Measuring Wireless Fingerprints Inside a Wireless Sensor Network

by

David A. Knox

A thesis submitted to the Faculty of Graduate and Postdoctoral Affairs in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Systems and Computer Engineering

Carleton University
Ottawa, Ontario

Abstract

Wireless fingerprints authenticate transmitters in wireless networks, using attributes of the physical wireless signal rather than information being conveyed with that signal. They can be used effectively in an intrusion detection system for networks, where nodes are physically vulnerable. While previous research shows that wireless fingerprints work reliably in a laboratory setting, little work has been published showing the feasibility of their implementation within a real network. We present methods for wireless fingerprints using data sampled at the demodulation rate. This eliminates the need for high bandwidth data processing, making them feasible from inside a wireless sensor network. We analyze their classification performance empirically for different network conditions and theoretically examine aspects of their secure usage in a network.

To the best of our knowledge, our research is the only published work that uses representative wireless networking hardware for wireless fingerprints. We discriminate between different IEEE 802.15.4 2.4 GHz Radio Frequency (RF) sources, using the SiLabs IEEE 802.15.4 WSN node development platform and the Ettus Labs USRP1 Software-Defined Radio. The wireless fingerprinting method implemented on the SiLabs WSN node discriminates between RF sources using differences in the Automatic Gain Control circuitry time response during the initial appearance of RF signals. The results show that different RF sources can be distinguished over short transmission distances. The more sophisticated USRP1 device exploits differences in the phase attributes of RF signals using a larger set of demodulated data samples than was available with the WSN node. The USRP1 classifies more accurately over a wider range of network conditions: time, transmission distance and also different receiving devices. Our average

classification accuracies are: 99.6% at short range, 95.3% at medium range and 81.9% at long range using five SiLabs devices.

We demonstrate the independence of classification errors made over different RF channels and the benefit of using multiple nodes for classification. This suggests that nodes can collaborate to increase the reliability of wireless fingerprints, either by comparing their classification decisions or by aggregating their fingerprints. We present a secure group key establishment protocol, using wireless fingerprints for authentication, for use in a network containing malicious nodes.

Acknowledgements

I would like to thank my family for their patience during this work. I would also like to thank my supervisor, Thomas; his excellent verbal and written feedback for my work over this time has been much appreciated. We have worked together remotely, via weekly PC-based video conference, for five years, which has allowed me to live outside of Canada with my family, during my research.

Table of Contents

List of Tables

List of Illustrations

List of Appendices

List of Acronyms and Terms

8051F121:	Microcontroller made by SiLabs
A/D:	Analog to Digital
ADC:	Analog/Digital Converter
AGC:	Automatic Gain Control
ASCII:	American Standard Code for Information Interchange
B0:	Bit 0 (least significant bit for LNA control input)
B1:	Bit 1 (most significant bit for LNA control input)
CC2420:	Texas Instruments RF Integrated Circuit
CCA:	Clear Channel Assessment (signal pin on the Texas Instruments RF IC)
Chip:	A single pulse of Direct Sequence Spread Spectrum code
CI:	Confidence Interval
CMOS:	Complementary Metal Oxide Substrate
CPU:	Central Processing Unit
DSP:	Digital Signal Processing
DSSS:	Direct Sequence Spread Spectrum
EEROM:	Electrically-Eraseable Read Only Memory
FFT:	Fast Fourier Transform
FIFO buffer:	First In First Out buffer (data buffer on Texas Instruments RF IC)
FIFO:	First In First Out (signal pin on the Texas Instruments RF IC)
FIFOP:	First In First Out Positive (signal pin on the Texas Instruments RF IC)
FLASH:	Type of non-volatile electrically-erasable programmable memory
FPGA:	Field-Programmable Gate Array
I/Q:	In-phase/ Quadrature-phase
I:	In-phase
IC:	Integrated Circuit
ID:	IDentifier (code which is unique to a specific Radio Transmitter)
IEEE:	Institute for Electrical and Electronic Engineers
IF:	Intermediate Frequency
ISR:	Interrupt Service Routine
JTAG:	Joint Test Access Group
LED:	Light Emitting Diode
LPF:	Low-Pass Filter
LNA:	Low-Noise Amplifier
LoS:	Line of Sight path (a direct and unobstructed path)
M/M:	Mueller and Müller (synchronization method)
MAC:	Medium Access Control
OFDM:	Orthogonal Frequency Division Multiplexing
OQPSK:	Offset Quadrature Phase Shift Keying

PC:	Personal Computer
PCA:	Principal Component Analysis
PCB:	Printed Circuit Board
PHR:	PHysical layer headeR
PHY:	PHYsical layer
PN:	Positive/Negative
PPDU:	PHY Physical Data Unit
PPM:	Parts Per Million
PR:	Phase Residual (or residual phase)
PrC	Principal Component
PrCA:	Programmable Counter Array
PSD:	Power Spectral Density
PSDU:	Physical Service Data Unit (IEEE 802.15.4 packet data field)
Q:	Quadrature-phase
RAM:	Random Access Memory
RF:	Radio Frequency
RFX2400:	RF Ettus Research 2.4GHz product
RPV:	Residual Phase Vector (or 'error vector')
RS-232:	Recommended Standard 232 serial communications standard
Rx:	Receiver
SDR:	Software-Defined Radio
SFD:	Start of Frame Delimiter (specific pin on the Texas Instruments RF IC)
SHR:	Synchronization HeadeR
SiLabs:	Silicon Labs (Manufacturing company name)
SPI:	Serial Programming Interface
SYNC:	SYNChronization signal in IEEE 802.11 wireless standard
Timer4:	Timing resource on the SiLabs 8051F121 microcontroller
TRPV:	Template for Residual Phase Vector (or residual phase vector template)
Tx:	Transmitter
UART:	Universal Asynchronous Receiver/Transmitter
USB:	Universal Serial Bus (common serial communications standard)
USRP1:	Universal Software Radio Peripheral (version 1)
VGA:	Variable Gain Amplifier
WFP:	Wireless FingerPrint
WLAN:	Wireless Local Area Network

1 Chapter: Introduction

In an 'Aging in Place' [**1**] application, a distributed monitoring system can be used that allows family members or paid care-givers to check the well-being of older adults in their own homes. As the average age of the world's population increases, caring for older people in their own homes becomes a necessity. This application is an example of a modern network, where the security and privacy needs of the users is important and must be protected because of the personal nature of the data being monitored [**2**]. In such networks, data confidentiality and authenticity are important and details of owner occupancy information must be kept secure to avoid their use by malicious third parties or even for financial purposes by non-malicious third parties. Wireless Sensor Networks (WSNs) are attractive for such applications because of low hardware cost, small node size and limited node power consumption.

1.1 Wireless Fingerprints

Wireless fingerprints (WFPs) are a biometric-style authentication mechanism that can be used to distinguish legitimate nodes from intruder nodes in WSNs. Using attributes of the RF signal for authentication, previous researchers have adopted the term 'RF Fingerprints' to describe such authentication mechanisms. We prefer the more general term of Wireless Fingerprints and we use it to refer to schemes that use *any* characteristics of the wireless signal (e.g. power, timing, or amplitude, phase or frequency). We define Wireless Fingerprints (WFPs) as *any digital representation of the physical layer attributes of a wireless signal that vary in a characteristic fashion with the particular wireless transmitter*.

Since WFPs are derived from physical-layer attributes of a wireless signal, they are more robust than other security methods against the lack of physical security that occurs when WSN nodes are placed in a residential setting. Traditional key-based security mechanisms are vulnerable to the recovery of key information by an attacker. Since WSN nodes have limited hardware and software memory protection mechanisms, less time is required for brute force key recovery attacks.

WFPs can be used effectively as part of an intrusion detection system, allowing unauthorized nodes to be detected and also for re-establishing trust in compromised networks. The RF interface on the WSN node is always present, so no additional radio or electrical interfaces are needed to implement WFPs.

1.2 Requirements for Wireless Fingerprints in Distributed Networks

Wireless Local Area Network (WLAN) nodes have large batteries, making centralized network designs feasible. In WSNs, however, nodes have limited battery charge and much less power is available for wireless reception and transmission than in infrastructure types of networks. A WFP implementation that does not require a transmission of signals to and from a central base station is preferable in WSN networks. Our algorithms are fully distributed, requiring no centralized processing or communications.

Previous wireless fingerprint research uses specialized auxiliary test equipment to make RF measurements, analyze the RF signals and perform the required discrimination tasks. These methods are not feasible for a fully distributed implementation. Our work represents the first published effort to implement WFPs on the WSN nodes themselves. We prove feasibility for WFPs on existing WSN node hardware and also demonstrate

better classification performance on a software-based platform that is similar to emerging current commercial consumer wireless products.

We implement different WFP algorithms on a Silicon Labs (SiLabs) WSN hardware platform and on an Ettus Research Universal Software Radio Peripheral Version 1 (USRP1) Software-Defined Radio (SDR) hardware platform. On the WSN node, bit-level access to Automatic Gain Control (AGC) control logic outputs is used as the input for the WFP algorithm. The AGC input is a signal derived from the amplitude attributes of the input signal.

