNet) and developed systems to automate content management
- Presented Act 48 training workshops on Filemaker Pro and Dreamweaver for staff, teachers and principals, to maintain and add value to the production of the above district-wide systems
- Interviewed for an Upper Darby School Board seat in November, 2000
- Served on the Upper Darby Technology and Grant Writing Committee under the school board in March, 2001

Application Developer and Consultant — StarComm Development, Inc.
Upper Darby, Pennsylvania — 1999-2002

- Developed and maintained dynamic, database-driven internet applications (ASPs) for clients
- Provided development support on FileMaker programming and educational projects for various companies and educational institutions
- Assisted in the development process and provided technical support for Drexel University's online college application system

Web Application Developer — Boss Entertainment
King of Prussia, Pennsylvania — 2001

- Designed, developed, managed and maintained e-Commerce web site

- Supported development of golf course management system

Script Reviewer — Bergwall Productions
Kennett Square, Pennsylvania — 1998-1999

- Read, reviewed, and edited scripts for Bergwall Productions instructional computer videos

Publications and Grants

Grants

1. NSF Smart and Connected Health (SCH): Smart and Connected Health for Neonatal Transport (co-Principal Investigator with Kapil R. Dandekar, Principal Investigator, and Vineet Bhandari, Anup Das, Genevieve Dion, co-Principal Investigators), 2019-2022, $792,008; Pending.
2. Internet of Things for Future Smart City Philadelphia (Co-Principal Investigator with Kapil R. Dandekar, Principal Investigator, *et al*), 2018; Drexel DARE Proposal, $250,000 (estimated); Pending.
3. CyberCorps Scholarship for Service (SfS) Site (co-Principal Investigator with Xia Lin, Dario Salvucci, Ali Shokoufandeh, Brian Smith, co-Principal Investigators), 2018-2022; Pending.
4. NSF Computer and Network Systems (CNS): NeTS: Small (award number 1816387): Architectures and Testbed for Functional Fabric RFID in the Internet of Things (co-Principal Investigator with Kapil R. Dandekar, Steven Weber, Genevieve Dion, co-Principal Investigators), 2018-2021, $499,647.
5. Arthur Vining Davis Foundations: Research on Experiential STEM Curricula for Authentic Learning Experiences (RESCALE) (co-Principal Investigator with Jennifer Stanford, Jason Silverman, Adam Fontecchio, Eric Brewe, co-Principal Investigators), 2018-2019, $299,957; Pending.
6. Colonial Academic Alliance IN/CO: Tracking Experiential Learning Outcomes Across Three CAA Campuses (co-Principal Investigator with N. John DiNardo, Adam Fontecchio, Jennifer Stanford, *et al*, co-Principal Investigators), 2017-2019, $40,000.
7. NSF Networking Technology and Systems (NeTS), EArly-concept Grants for Exploratory Research (EAGER): SC2: Team Dragon Radio (co-Principal Investigator with Kapil Dandekar, Principal Investigator, Steven Weber, Nagarajan Kandasamy, and Geoffrey Mainland, co-Principal Investigators), 2017, $99,978.
8. Analytics on Real-Time Biometrics from Passive Wearable Smart-Garments (co-Principal Investigator with Adam Fontecchio, Principal Investigator); 2017-2018, Commonwealth Universal Research Enhancement (CURE) Formula Grant (SAP117558-014), $75,000. Drexel University Co-op Funding Award supplement of $7,250 to support undergraduate experiential learning in research.
9. Integrated Design Research and Engineering for Advanced Manufacturing (iDREAM) (Senior Personnel with Genevieve Dion, *et al*), 2016; Drexel DARE Proposal, $250,000.
10. Experiential Learning through the Cooperative Education Lifecycle (ExCEL) (Senior Personnel with Adam Fontecchio, Jason Silverman, Jennifer Stanford, Pramod Abichandani, Kapil Dandekar, David Goldberg, Antonios Kontsos, Suzanne Rocheleau, and Brian Smith), 2016; Drexel DARE Proposal, $250,000.
11. Three Proosals for Student-Centered Spaces for Computing and Informatics. Bruce Char and William Mongan. Drexel University Steelcase Award, 2015.
12. NSF/IEEE-TCPP Curriculum Initiative on Parallel and Distributed Computing – Core Topics for Undergraduates: Using BigData for Learning about a Slice of Parallel Computation in Several Courses (Co-Principal Investigator with with Bruce Char and Jeffrey Popyack), July, 2015-July, 2017. $2,500 faculty development award.