On the SDR, raw symbol-rate data is used for the WFP algorithm input and our algorithm uses phase attributes of this input signal. Both platforms use a zero-Intermediate Frequency (zero-IF) receiver architecture, but the SDR provides access to an uninterrupted stream of demodulated data samples. This makes the SDR ideal for WFP algorithm experimentation. We show that WFP algorithms can be implemented and used for receiver discrimination within a WSN network using either of our two reference platforms, although we achieve better classification performance using the USRP1 platform.

We study whether WFPs remain consistent for different receivers over different channel conditions and over time. If WFP classification errors at different receivers are independent under such conditions, multiple nodes can collaborate to improve overall WFP classification performance in a distributed network and we verify this experimentally. We modify an existing protocol for deriving a shared group key in a distributed network that can contain a minority of malicious nodes, augmenting the protocol to use WFPs for authentication.

1.3 Thesis Contributions and Organization

The basic contributions of this thesis are as follows:

- Implementation of basic Wireless Fingerprints on a real WSN node, showing feasibility for discriminating between different RF sources. The following papers document our results in this area:
    - D. A. Knox and T. Kunz, "AGC-based RF Fingerprints in Wireless Sensor Networks for Authentication," in *IEEE World of Wireless Mobile and Multimedia Networks (WoWMoM)*, Montréal, Canada, 14-17 June 2010 , pp. 1-6. [**3**]
    - D. A. Knox and T. Kunz, "RF Fingerprints for Secure Authentication in Single-Hop WSN," in *IEEE International Workshop on Security and Privacy in Wireless and Mobile Computing Networking and Communications*, Avignon, France, 2008, pp. 567-573. [**4**]
- Analysis and performance characterization of wireless fingerprint algorithms for a WSN platform and identification of significant design criteria for the USRP1 SDR wireless fingerprint algorithm
- Implementation of a new Wireless Fingerprint algorithm using the USRP1 SDR. The following paper documents our results in this area:
    - D. A. Knox and T. Kunz, "Practical RF Fingerprints for Wireless Sensor Network Authentication," in *8th International Wireless Communications and Mobile Computing Conference*, Limassol, CYPRUS, August 2012, pp. 531-536. [**5**]

- Presentation of a modified protocol to establish a shared group key in a wireless setting, augmenting it to use wireless fingerprints for message authentication. The following paper documents our results in this area:
    - D. A. Knox and T. Kunz, "Secure Authentication in Wireless Sensor Networks Using RF Fingerprints," in *IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC '08)*, Shanghai, China, 17-20 December, 2008, pp. 230-237. [**6**]

In Chapter 2, we review related work from the literature. In Chapter 3, we review the stages of the WFP authentication process and present our approach. In Chapter 4, we present the WFP algorithm which we implemented on the WSN hardware platform. In Chapter 5, we analyze the results obtained with the WSN algorithm and identify the areas for further experimental study using the USRP1 WFP algorithm.

In Chapter 6, we describe the USRP1 SDR platform and present several USRP1 WFP algorithms. In Chapter 7, we analyze the classification performance results that we obtained using the USRP1 algorithm variants, selecting one for further analysis over a wider range of experimental conditions. In Chapter 8, using this selected variant of the algorithm, we show that WFP templates produced by other nodes can be used to achieve similar classification accuracy as when using templates that are generated locally. We also show that, when those other nodes are nearer to the RF source for which that WFP template is being created, classification performance can be superior to when using templates generated locally from farther away.

Both of these results have implications for WFPs at a network level. We present the results of a statistical analysis that demonstrates that classification using multiple nodes is

worthwhile and that classification errors are independent on two USRP1 platforms. To this end, we also summarize modifications to an existing group key establishment protocol that uses WFPs from multiple nodes for the authentication of message content. The resulting protocol allows secure establishment or re-establishment of key information in a network that can contain malicious nodes. In the final chapter, we draw conclusions and make recommendations for future work.

2 Chapter: Related Work

As wireless networks (WiFi and cellular) and wireless devices, such as personal organizers, smartphones, media players, become more ubiquitous and contain more personal data, the need for authentication of wireless information sources is growing. The Aging in Place [**7**] application provides an example of this growing need to provide wireless devices that can authenticate each other to meet the strong security and privacy requirements of its users [**2**].

Key distribution methods are traditionally used to establish security and authenticate nodes in WSN networks. WFPs could allow the authentication of nodes without relying on key distribution, but this places severe requirements on their classification accuracy performance. Even with poorer classification performance, WFPs are well-suited for use as part of the key establishment process or perhaps even more applicable as part of an intrusion detection system, once security has been established. In this chapter, we justify the use of WFPs as a method for establishing and maintaining security in a WSN and review the previous WFP research.

2.1 Establishing and Maintaining Security in Wireless Sensor Networks

Modern cryptographic security relies only on the security of key information. Attackers are assumed to know the details of all of the other cryptographic processes that use those keys [**8**]. Wireless system security depends on the secure deployment and use of such key information. Authentication of senders of key information is an important part of a secure key distribution system [**9**] or an intrusion detection system [**10**].

Secure methods to provision or re-establish key information through physical access to nodes have been proposed [**11**][**12**][**13**]. However, direct physical access to WSN nodes

after deployment may not be practical [**14**]. Even if physical access is possible, additional circuitry may be required on the WSN node to allow key information to be provisioned. A WSN node that has been made waterproof by encapsulation in plastic requires mechanisms to allow physical access to take place through the encapsulation. Wireless interfaces are an attractive method for access in such situations. Methods to populate key information over wireless interfaces have been proposed, but making them secure against authentication attacks is difficult [**9**][**15**]. Key deployment over the RF interface is vulnerable to attacks from malicious nodes that can falsify their identities. Researchers have derived innovative key provisioning methods that use the RF wireless interface, augmented with a physical mechanism. One such method requires physical contact in parallel with wireless signaling [**16**] at provisioning time and another uses sequenced patterns from Light Emitting Diodes (LEDs) [**12**]. Creating an isolated RF environment, like the one required in [**17**] is less practical for recovering compromised nodes deployed in home applications. A more recent contribution that does not rely on additional physical methods uses the randomness and symmetry of an RF channel to establish a shared secret key for private and confidential communications between pairs of nodes [**18**].

All of these methods allow security to be boot-strapped without key deployment, prior to provisioning or could be used in an intrusion detection system. However, they still use the data being carried by the wireless signal for authentication purposes, making the system vulnerable to attacks if this information can be compromised. As a minimum, they are vulnerable to the 'man-in-the-middle' type of attack, where a malicious attacker acts as an active wireless bridge between two honest parties [**19**].

2.2 Identification Methods for Wireless Nodes

Location information has also been proposed to 'identify' RF sources [**20**]. The disadvantages of such schemes in a WSN "Aging in Place" application are the ease of physical access to devices and the need to update 'identities' as devices are moved or RF interference conditions change. With these methods, the requirement to establish and maintain the integrity of key information has been substituted for a requirement to establish and maintain the integrity of location information. Other researchers [**21**] have timed node responses to determine their identity. These authors use passive methods to avoid alerting a node that it is being profiled. Alternatively, they also show how to use adaptive or interactive methods, where more accurate and timely measurements can be made at the expense of alerting the node that is being measured.

The authors in [**22**] use a 'passive fingerprinting' technique, that authenticates a node on the basis of the characteristic statistics of sensed variables.. Perhaps the easiest user requirement when authenticating wireless nodes is given in the work of [**23**] where the user just needs to shake nodes together to allow them to authenticate each other. The WSN nodes need to be equipped with accelerometers for this process to work, which is a minor drawback. They establish a common source of randomness based on the accelerometer readings in the two devices being paired. Unfortunately, the authors discovered that the security of their protocol is also very sensitive to the particular way that devices are shaken, leading them to require user training for proper shaking.

Signal strength information (assessed using a Receive Signal Strength Indicator/RSSI) has been proposed as an indicator of 'nearness' for trust purposes and refined to use Signal Strength Difference in [**24**]. Signal strength is used as a biometric attribute of the

wireless signal and the dependence on the integrity of the transmitted data is removed. Unfortunately, signal power can still be adjusted on a per-message basis, making the attribute easier to forge.

However, this research direction is a more promising one for the WSN environment. Other information can be derived from the RF signal to provide a reliable biometric authentication mechanisms and these are surveyed in [**25**]. We now review the research dedicated to deriving reliable wireless biometric authentication mechanisms.

2.3 Wireless Fingerprints

Biometric-based authentication using human fingerprints and other biological processes has been an active area for the last twenty years [**26**]. Here, address uniqueness depends on natural variation of complex life-forms and configuration is not an issue but is automatic and implicit, relying on the fact that every subject has unique fingerprints. The focus of much of the research work has been on: the classification of 'noisy' data, pattern matching algorithms and on the secure storage of the biometric data itself. Typically, the process [**27**] for biometric identification is biometric data capture, feature/classifier separation and extraction, followed by matching with previously gathered templates from a database of biometric features captured earlier of different subjects.

For human fingerprints and other biometric classifiers like human iris scans, there are natural changes over time (e.g. from tissue damage and aging effects). Samples to be used for identification may not be perfect themselves (e.g. they could be smudged). For human biometric classifiers [**28**], the classic approach is to establish a large database of samples in an 'enrolment' stage. During enrolment, the highest quality sample features

are extracted and stored in the database. This database is then used later in a verification stage for actual matching and identification of subjects.