13. NSF CPS: TTP Option: Synergy: Sensing, Processing, and Actuation of Biomedical Smart Textiles for Deep Venous Thrombosis Prevention (award number U01EB023035), March, 2016-March, 2020 (co-Principal Investigator with Kapil Dandekar, *et al*), $1,392,557 (estimated).
14. NSF Partnerships For Innovation: Building Innovation Capacity (PFI:BIC) Grant Number 1430212 on Biomedical Smart Textiles, 2014-2017 (Senior Personnel with Kapil Dandekar, *et al*, co-Principal Investigators), $799,577.
15. IBM Big Data and Analytics Grant, 2013 (Jeffrey L. Popyack and William M. Mongan, co-Principal Investigators), $10,000.
16. NSF RET in Engineering and Computer Science Site for Machine Learning, Big Data and CS Principles, National Science Foundation, DUE-0837665, July 2013-June 2016 (Senior Personnel and co-Director with Jeffrey L. Popyack, Principal Investigator, Mary Jo Grdina, co-Principal Investigator), $499,990.

Book Chapters

1. Duc N. Nguyen, Kyle Usbeck, William M. Mongan, Christopher T. Cannon, Robert N. Lass, Jeff Salvage, William C. Regli, Israel Mayk, Todd Urness. **A Methodology for Developing an Agent Systems Reference Architecture**. *Agent-Oriented Software Engineering XI*, pp. 177-188. Danny Weyns, Marie-Pierre Gleizes, eds, Springer Berlin Heidelberg: 2011.

Archival Journal Publications

1. Sayandeep Acharya, William M. Mongan, Ilhaan Rasheed, Yuqiao Liu, Endla Anday, Genevieve Dion, Adam Fontecchio, Timothy Kurzweg, and Kapil R. Dandekar. **Ensemble Learning Approach via Kalman Filtering for a Passive Wearable Respiratory Monitor**. *IEEE Transactions of Biomedical and Health Informatics, July 2018.*
2. Damiano Patron, William Mongan, Timothy Kurzweg, Adam Fontecchio, Genevieve Dion, Endla Anday, and Kapil R. Dandekar. **On the Use of Knitted Antennas and Inductively Coupled RFID Tags for Wearable Applications**. *IEEE Transactions on Biomedical Circuits and Systems, January 2016.*
3. William Regli, Israel Mayk, Christopher Cannon, Joseph Kopena, Robert Lass and William M. Mongan. **Development and Specification of a Reference Architecture for Agent-Based Systems**. Published in the *IEEE Transactions on Human-Machine Systems, 2013.*
4. William Regli, Israel Mayk, Christopher J. Dugan, Joseph B. Kopena, Robert N. Lass, Pragnesh Jay Modi, William M. Mongan, Jeff K. Salvage and Evan A. Sultanik. **Development and Specification of a Reference Model for Agent-Based Systems**. Published in *IEEE Transactions on Systems, Man, and Cybernetics, September 2009.*

Conference and Workshop Proceedings Publications

1. William M. Mongan, Robert Ross, Ilhaan Rasheed, Yuqiao Liu, Khyati Ved, Endla Anday, Kapil Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam Fontecchio. **Data Fusion of Single-Tag RFID Measurements for Respiratory Rate Monitoring**. *IEEE Signal Processing in Medicine and Biology (SPMB) 2017, December, 2017.*
2. Shrenik A. Vora, William M. Mongan, Endla K. Anday, Kapil R. Dandekar, Genevieve Dion, Adam K. Fontecchio, and Timothy P. Kurzweg. **On Implementing an Unconventional Infant Vital Signs Monitor with Passive RFID Tags**. *IEEE International Conference on RFID, 2017, May, 2017.*
3. William Mongan, Ilhaan Rasheed, Khyati Ved, Shrenik Vora, Kapil Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam Fontecchio. **On the Use of Radio Frequency Identification for Continuous Biomedical Monitoring**. *ACM/IEEE International Conference on Internet-of-Things Design and Implementation (IoTDI) 2017, April, 2017.*