In the verification stage, a measured sample is taken and the appropriate features are extracted from it. These features are then compared with all templates that are currently enrolled in the database and a potential match found. There may be no matches with any stored templates in the database. Therefore, confidence thresholds need to be determined to ensure that there are no false positives as well as to ensure that there are no false negatives when a template is actually in the database but noise effects cause differences.

Rather than using a database and a verification process for matching, Clancy et. al [**29**] extract biometric features and use them directly to obscure a password for authentication purposes. For example, fingerprint mapping can be used on smart cards. Besides not requiring a database, the main advantage of this technique is that no biometric data is stored on the smart card. This is useful if the smart card is assumed to be vulnerable and subject to tampering (as with our WSN nodes in an 'Aging in Place' application).

Key information is encrypted using extracted fingerprint features to create a 'fingerprint vault'. This is derived from the 'fuzzy vaults' proposed earlier by Juels [**30**]. Problems with Juels' implementation were later identified and fixed by the authors in [**31**]. Note that standard cryptographic methods are not applicable since slight variations in key values will map to very different encryptions. As we have stated, practical biometric identifiers have noise associated with them for various reasons and 'fuzzy' encryption attempts to address this.

Tekbas et. al [**32**] gives methods to extract features with orthogonal dimensions and a method to reduce the size of the data and feature set used for classification. They show

that a matched filter to the channel provides the best basis for orthogonalization. For smaller networks with 'tens of neighbour nodes', these researchers state that amplitude characteristics suffice for classification and that phase ones need not be considered. The authors show that both voltage and temperature variations can be calibrated out, using a neural classification approach with extensive training sequences. However, they were unable to determine deterministic effects of temperature and voltage on the classification features. Different nodes will be more likely to have unique RF Fingerprints, given that their voltages or temperatures are different.

This may also imply a need to re-measure or even adaptively adjust RF Fingerprints in a dynamic fashion (as temperature changes and battery voltages fluctuate) and that a certain amount of channel noise variation cannot be tuned out, but simply degrades classification performance. However, they claim that the classification algorithm can be improved by training under noisier conditions and then using those 'noisy' parameters when the Signal-to-Noise ratio degrades down to similar levels.

The basic task of the designer and manufacturer of electronic hardware is to make devices within tight performance tolerances (e.g. to meet a specification like RF spurious emissions). Despite this, variations in performance still occur. Researchers have proposed different methods to exploit these differences for authentication and identification purposes. Suh et. al [**33**] use 'Physical Unclonable Functions' based on variations in hardware circuits like path delays as a basis for non-volatile key generation. The earliest published research that studied biometric authentication mechanisms for wireless networking devices was done by [**32**][**34**] although similar work was started during the Second World War to identify radar installations. The early WFP researchers

and most researchers since used the initial transient portion of the RF waveform to provide the identifying information for the wireless signal. Capturing such transients requires sensitive test equipment with a sampling rate of GHz and specialized low-noise receiver circuitry [**35**]. This high-speed sampling and processing logic is not yet present on WSN nodes. To derive consistent WFPs based on such transient data also requires an accurate estimation of the starting time for a signal transmission, which is more complicated during noisy RF channel conditions or over larger transmission distances. WFPs have been proposed for use in networks with fixed architectures like WiFi WLANs [**10**]. The requirement that the central router node can be reached by all nodes requires higher transmitter power levels than practical for WSN nodes. More recently, researchers showed the feasibility of distinguishing between different WSN RF sources and also proposed the use of WFPs for WSNs [**36**]. While reasonable classification performance is demonstrated, identification is based on measurements made on a high-speed (1GHz) sampling oscilloscope, again making such WFPs feasible only from 'outside' the network. Brik et al. present a method called PARADIS [**37**], which derives fingerprints from the physical attributes of demodulated baseband data like we do. They obtained excellent classification results with 97-99% classification accuracy for a sample of 138 IEEE (Institute for Electrical and Electronic Engineers) 802.11 Network Interface cards in a controlled indoor WLAN environment with line-of-sight (LoS) transmission conditions to a single router location. Their data was measured using a Vector Signal Analyzer test device co-located with this router. This device has a sophisticated low-noise receiver combined with specialized analysis software and is capable of extremely accurate measurements of phase and amplitude.

In spite of the differences in measurement equipment from ours, we believe that their work is still comparable, based on the demodulated nature of their data input. However, their WFP is significantly more complicated than ours, defined as a 5-tuple quantity consisting of: frequency errors, phase errors, magnitude errors, Synchronization (SYNC) errors and In-Phase/Quadrature-Phase (I/Q) origin offset errors. Additionally, the resolution bandwidth of their input data is higher than can be expected on less specialized hardware.

Very recently, Rehman et al. [**38**] have attempted to implement WFPs for resource-constrained mobile devices. They use the time-varying properties of amplitude envelopes of Bluetooth wireless signals to create their WFPs. They obtained excellent classification performance with their method in a noise-less anechoic chamber. While they do make their measurements with low-noise, high-frequency test equipment, they show that they can subsample their data down to lower rates, without degrading performance of their WFP significantly.

Suski et al. [**39**] examined classification accuracy performance for 802.11a Orthogonal Frequency Division Multiplexed (OFDM) RF signals with different Signal-to-Noise ratios. Their objective was to identify the specific manufacturer of a device (e.g. classes of device) rather than individual RF sources. By subtracting ideal waveforms, based on simulated results, from measured ones, they create a WFP. Using this method, they were also able to add calibrated noise levels for different SNRs, allowing the generation of different fingerprints for different transmission noise conditions. Using sophisticated test equipment, they extract a 36 MHz segment of RF spectrum, sampled at 95MHz as input to their WFP algorithm, which is a rate between the 1GHz transient bandwidth used by

most of the previous researchers and the 2MHz baseband signal bandwidth used in our work. Other researchers [**40**] have also attempted RF Fingerprint classification results using methods operating near baseband data rates. However, they use subsequent high-speed digitization of that data, resulting in WFP measurement bandwidths that are still too large to be practical for our purposes.

3 Chapter: New Wireless Fingerprint Approach

To determine WFP information without assistance from other parties, WFP algorithms must be implemented within the network nodes. We restate our general research problem as: "Measuring classification accuracy of wireless fingerprints (WFPs) implemented inside a wireless network". We are not aware of any work published in the area of WFP measurement inside WSNs. No experimental implementations of WFPs that use representative hardware platforms have been published so far, besides our own work [**3**][**5**].

Our WFP algorithm for the SiLabs WSN hardware platform is based on the properties of the Automatic Gain Control circuitry [**41**][**42**]. As a representative future WSN node platform, we also use the Universal Software Radio Peripheral (USRP1) software-defined radio (SDR) [**43**]. SDRs are used extensively for cognitive radio and for recognition of RF modulation schemes [**44**]. Their emerging use in mobile handsets, which also have severe power constraints, also indicates their feasibility for use in WSN node architectures. This chapter gives a high-level overview of our WFP algorithm and an explanation of the design choices made during implementation on the two platforms.

3.1 Wireless Fingerprint Algorithm

In this section, we present the steps required to measure and use a WFP in a network. This provides a framework for the implementations on each platform, which are presented in more detail in later sections.

The WFP measurement process consists of two distinct phases:

- Training Phase: In this phase, the neighbours of a node are discovered and their signals used to create a database of WFP templates. Each template is a compact

representation of the RF signal for a single RF source. An important requirement of the WFP template is that it is unique for each RF source.

- Classification Phase: In this phase, new signals are matched with the templates established during the training phase. A decision is made about whether the signal is likely to have been transmitted by one of the known neighbouring nodes, or by a new node which does not yet have a template in the database.

3.1.1 Training Phase

In the training phase, each WSN node creates and maintains a database of templates containing WFP information for each neighbouring WSN node. These templates are created and maintained based on measurements made via the RF interface. Initially, the WSN node has no entries in the template database. A neighbour-discovery process is typically used to establish communication between nodes in wireless networks. To reduce the amount of power consumed, the WFP template database should be built during the neighbour discovery process, with WFP training templates added for nodes as they are discovered.

A re-training phase is required for specific nodes if templates for particular nodes become invalid or are determined to be duplicates of other templates. The following conditions trigger a re-training phase:

- Aging, making a previous template obsolete.
- Corruption of a template with noise (malicious or otherwise) during the training process, making the template a poor fit during subsequent classification.
- Intrusion detection initiating the deletion (and subsequent required re-training) of a template.

For normal training operation and the rest of the following subsections, a one-to-one correspondence is assumed between the node IDs being claimed by a wireless source and the WFP templates for those IDs.

3.1.2 Classification Phase

In the classification phase, newly-arrived samples are compared with each template in the training template database. We distinguish between the following cases for the WFP-based classification of new samples:

- 'Matching' case: A single 'close enough' template exists in the database already.
- 'Unsure' case: More than one template is 'close enough'
- 'No Match' case: No templates exist in the database that are close enough. The implicit conclusion is that this is a new RF source.

The method for estimating the difference between the template and a sample depends on the specific WFP classification algorithm. The dimensions of the specific classifier form the basis vectors for both templates and samples. For our USRP1 work, we use an averaged Euclidian distance measure, calculated as the mean square error using different basis vectors for the template and for the sample being classified. The basis vectors correspond to the specific chip (single coded phase shift pulse) position in the preamble where residual phase errors are being measured (Section 6.2.5).

For the WSN platform, the basis vectors roughly correspond to the different histogram bin indices (Section 5.3) but we do not perform classification with them nor characterize classification performance in the same way we do with the more capable USRP1 platform. Instead, we collect results for the bin indices and analyze them statistically and graphically.

For authentication purposes, the WFP-based identity must be reconciled with the identity being claimed (or that has been authenticated using other mechanisms) in the message that is being carried in the RF signal. The 'first' time that an identity is claimed, it is linked with the corresponding WFP at the receiver in question. The WFP algorithm must detect when there are changes in the WFP portion of these linked entries during subsequent transmissions, as part of the intrusion detection system.

3.1.3 Training and Classification Sample Alignment

For both the training and classification phases, we need a consistent time reference for the received data. This provides a reference for the establishment of training templates and permits the appropriate constituent parts of a template to be applied to the corresponding constituent parts of a sample being classified. Even WFP methods that operate in the frequency domain (e.g. ones based on Fast Fourier Transforms/FFTs, Power Spectral Densities/PSDs) require this alignment. Failure to do so results in the addition of noise to the WFP templates at the training stage or during the classification process.

The method for aligning templates and samples being classified is different for our two platforms. For the WSN node, we take advantage of the AGC state change that occurs when an initial magnitude threshold is crossed to define our starting time. For the USRP1, we frame to the defined signal content existing within the IEEE 802.15.4 preamble.

3.2 Wireless Fingerprints Using Non-Transient Data

On the WSN platform, we focused initially on the samples corresponding to the transient portion of the RF signal, since this approach was motivated by the success of previous

researchers [**10**][**45**]. Because of limitations in our WSN platform architecture, access to evenly-spaced sets of samples of data near the transient proved to be impossible with the available software and hardware (Section 4.3.1). Williams et al. [**46**] show that different portions of an RF message (i.e. not just the initial transient) from a cell phone can be useful for distinguishing between different cell phone manufacturers. This drove us to look at more than just the transient portion of the RF signal. With the USRP1 platform, we use more of the sampled received data to increase both the accuracy and the consistency of our measured WFPs.

For the WSN nodes, we use the Automatic Gain Control (AGC) control loop circuitry to provide the inputs to our WFP algorithm. This circuitry is designed to operate on the preamble data and has full access to all IEEE 802.15.4 preamble symbol information [**47**], without requiring software intervention during the sampling process. The AGC outputs are also updated at reduced rates and, as importantly, these outputs are accessible from user software on our experimental platform. AGC loops are required for normal data reception and are present in most WSN receivers.

The AGC is designed to respond to the amplitude envelope of the newly-arrived signal as quickly as possible. Multiple input samples and averaging are used to stabilize the loop response. This prevents excessive gain changes and further reduces the required information processing bandwidth, which makes using the AGC outputs attractive for a practical WFP implementation on the WSN platform.

Our WSN node WFP algorithm is based on the timing characterization of the AGC output signals, which varies most strongly with the received signal. These outputs are active during the initial transient portion of the signal, but can also change during the

remainder of the preamble period, as the AGC adjusts the gain to provide an input at nominal levels. Further changes are disabled once payload data transmission starts.

Our USRP1-based method also uses the IEEE 802.15.4 preamble symbols. These are the same demodulated and sampled receiver data samples that are used as the inputs to the AGC circuitry. However, the USRP1 provides direct access to these samples. Our initial measurement results on the USRP1 using the input phase attributes of the demodulated data proved more reliable for WFP classification purposes than ones that were based on the magnitude attributes. Therefore, we focused our later research on phase-based methods, obtaining good results.

We did not synchronize the transmitters being characterized with each other, although the receiver must still synchronize to each received asynchronous signal. The clock recovery feedback loop used to do this synchronization has been observed to stabilize during the first few IEEE 802.15.4 symbol periods of a message, which is consistent with the IEEE standard [**47**]. The algorithms for clock recovery that we use are based on theory described in [**48**], but other methods are also possible [**49**].

The AGC and clock-recovery blocks are present in both platforms and are non-linear feedback loops with tunable parameters that have a filtering effect on the input. We use the 'standard' parameter settings for these loops. This is consistent with our objective of using our WFP algorithms in parallel with the 'normal' data reception processes. We believe that we are using 'normal' indoor operating conditions and 'normal' operating modes in all of our experiments.

3.3 Wireless Fingerprints Using Symbol Rate Information

Both our WSN AGC-based algorithm and our USRP1 algorithm use the digitized preamble symbol samples, which change at a 2Mchip/s rate, making the implementation feasible within the network node. WSN node receivers (and the USRP1) use analog high-speed RF circuitry to demodulate received data from the antenna and remove the RF carrier (see Section 4.1). After removal of the carrier and filtering with a nominal bandwidth of 83.5 MHz[1], the demodulated data is sampled with an Analog to Digital (A/D) converter. The sampling rate for this sampling process and for any further decimation after this conversion must be high enough to permit the baseband data to be recovered without aliasing, requiring further low-pass filtering in the software. We use a 4MHz sampling rate, which is twice the transmitter chipping rate for the IEEE 802.15.4 2.4GHz Offset Quadrature Phase Shift Keying (OQPSK) WSN node transmitters that we are testing.

Once sampled, further signal processing is accomplished digitally. For example, filtering for RF channel effects and AGC functionality are both digital functions on our WSN nodes. Digital processing of full RF information, with a bandwidth of 2x2.4 GHz=5.8GHz, is not practical on WSN or USRP1 devices.

On a typical WSN node, the full RF bandwidth is processed using a minimal amount of analog electronics. We believe that using analog electronics for WFPs is impractical for both power and space/complexity reasons. Even processing digital information at the

---

[11] SAWTEK 855916 SAW filter specification

[2] The IEEE 802.15.4 specification (Appendix A) gives -20dBm as the maximum input signal level to be tolerated and also specifies a minimum RX sensitivity of -85dBm, resulting in this total range.

[3]From left-to-right, the 'double-rate sampled' preamble pattern shown in Figure 18 and the rising vertical

raw sampling rate directly after the initial A/D conversion (16-32 MHz) needs to be minimized, because of the severe power consumption constraints for WSN nodes.

Our USRP1-based WFPs can be termed Baseband Wireless Data Fingerprints, in that they use sampled, demodulated baseband-rate information directly. A recent taxonomy of physical wireless identification methods [**50**] would term our approach a 'modulation approach', since it operates on demodulated data. Using this taxonomy, the majority of previous 'RF Fingerprint' work would be classified as 'transient data approaches', operating directly on the measured RF transient information with the associated large information bandwidth requirement. Digital processing of symbol rate information, with a bandwidth of 2MHz in our case, is feasible.

3.4 Comparison of Platforms

WFPs can use different physical attributes of an input signal. Table 1compares the physical attributes that are important for the WFP algorithms that we implemented on the WSN and USRP1 platforms.

Table 1: Significance of Input Signal Physical Attributes for Each Platform

Platform	***Timing***	***Amplitude***	***Phase***
WSN Node	Critical	Critical (AGC input)	Unused
USRP1	Minor	Unused	Critical

Because they are based on AGC responses, our SiLabs WSN node WFPs use amplitude-based input information, with WFP accuracy depending on careful timing of AGC outputs. Because we are using an IEEE 802.15.4 OQPSK transmitter when measuring WFPs for both platforms, phase information is a natural selection for algorithm input and was used for all of our final USRP1 algorithms (described in Chapter 6 and Chapter 7). Phase differences between samples require access to an uninterrupted stream of input

samples, not available on our WSN platform.

4 Chapter: SiLabs WSN Node Wireless Fingerprint Measurement

In this chapter, we present the details of our algorithm for measuring WFPs on the SiLabs WSN platform. The algorithm presented here is applicable to any WSN node. However, the methods are tailored to match the hardware and software capabilities of our platform and were motivated strongly by the capabilities of the hardware and software.

The first subsection describes the hardware architecture, on which the algorithm is based. The second subsection gives an overview of our WFP algorithm, which is based on the characteristic response of Automatic Gain Control (AGC) circuitry. The third subsection describes the software architecture and implementation of our WFP algorithm and other implementation details. The final section contains an analysis of the implementation constraints on the WSN platform.

4.1 SiLabs WSN Node Platform

Our first experimental platform is a representative IEEE 802.15.4 WSN node development platform made by SiLabs (Figure 1). This WSN node hardware is quite typical of other IEEE 802.15.4 WSN node hardware experimental platforms. WSN node hardware consists of the following main blocks:

- **Antenna**, attached with a coaxial connector.
- **RF circuitry**, to receive signals from the antenna.
- **Embedded microprocessor with Flash (type of non-volatile erase-able) memory** for the processing of data at higher levels than the physical layer.
- **Oscillators**, to power the processor (digital oscillator) and to provide the 'local oscillator' signal in the RF demodulator (RF oscillator). The digital and RF oscillators are separate devices for electrical noise control reasons.

- **External interface logic**, based on a Joint Test Access Group (JTAG) standard, for user FLASH memory programming, a Universal Serial Bus (USB) interface for user ASCII (American Standard Code for Information Exchange) text character-based I/O and some push buttons and LEDs.