4. William M. Mongan, Ilhaan Rasheed, Khyati Ved, Ariana Levitt, Endla Anday, Kapil Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam Fontecchio. **Real-Time Detection of Apnea via Signal Processing of Time-Series Properties of RFID-Based Smart Garments**. *IEEE Signal Processing in Medicine and Biology (SPMB) 2016, December, 2016.*
5. William Mongan, Endla Anday, Genevieve Dion, Adam Fontecchio, Tim Kurzweg, Yuqiao Liu, Owen Montgomery, Ilhaan Rasheed, Cem Sahin, Shrenik Vora, and Kapil Dandekar. **A Multi-Disciplinary Framework for Continuous Biomedical Monitoring Using Low-Power Passive RFID-based Wireless Wearable Sensors**. Proceedings of the *IEEE Smart Systems Workshop 2016, May, 2016.*
6. Shrenik Vora, William Mongan, Kapil Dandekar, Adam Fontecchio, and Tim Kurzweg. **Wireless Heart and Respiration Monitoring for Infants through Passive RFID Tags**. *International Conference on Biomedical and Health Informatics (BHI), February, 2016.*
7. Duc N. Nguyen, Robert N. Lass, Kyle Usbeck, William M. Mongan, Christopher T. Cannon, William C. Regli, Israel Mayk and Todd Urness. **Developing an Agent Systems Reference Architecture**. Published in the *Proceedings of the* 11^{th} *International Workshop on Agent Oriented Software Engineering, May 2010.*
8. William M. Mongan and William C. Regli. **A Cyber-Infrastructure for Supporting K-12 Engineering Education through Robotics**. Published in *The Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI) Education Track 2008.*
9. William M. Mongan, Maxim Shevertalov, Spiros Mancoridis. **Re-engineering a Reverse Engineering Portal to a Distributed SOA**. Published in *IEEE Proceedings of the* 16^{th} *International Conference on Program Comprehension (ICPC) 2008.*
10. Quincy Brown, William Mongan, Elaine Garbarine, Dara Kusic, Eli Fromm, Adam Fontecchio. **Computer Aided Instruction as a Vehicle for Problem Solving: Scratch Programming Environment in the Middle Years Classroom**. Published in *Proceedings of the American Society for Engineering Education (ASEE) K-12 Track 2008.*
11. William M. Mongan, Christopher J. Dugan, Robert N. Lass, Andrew K. Hight, Jeff Salvage, William C. Regli, Pragnesh J. Modi. **Dynamic Analysis of Agent Frameworks in Support of a Multiagent Systems Reference Model**. Published in *IADIS Proceedings of the International Conference Intelligent Systems and Agents (ISA) 2007.*
12. Christopher J. Dugan, Pragnesh Jay Modi, Joseph Kopena, William M. Mongan, William C. Regli, Israel Mayk. **A Reference Model for Agent-Based Command and Control Systems**. Published in *Proceedings of the* 25^{th} *Army Science Conference 2006.* **Acceptance Rate 14%**
13. Pragnesh Jay Modi, Spiros Mancoridis, William M. Mongan, William Regli, Israel Mayk. **Towards a Reference Model for Intelligent Agent Systems**. Published in *Proceedings of the International Conference of Autonomous Agents and Multiagent Systems (AAMAS) 2006.*

Technical Reports

1. Israel Mayk and William C. Regli, eds; William M. Mongan, *et al.* **Agent Systems Reference Architecture**. Published 2008.
2. Israel Mayk and William C. Regli, eds; William M. Mongan, *et al.* **Agent Systems Reference Model**. Published 2006.
3. C. Liao, S. Mancoridis, W. Mongan. **REportal: A Reverse Engineering Portal**. Drexel University Technical Report. Published 2003.

Invited Talks

1. **Contextualizing Principles of Computer Science**. *CASTLE Pedagogical Happy Hour.* Philadelphia, PA. May, 2018.
2. **CASTLE Summit at Drexel University**. *Faculty Panelist.* Philadelphia, PA. May, 2018.