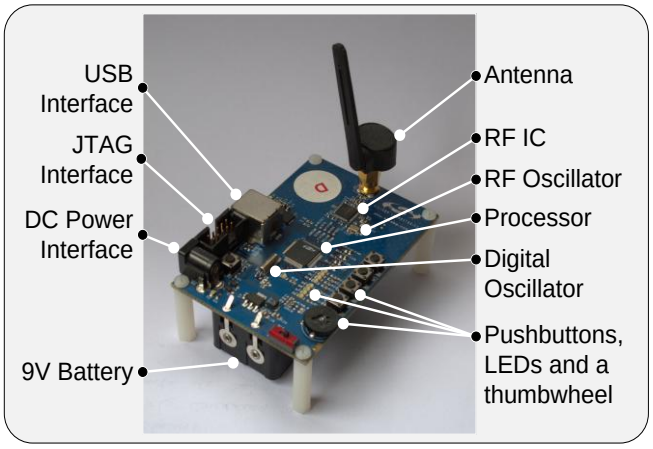


Figure 1: SiLabs WSN Node

The RF circuitry for our WSN node is almost entirely contained in a specialized integrated circuit (IC) augmented with a few external passive components. These components are mounted on the WSN Printed Circuit Board (PCB), with a critical layout that connects the RF electronics to the antenna in such a way that the electronic matching characteristics almost completely eliminate all unwanted signal reflections over the frequency range of interest.

4.2 Automatic Gain Control (AGC)-Based Algorithm

Our WFP algorithm uses the Automatic Gain Control (AGC) hardware and firmware feedback loop circuitry. The control logic of the AGC adjusts the gain used to amplify the input signal, using digital control outputs. The AGC keeps the Analog/Digital Converter (ADC) input at a constant magnitude even when the magnitude of the input signal from the antenna changes.

There are two gains stages controlled by the AGC on the SiLabs WSN node receiver front-end (Figure 2). The initial low-noise-amplifier (LNA) has three gain setting values and is followed by a Variable Gain Amplifier (VGA) with a higher resolution (128 gain settings). The gain of the I/Q mixers and filters cannot be adjusted.

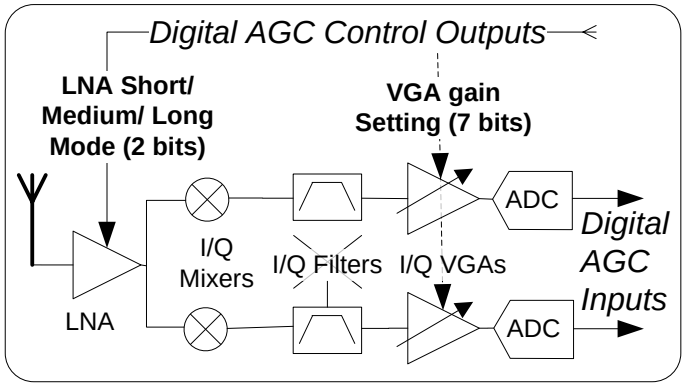


Figure 2: SiLabs WSN Node Receiver Front-End Architecture

LNA and VGA gain settings are adjusted to amplify the incoming signal by different amounts, keeping signal levels constant into the ADC. For long wireless transmission distances, a high gain setting is required. Little or no amplification is required for short range transmissions. The total dynamic range for the IEEE 802.15.4 specification is 65dB[2].

Initially, the AGC uses a larger amplification gain setting (i.e. it starts in a long-range mode). If an input signal is detected, it induces a large amplified signal at the ADC input that exceeds a detection threshold. The AGC loop reduces the amplification gain until the ADC input decreases back to the required level.

This adjustment must be done quickly enough during the fixed portion of the message so that the gain settles to the optimum setting prior to the beginning of the user message.

[2] The IEEE 802.15.4 specification (Appendix A) gives -20dBm as the maximum input signal level to be tolerated and also specifies a minimum RX sensitivity of -85dBm, resulting in this total range.

However, the adjustment must not be too quick, to avoid a situation in which small noise fluctuations on the RF channel cause large gain adjustments. Typical AGC loops are second-order feedback control loops, with parameters designed to trade off stability with responsiveness and operate over multiple input samples. After the adjustments during the preamble, the AGC does not adjust the gain settings again, until the start of the next message.

The AGC feedback loop is implemented in the firmware of the Texas Instruments CC2420 RF Integrated Circuit (IC) [**51**]. AGC control outputs and status are accessible through bit ports on the RF IC and also as registers which can be polled over the serial interface between the RF IC and the microcontroller. The RF IC output lines are assumed to activate with low jitter, but fixed latency in the RF IC. We observed three distinct adjustment strategies for setting the gains of the LNA and VGA, corresponding to different ranges of total amplification gain.

We define these three modes as:

Short range (1m): Little or no gain is required and the LNA gain is adjusted from high-gain (mode value of 3) quickly through medium gain (mode value of 2) to low-gain (mode value of 1). The VGA gain setting changes from a maximum value (near 7F) to a minimum value (near 0).

Medium range (4m): VGA adjustment is still not required, but the LNA gain changes from high-gain (3) to medium-gain (2) mode.

Long range (10m and greater): The LNA is maintained in high-gain (3) mode and the VGA is adjusted by appropriately small amounts.

Our WFP algorithm times AGC gain adjustment events in the three different modes of operation as an indicator of the amplitude envelope shape for a particular RF source. Accurate processor timing resources, with the resolution of a 96 MHz system clock cycle, in the embedded processor are used to measure the time of rising/falling signals of connected RF IC output pins (Figure 3). The output of these timers is used to generate interrupts inside the processor, so that the timer values are not over-written with new ones before they can be recorded.

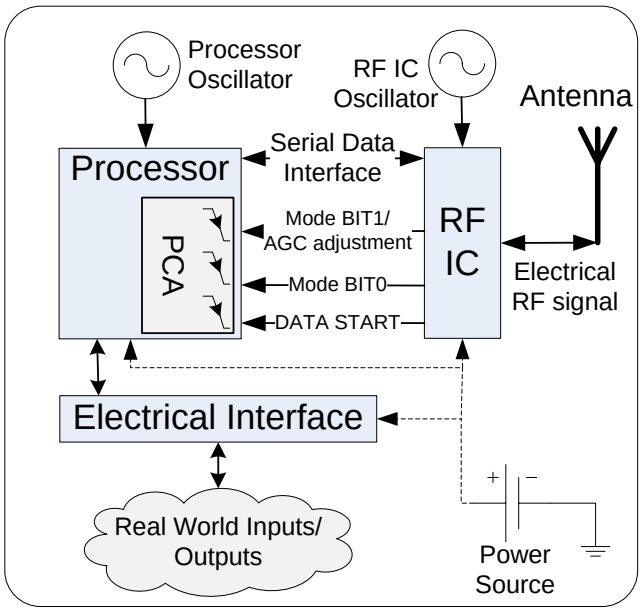


Figure 3: AGC Monitoring Paths on the SiLabs WSN Node

The precise timing of state changes in the AGC logic running in the RF IC is measured while the algorithm is operating. Combining these event time values with the full-resolution (LNA and second-stage) amplifier gain values obtained from test registers, polled over the serial interface, allows AGC performance to be characterized in real time. Figure 4 shows AGC adjustments for 90 messages transmitted over a 1m distance by two different RF sources ('B' and 'F') to the same receiver ('D'). Each curve represents the

samples for a single message. Time is shown on the horizontal axis and is normalized relative to the start of the message.

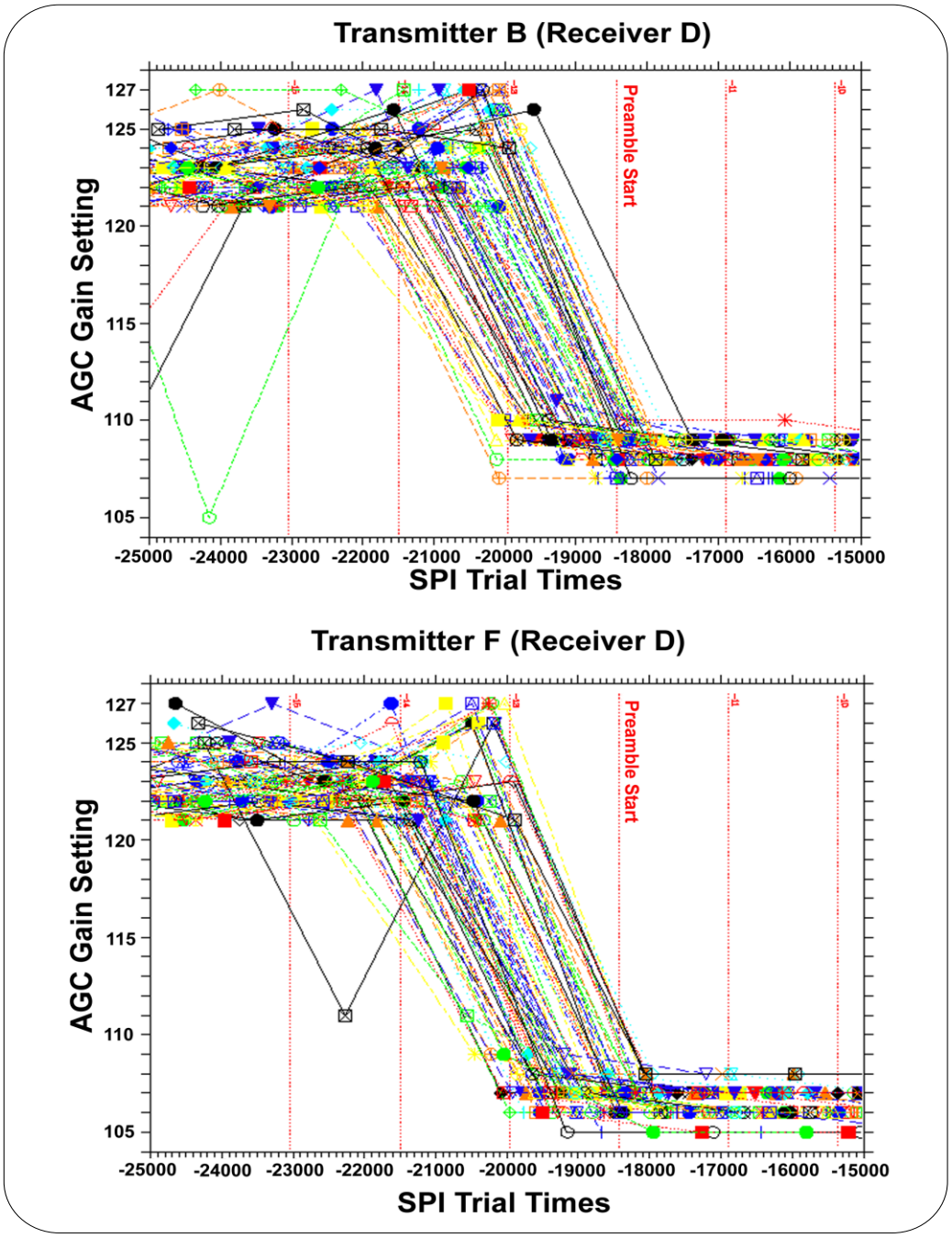


Figure 4: AGC Adjustment Events for Two Different RF Sources

The initial transient period and the first two symbols of the preamble are shown. The vertical axis displays the AGC VGA setting. The differences observed in the location of

the main falling edge for each gain transition and the slope of that edge motivated the algorithm (e.g. transmitter 'F' has steeper gain adjustments and a higher concentration of earlier gain transitions than transmitter 'B'). The implementation details of the algorithm are discussed in the next subsection.

4.2.1 WSN WFP Algorithm Pseudo-Code

```
Initialization Routines
While (forever) {
/* Main Loop */
   While (no incoming signal) {
   /* Sampling Loop */
      Poll RF IC VGA gain setting
      Check for timestamp updates
      Check for incoming signals
   }
   If (AGC adjustment event occurs) {
      Latch the AGC event time
   }
   Determine the validity of data bytes in received packet
   If (data packet valid AND event time within bounds) {
      If (training database at capacity AND source is trusted) {
         Delete oldest sample from Fingerprint database
         Decrement histogram bin for this removed element
      }
      Determine correct histogram bin based on event time
      Increment this histogram bin for the new element
   }
   else {
      Log false trigger event (Intrusion Detection System)
   }
}
```

Figure 5: WSN Node WFP Training Algorithm

The general pseudo-code for our implemented WFP algorithm for the training stage is given in Figure 5. The 'AGC adjustment' events in the pseudo-code are the gain transitions of signals generated in the RF IC and timed by the processor that we described

in the previous section. The particular signals used for event timing by the processor will change for the three different wireless range conditions.

WSN nodes have a finite amount of memory, limiting the size of the training database that they can store. Training algorithms for different applications can use different criteria to determine which (if any) templates to prune out once the database reaches this maximum capacity. The figure shows a very simple 'aging' criterion, deleting the oldest template when storing new ones.

4.2.2 WSN WFP Software Architecture

The 8051F121 processor on the SiLabs WSN node is programmed using the 'C' programming language.

The code consists of:

- A series of initialization routines that are executed sequentially (including the locking together of the system timers described further below),
- A main loop which executes forever and transmits information to the monitoring Personal Computer (PC) periodically via the Universal Asynchronous Receiver/Transmitter (UART) USB interface. Captured data from the sampling loop (see the next bullet) is partially processed in the main loop and then sent to the PC via the USB UART interface. The main loop also prints a periodic timestamp status message once every 32 seconds.
- A sampling loop (within the main loop) that executes continuously, collecting information from the RF IC over the Serial Programming (SPI) interface. The sampling loop stores samples on a continuous basis in a circular buffer, until an incoming radio signal is detected by the RF IC. Once an RF signal is received by

the RF IC, execution of this inner sampling loop is suspended and control passes back to the main loop. The basic structure of the sampling loop is:

  - SPI access (polled VGA gain values are stored in a First In First Out/FIFO data structure, with the oldest values being discarded first), followed by
  - check of the flag set by the Interrupt Service Routine (ISR) that runs when RF signals have been detected and their timestamp latched, followed by
  - check of the flag set by the ISR that processes a periodic timestamp initiated by the Programmable Counter Array (PrCA) ISR (every 32 seconds).

- ISRs that handle the different interrupts generated by the hardware resources in the 8051F121 in response to external events from dedicated test lines from the RF IC (i.e. SFD, FIFO, FIFOP and CCA)

Timing of signals is accomplished using the timer and counter resources available on the 8051F121 processor.

There are three main ISRs:

- PrCA_ISR: High-priority interrupt. This ISR increments the PrCA super-frame variable, which is used for the timing of all AGC adjustment events. This ISR is the most complicated routine in the sense that it must handle execution of each of the different PrCA resource blocks (e.g. for SFD timing and PrCA super-frame 'wakeup'/incrementing).
- Timer4_ISR: High-priority interrupt (but lower priority than PrCA). This ISR increments the 'slow timer' superframe variable used for timestamps.
- SPI_ISR: Normal-priority interrupt. This ISR processes activity on the serial SPI interface (e.g. when data bytes arrive from the RF IC).

4.3 Implementation Constraints

Our WFP algorithm discrimination is influenced by the embedded processor and the specialized RF IC architectures. We now examine the critical aspects and constraints for the architecture of these two devices and the restrictions that their limitations place on the WFP algorithm.

4.3.1 CPU/ Radio IC Communications Bandwidth and Jitter Limitations

The RF IC is connected to the embedded processor (Central Processing Unit or CPU) via a two-wire serial interface. The link is not fast enough to capture all significant AGC gain adjustment events for our WFP algorithm. Also, the variability in the arrival times for data over this interface increases timing variability for those events. Both of these decrease the consistency of WFP measurements, blurring differences between the WFPs created for different RF sources.

The speed of the serial interface can be adjusted up to 10MHz for the RF IC, subject to clock divider constraints in the processor, which has to match the selected interface rate. We ran the interface at 9.6MHz in our experiments. In an attempt to decrease the delay between measurements, we also experimented with over-clocking (at rates of 16MHz) although serial link reliability deteriorated and data errors occurred.

Because the A/D updates raw I/Q samples in an internal register at a rate of 2Msamples per second and each sample is each 8 bits wide, it is not possible to access the continuous stream of In-Phase (I) and Quadrature-Phase (Q) samples over the 8Mb/s serial interface. At the maximum specified 8MHz rate, combined with the serial addressing protocol and formatting overhead present on the serial line, the maximum rate at which data can be sent in real time is limited to about 500 ksamples per second (8 bits samples with about

100% overhead). We performed a sub-sampling study (e.g. at a sample rate of approximately 500kHz) of the I/Q data, but the variability of our results was too high. The jitter arising from the polling of registers over the asynchronous serial interface further compounds the delays that arise from the signaling overhead on the link. The serial interface timing is asynchronous to the Receiver sampling clock on the RF IC. Information on the RF IC is being sampled using the RF IC clock, requiring a handoff buffer to exchange information to the processor, which is running off the digital clock. For the baseband data processing path, both the speed performance bottleneck and asynchronous timing problem is solved using a data FIFO buffer inside the RF IC. The FIFO buffer allows full capture of all received data bits, without processor involvement as each byte is received. The processor interface to the RF IC is also not required to run fast enough to keep up with the real-time storage of received baseband data. There is no requirement to log and process samples simultaneously, so the two processes can be performed serially.

This type of FIFO buffer is not provided for the raw I/Q samples nor for the VGA setting history. The resulting jitter and delay effects on these measurements cannot be avoided when polling registers for WFP measurements over the serial interface. Our solution is to use the bit port interrupts between the RF IC and processor, which are captured with dedicated and asynchronous interrupt circuitry. This is not an option for the 7-bit VGA settings, but works for the 2-bit LNA settings.

4.3.2 Timer Resolution

The accuracy of measurements of the AGC adjustment event time for the WFP algorithm depends on the accuracy of the timing resources on the WSN node. In our

implementation, events can be timed with an accuracy of up to 10ns. This resolution corresponds to the fastest digital clock available on the hardware and cannot be optimized further. However, digital clocking is asynchronous to the analog clocks from the RF IC, adding jitter in handoffs of data between the two clocking zones. This jitter results in reduced discrimination capability for our WSN WFP algorithm.