3. **MapReduce Parallelism across the Curriculum: an Interim Report**. *8th NSF/TCPP Workshop on Parallel and Distributed Computing Education (EduPar-18) with Bruce Char and Jeffrey Popyack.* Vancouver, BC, Canada. May, 2018.
4. **National Council of Women in Information Technology (NCWIT)**, *Invited Faculty.* Tucscon, AZ. May, 2017.
5. **Integrating Active Learning in the STEM Classroom (Inaugural Event)**. *Pedagogical Readiness Oversight for Future Educators in STEM Subjects (PROFESS) at Drexel University.* Philadelphia, PA. April, 2017.
6. **A Panel Discussion on Computer Science Education**. *Computer Science Teachers Association Fall Symposium.* Villanova, PA. November, 2016.
7. **Wearable Technology Advances Health for Mothers and Babies**. *Yale Tech Summit presentation with Owen Montgomery.* New Haven, CT. October, 2016. Also presented to Teva in Frazer, PA, November, 2016.
8. **Using BigData for Learning about a Slice of Parallel Computation in Several Courses**. *NSF/TCPP Workshop on Parallel and Distributed Computing Education (EduPar-16) Poster with Bruce Char and Jeffrey Popyack.* Chicago, IL. May, 2016.
9. **Pixels, Post-It's and CS Principles**. *SIGCSE 2016 Poster with Jeffrey L. Popyack.* Memphis, TN. March, 2016.
10. **Statistical Analytics of Wearable Passive RFID-based Biomedical Textile Monitors for Real-Time State Classification**. *IEEE Signal Processing in Medicine and Biology (SPMB) Symposium Poster with Kapil Dandekar, Genevieve Dion, Tim Kurzweg, and Adam Fontecchio.* Philadelphia, PA. December, 2015.
11. **Wearable Smart Textiles Based on Programmable and Automated Knitting Technology for Biomedical and Sensor Actuation Applications**. *BIO International Convention with Kapil R. Dandekar, Genevieve Dion, Adam Fontecchio, Tim Kurzweg, Owen Montgomery, V.K. Narayan.* June, 2015.
12. **Feedback at Scale - Automatically Generated Feedback for CS Student Work: Best Practices**. *SIGCSE 2015 Birds-of-a-Feather Session with Bruce W. Char, Jeffrey L. Popyack, and Jeremy Johnson.* Kansas City, MO. March, 2015.
13. **Raspberry HadooPI: A Low-Cost, Hands-On Laboratory in Big Data and Analytics**. *SIGCSE 2015 Poster with Ken Fox and Jeffrey L. Popyack.* Kansas City, MO, March, 2015 and *Drexel University Research Day 2015.*
14. **A Software Framework for Monitoring and Performing Analytics on Real Time Medical Device Data**. *American Society for Engineering Education Poster with Rachel M. Goeken, Kapil Dandekar, Timothy Kurzweg, Gionvieve Dion, and Adam K. Fontecchio.* Swarthmore, PA. November, 2014. **Runner-Up for Best Poster**
15. **Marconi: The Master of Wireless**. *Bellyband Demonstration at Drexel University.* Philadelphia, PA. October, 2014.
16. **Big Data, Big Deal; Welcome to the Twitterverse**. *Reboot, REThink, Refresh with Jeff Popyack.* July, 2014.
17. **Maternity Smart Fabric Bellyband to Monitor Uterine Activity and Assess Fetal Well-Being**. *Wearable Technology in Healthcare Society (WATCH) Conference with Kapil Dandekar, Genevieve Dion, Adam Fontecchio, Timothy Kurzweg, and Owen Montgomery, MD.* Indianapolis, IN. July, 2014.
18. **Big Data is Everywhere: Bridging Computing Disciplines and Society**. *Webinar for Drexel University with Jeff Popyack.* June, 2014 and January, 2014.
19. **What Is/Are CS Principles?** *Philadelphia Science Festival with Jeff Popyack and Omar Ali, School District of Philadelphia.* April, 2014.
20. **Learning Real-World Skills while Majoring in Computer Science**. *GETT: Girls Exploring Tomorrow's Technology with Jeff Popyack, Suzanne Hanbicki, and Hannah Pinkos.* April, 2014.