In our implementation, the slow (Timer4) counter and the high-resolution (PrCA) counter are used together for accurate time-stamping. The hardware architecture of the 8051F121 allows very accurate timing resolution (within one 96 MHz/10 ns clock cycle) and time-stamping of events. In spite of this, ambiguities arise because the processor timing is asynchronous to that of the RF IC. For bit outputs of the RF IC, this is less of an issue since the error will be on the order of a single clock cycle. However, as already discussed, random jitter is introduced on the measurements made using the clocks of the RF IC and reported over the serial interface.

For analysis and debugging purposes, we capture the arrival times for each new message. We are able to correlate PC logging time easily with the measured time inside the processor on the WSN node by storing the PC timestamp and the processor timestamp in the file. However, events latched inside the RF IC have a variable timing resolution which is not under our control. Without knowledge of the timing architecture of the RF IC, it is difficult to estimate the bounds of timer resolution and error for events that occur there and that are reported via register access to the processor using the serial interface.

4.3.3 CPU Processing

When an external event causes the AGC to change the LNA gain state, the corresponding interrupt service routine on the CPU runs, stores the timer value and then clears the

interrupt, ready for the next message. The processor is not capable of parallel processing, so polling and further interrupt activity is paused while the interrupt is handled. The UART Interface on the processor is connected to an RS-232 terminal port and generic terminal data logging software on a PC. The WFP data and timestamp information is sent over the UART interface to the PC for logging.

Such logging is only required for experimental purposes. In a complete implementation of WFPs on the WSN node platform, WFP processing will consist of the creation of templates during the training process and the comparison of the newly-arrived data with those templates during the classification process. This replaces the real-time logging task in our experimental code, but still requires the node to be 'offline' for a short time. The effect on WFP algorithm discrimination will be potentially missed messages for authentication unless WFP-specific data buffering is included.

4.3.4 Transmission-Range Dependent Gain Adjustment Events

Gain adjustment events are reported using different signal paths that depend on the specific mode for the adjustment (defined in Section 4.2). When the serial interface is used, more noise will be added to our WFP implementation, resulting in fewer observed differences between distinct transmitters. However, if bit ports into the processor are used, the algorithm will discriminate better between RF sources.

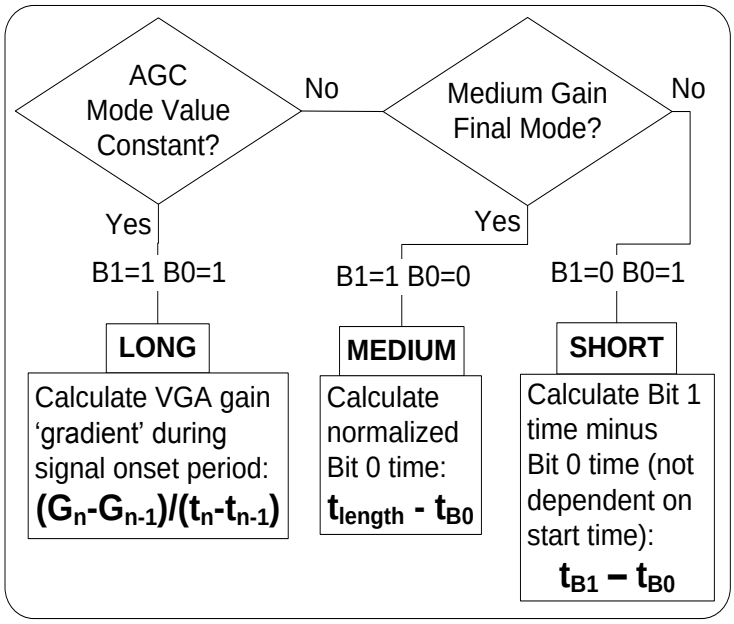


Figure 6: Flow Diagram for WSN Node AGC Adjustment Events

The flow diagram used to determine the 'AGC adjustment' event used in the WFP algorithm is shown in Figure 6. G_{n-1} and G_n represent the VGA gain setting values sampled at 'Data Start' time (t_{n-1}) and 'Preamble End' time (t_n) via the serial interface. The values of t_{B0} and t_{B1} represent the time of the edge change for the LNA Gain mode bits. The bit settings for the different gain modes are specified in Table 2. While waiting for a signal input, the initial AGC LNA gain mode is set to High. As the signal level rises, the gain is adjusted down to Medium and then to Low. This intermediate change results in signal edge changes in both Bit 0 (B0) and Bit 1 (B1).

Table 2: LNA Gain Modes

LNA gain Mode	*B1*	*B0*
- Unused -	0	0
Low	0	1
Medium	1	0
High	1	1

For a transition from High to Medium, only B0 will change, requiring us to measure the time from 'Data Start' to the B0 edge change (t_{length}). For long range transmissions, there will be no change in the LNA gain mode and only the VGA gain will be adjusted. For our hardware, it is also possible to adjust the point at which these different gain modes change (in 2dB steps) as well as the amount of hysteresis for the changes. We varied these thresholds and experimented with different values in an attempt to improve the discrimination results, but without much success.

WFPs could change over time, if nodes are moved or maybe for other reasons. If so, the retraining process could be repeated, where older samples are discarded in favour of newer ones. These newer samples would then be used to recalculate the histograms for the different timing range bins, updating the templates for the RF sources in question. The disadvantage of using such an 'aging' approach for templates is that an attacker could overwhelm a database for a specific claimed ID. Therefore, a throttling mechanism is required for the updates of the template database, which must be co-ordinated with other trust systems or intrusion detection mechanisms operating on the WSN.

4.3.5 Summary

The sampling process over the serial interface adds enough variability to make detection of outlier measurements difficult. Accurate timing information for signal changes is hard to determine on the processor using the serial interface alone. To reduce time measurement jitter and obtain more accurate results, we use dedicated serial lines between the two chips instead to activate interrupt-based timer resources on the processor.

This is accurate to the granularity of a few 96 MHz processor system clock ticks, corresponding to the interrupt latency and variation in the processor. Unfortunately, this only applies to the short and medium range cases, where AGC gain adjustments are the largest and the measurement uncertainty associated with detection of a signal above the noise floor is also at its lowest.

We collect these results and analyze the differences statistically and graphically. Using our AGC-based method, long-range WFPs can be expected to be poorer than short-range ones. At the limit of wireless transmission range, AGC adjustments are no longer made at all and the gain setting is maintained at a maximum level, rendering the algorithm useless for these extreme cases (since the gain change will be zero).

The IEEE 802.15.4 2.4GHz standard specifies the use of Direct Sequence Spread Spectrum (DSSS). With DSSS, the problem is compounded further since the signal power is spread out over time using coding techniques, even below the noise floor. This results in very small changes of average power when RF signals first appear at the receiver. In general, WFPs can be expected to degrade as the transmission distance is increased and this is especially true with WFPs based on signal amplitude characteristics in noisy RF environments, where signal level changes become imperceptibly small.

5 Chapter: SiLabs WSN Node Experimental Results

This chapter presents and analyzes the experimental results obtained on the WSN node, showing that discrimination is possible between RF sources. From this work, we identify the main parameters to be studied later for our WFP algorithm implemented on the more flexible USRP1 platform.

5.1 Experiment Design

The experimental setting is a multi-storey residential dwelling with concrete and plaster wall construction. Measurements are made over a distance of 3m with LoS visibility between six transmitter/receiver nodes. This distance is short enough that the LNA gain bits change at the beginning of each received message. Each node transmits 500 consecutive messages to all five other receivers, which are logging concurrently. The same locations were used for the transmitters and receivers, but a specific node is not always receiving signals in the same position.

For each message sent, the VGA setting values polled over the serial interface on either side of the LNA gain adjustment event are recorded using a PC logging tool, concurrently, for all receivers. The exact times of the changes in the LNA gain mode bits are recorded using the PrCA bit timing method described in Section 4.2. The change in VGA gain setting is approximated using a straight line equation derived from the two sampled gain values and the measured times of their arrival measured from the serial interface message arrival times.

The midpoint of the VGA gain change on that line is set between the two outermost sampled values, and the corresponding time value for the gain change are determined using the estimated line equation. We use the term 'centroid' to identify the time

corresponding to the middle of this main gain transition. We estimate the centroid with a straight-line approximation using the sample values and occurrence times on either side. Histograms of the time of occurrence of the VGA gain change either side of the LNA gain transitions are calculated and plotted.

5.2 Experimental Results - Discrimination using AGC Information

The experimental results in this subsection show that there are observable differences in the VGA adjustment patterns for different RF sources, but there is also considerable variability in the results. VGA gain adjustments for several messages from the same source (node 'B') and same receiver are shown, each with a different colour/ marker combination, in Figure 7.