21. **Internet Protocols and Cryptography**. *Reboot Renew REThink Workshop at Drexel University.* July, 2013.
22. **Mobile Application Development with Web Services**. *Google CS4HS Workshop at the University of Pennsylvania.* August, 2012.
23. **Networking Applications, Protocols, and Cryptography**. *Computing Tapestry Workshop at the University of Pennsylvania.* July, 2012.
24. **Linux Kernel Vulnerabilities**. *Drexel University Math and Computer Science (MCS) Society Talk.* March, 2012.
25. **An Integrated Introduction to Network Protocols and Cryptography to High School Students**. *Poster Presentation at ACM SIGCSE 2012, Raleigh, NC.* March, 2012.
26. **Networking Applications, Protocols, and Cryptography with Java**. *Google CS4HS Workshop at the University of Pennsylvania.* August, 2011.
27. **Version Control Systems**. *Drexel University Math and Computer Science (MCS) Society Talk.* July, 2010.
28. **Certification and standards for computing education in Pennsylvania**. Invited Panelist with Dougherty, J.P., Griffin, J., Pirmann, T., and Powell, R. Panel presentation submitted to *The Twenty-fifth Annual Consortium for Computing Science in Colleges (CCSC) Eastern Conference*, Villanova University, Villanova, PA. October 30 - 31, 2009.
29. **How the PC Starts its Day: From Boot Code to Boot Viruses**. *Drexel University Math and Computer Science (MCS) Society Talk.* February, 2009.
30. **Circuit Design with Breadboards, Microcontrollers, and FPGAs**. *Drexel University Math and Computer Science (MCS) Society Talk.* November, 2008.
31. **Computer Organization Workshop**. *The Pennsylvania Governor's Schools of Excellence: Information, Society & Technology.* Summer, 2008.
32. **GK-12: Engineering as a Contextual Vehicle for Science and Mathematics Education Poster**. *2007 NSF GK-12 Annual Meeting, Washington DC* and *2007 Drexel Research (RISC) Day.*
33. **REportal: A Web-Based Reverse Engineering Portal**. *2003 Drexel University Research Day.*

Media Coverage

1. Bellyband and Exoskin Glove featured at Chemical Heritage Foundation, October, 2016.
2. Science Nation Coverage of Biomedical Smart Textiles: "These Smart Threads Could Save Lives"[2], September, 2016.

Honors, Certifications, Professional Membership, and Mentorship

Honors and Certifications

- Drexel Center for the Advancement of STEM Teaching and Learning Excellence (CASTLE) Instructor of the Week, February, 2018
- FCC Amateur Radio License with callsign W1CLK (Technician, General), 2017
- Pennsylvania Boater Safety Certification, 2017
- FCC Restricted Radiotelephone Operator (RR) License, 2016
- College of Computing and Informatics (CCI) Teaching Excellence Award, 2014
- Pennsylvania Instructional I Teaching Certificate earned September, 2008
- Dean's Fellow September, 2005 through September, 2008
- Member of the Drexel University Upsilon Pi Epsilon International Honor Society for the Computing Sciences since June, 2004; elected Treasurer August, 2006
- Security clearance at the level of SECRET granted April, 2006
- Complex-rated Private Pilot (Airplane Single Engine Land) May, 2005; Instrument-rated Private Pilot (Instrument Airplane) March, 2006
- Member of the Golden Key International Honor Society since January, 2001

[2] https://www.nsf.gov/news/special_reports/science_nation/biomedtextiles.jsp?WT.mc_id=USNSF_51

- Member of the National Society of Collegiate Scholars since January, 2001

Professional Membership and Service

- Reviewer for IEEE Sensors Journal, 2018
- Reviewer for PLOS One Journal, 2018
- Member of the American Radio Relay League (ARRL), 2017-Present
- Technical Editor for Ethical Case Studies for NSF EESE: The Ethics of Algorithms, 2016-2017
- Reviewer for Elsevier Pervasive and Mobile Computing, 2015
- Reviewer for SIGCSE 2012-2013; Session Chair in 2012
- Member of and Reviewer for the International Advisory Committee for the International Joint Journal Conference in Engineering (IJJCE) 2009
- Fellow of the Association of Computer Electronics and Electrical Engineers (ACEEE) since January, 2009
- Member of the Experimental Aircraft Association (EAA240) from May, 2008 through May, 2009
- Member of the National Council of Teachers of Mathematics (NCTM) from March, 2008 through March, 2009
- Student Member of the American Institute of Aeronautics and Astronautics (AIAA) from October, 2006 through September, 2008
- Member of the Institute of Electrical and Electronics Engineers (IEEE), IEEE Computer Society, and IEEE Signal Processing Society from September, 2005 through January, 2011, and since August, 2014, and IEEE Circuits and Systems Society since November, 2017
- Senior Member of the Association for Computing Machinery (ACM), Special Interest Group in Computer Science Education (SIGCSE), and Computer Science Teachers Association (CSTA) since May, 2004; Senior Member since March, 2012
- Member of the Aircraft Owners and Pilots Association (AOPA) since May, 2004

Mentorship

- Faculty Mentor to 6 Vertically Integrated Projects (VIP) students, including STAR Scholars, for Biomedical Textile Analytics projects, 2018
- Faculty supervisor of two undergraduate co-op students in Biomedical Analytics under CURE grant, 2017-2018; one of whom was named a 2018 Goldwater Scholar
- Co-Faculty Mentor of Research Experiences for Teachers (RET) project in Biomedical Multisensor Data Fusion, 2016-2017 and 2017-2018
- Co-Faculty Mentor of Students Tackling Advanced Research (STAR) freshman research student in Biomedical Multisensor Data Fusion
- Co-Advisor for CS and ECE senior design group project: Biomedical smart actuation
- Co-Advisor for CS, Mechanical Engineering, and ECE senior design group project: Actuation for Deep-Veneous Thrombosis
- Faculty Advisor for DUCSTeach education outreach student organization, 2016-2017
- Faculty supervisor for two students under an NSF-Sponsored Research Experiences for Undergraduates (REU) Supplement, 2016
- External Stakeholder for Smart Home Senior Design Project
- Technical co-Advisor for Which MBA 2016, sponsored by the Economist: "Kaspersky Lab Cyber Security Case Study"
- Co-Advisor for ECE senior design group project: Software Infrastructure for Secure and Scalable Medical Sensor Networks
- Co-Faculty Mentor of Students Tackling Advanced Research (STAR) freshman research student in Biomedical Smart Textile Analytics
- Co-Faculty Mentor of Students Tackling Advanced Research (STAR) freshman research students (2) in Distributed MapReduce with Hadoop
- Mentor and Judge for Philly CodeFest Hackathon 2015
- Judge for Students Tackling Advanced Research (STAR) Undergraduate Research Program
- Advisor for CS senior design group project: Scan Technologies
- Advisor for ECE senior design group project: Smart Fabric Monitoring Devices
- Faculty co-Sponsor of Masters in Medical Science (MMS) Student Projects: *Maternity "SMART Fabric Bellyband" to Monitor Uterine Activity and Assess Fetal Well-Being* and *Smart Onesie:*

Early Warning System for the Prevention of Sudden Infant Death Syndrome (SIDS)

- Faculty Advisor to the Arduino Developers at Drexel University
- Founder of the Web Services User Group at Drexel University
- External Stakeholder for Flight Control and Failure Detection Senior Design Project
- Advisor for Life Insurance Big Data Senior Design Project
- Co-Founder of the Project Nebula Big Data Research Group
- Co-Faculty Mentor of Students Tackling Advanced Research (STAR) freshman research students (2) in Big Data and Analytics
- Faculty Mentor of Students Tackling Advanced Research (STAR) freshman research student group in Biomedical Systems
- Co-Advisor of Freshman Design Project: Big Data and Analytics using the Mathforum at Drexel University
- Judge for Drexel University Research Day Competition
- Faculty Advisor to the Math and Computer Science (MCS) Society, ACM undergraduate student chapter, 2012-2015
- Advisor for Senior Design Group: Dynamic Traffic Control System
- Advisor for Freshman Design Group: Drexel Map Android Application
- Co-Advisor for CS senior design group project: Classroom Application Media Center (CRAM)
- Advisor for ECE senior design group project: Alzheimer's In-Home System
- Advisor for CS senior design group project: Project Sekai - Experiencing the World
- Faculty Advisor to an IEEE Systems Student Competition team
- Advisor for ECE senior design group project: AutoCaddy (Autonomous Golf Cart)
- Technical Advisor for CS senior design group project: ANFS (Advanced NFS)
- Advisor for engineering freshman design group project: Interactive Android Battlefield Map
- Co-Advisor for CS senior design group project: Visual VHDL (V2HDL)
- Co-Advisor for CS senior design group project: REportal service integration