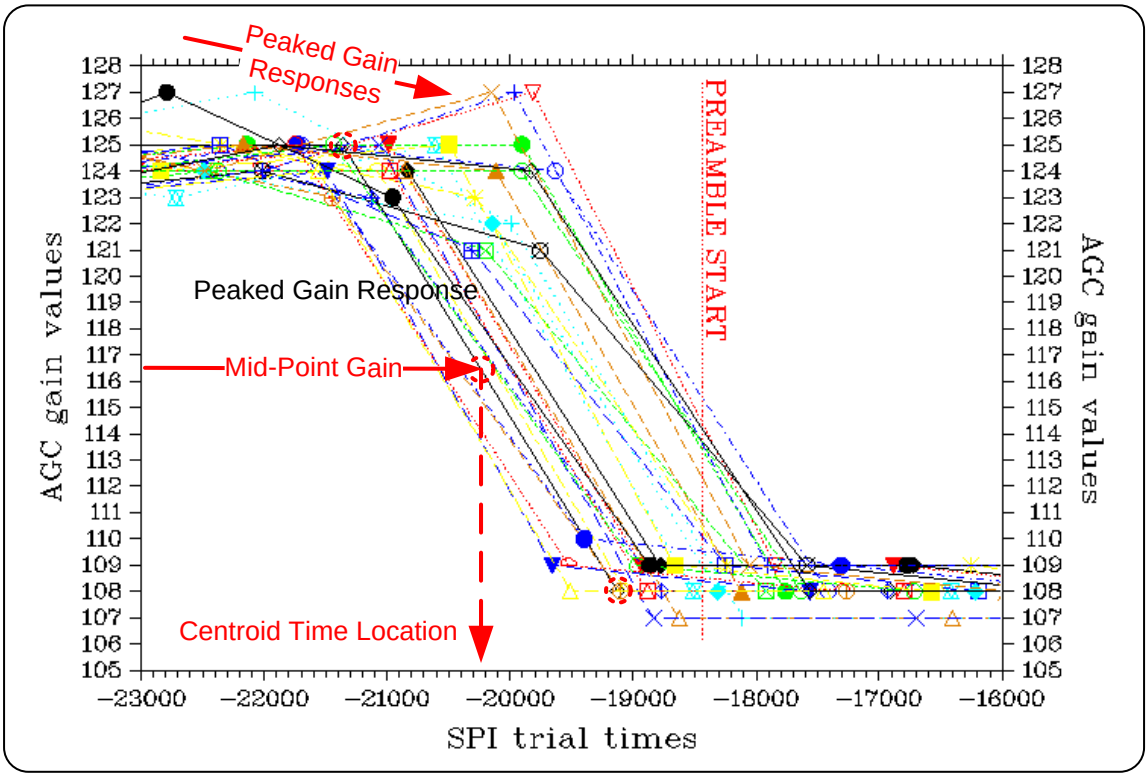


Figure 7: Initial AGC Gain Adjustment

The 7-bit gain value is initially near its maximum value of 128 before being adjusted to a

new value of about 108, which occurs just before the first preamble byte start, shown by the red dotted line. This adjustment occurs immediately after the RF transient period. The AGC continues to make smaller adjustments during the remainder of the preamble. The preamble start time is estimated using the measured time of assertion of the status signals (e.g. SFD and FIFO) coming from the RF IC with a calculation to subtract the known delay for the preamble, SFD and length bytes, as specified in the IEEE 802.15.4 standard (Appendix A). Although samples are measured at the 'maximum rate' of the serial interface using software tuned for the purpose, we see an average delay/jitter between the samples of about 2000 96 MHz clocks (about 20 μs) caused by the timing constraints described in the previous chapter.

In Figure 7, the mid-point gain setting for the message samples highlighted with dotted red circles is 116.5 (between 108 and 125) and the corresponding centroid value would be about 20,200 (measured in units of 96 MHz clock ticks before the start of the payload). The variability in the exact timing of the centroid has a similar 20 μs spread to the SPI sampling rate interval. There are only a few samples which occur near the centroid, indicating that the gain value transition is adjusted quickly inside the RF IC AGC algorithm.

There is also a different profile to the value of the gain change for some messages, with a 'peaked' gain response. An initial increase in gain occurs before the gain finally decreases to the new lower stable level. This level of peaking varies noticeably with different RF sources. We believe that this peaking is due to an interaction between the AGC logic and the properties of the specific signal.

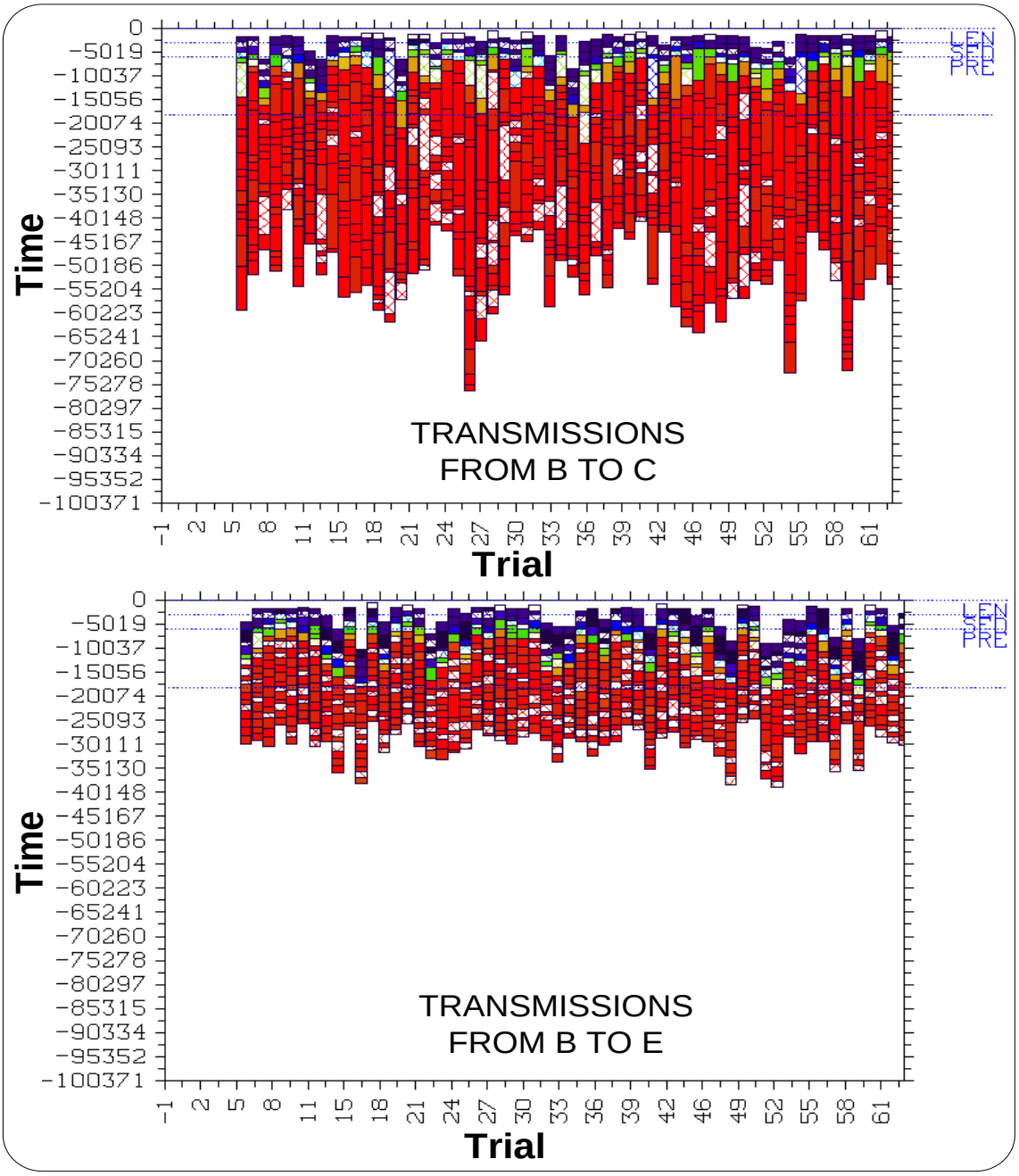


Figure 8: Differences in AGC VGA Adjustments by Receiver

There also appears to be variation in the starting time of the gain adjustment with different receivers. This is seen in Figure 8, which shows gain setting changes, where the bar colour is proportional to the gain setting, organized in a spectrum with blue colours representing small gain values and red colours representing larger gain values. The graph

shows 60 individual trials for two different receivers ('C' and 'E') listening to the same transmitter ('B') over the same nominal RF transmission range. The time axis in the figure is vertical and the trials are separated from each other horizontally.

Each bar segment represents a change in the polled VGA gain value. Cross-hatched bars represent values which were interpolated because the LNA gain mode bit changed but the VGA value did not appear to have changed yet. The IEEE 802.15.4 byte locations are shown on the far right, where 'LEN' represents the length field, 'SFD' the SFD field and 'PRE' the preamble field in the IEEE 802.15.4 frame (Appendix A).

At receiving node 'E', large gain adjustments (changes in colour from navy blue to red) are often made in the middle of the preamble field, while at node 'C', the large gain adjustments are usually completed by the end of the SFD byte. We theorize that the variability in adjustment times depends mostly on the sensitivity of the particular receiver, since the profile stays relatively consistent even when the node is tested in different static positions.

5.3 Experimental Results- AGC Gain Transition Histograms

To simplify the use of WFPs in a network, we would prefer to find differences in gain transition times that are specific to the transmitter only and that are independent of the receiver. Failing this, we want to be able to remove any WFP variability that is due to the specific receiver. We now analyze the differences between the discrimination results that are due to the receiver. Because of the variability in the results shown in the previous section, we use histograms for this analysis.

We create histograms for the AGC centroid values like the one shown in Figure 9. Gain adjustment centroids are not uniformly distributed for a specific transmitter/receiver pair.

For example, histogram bins centered at 15628 and at 4645 clock periods before the payload start in the figure occur more frequently than others.

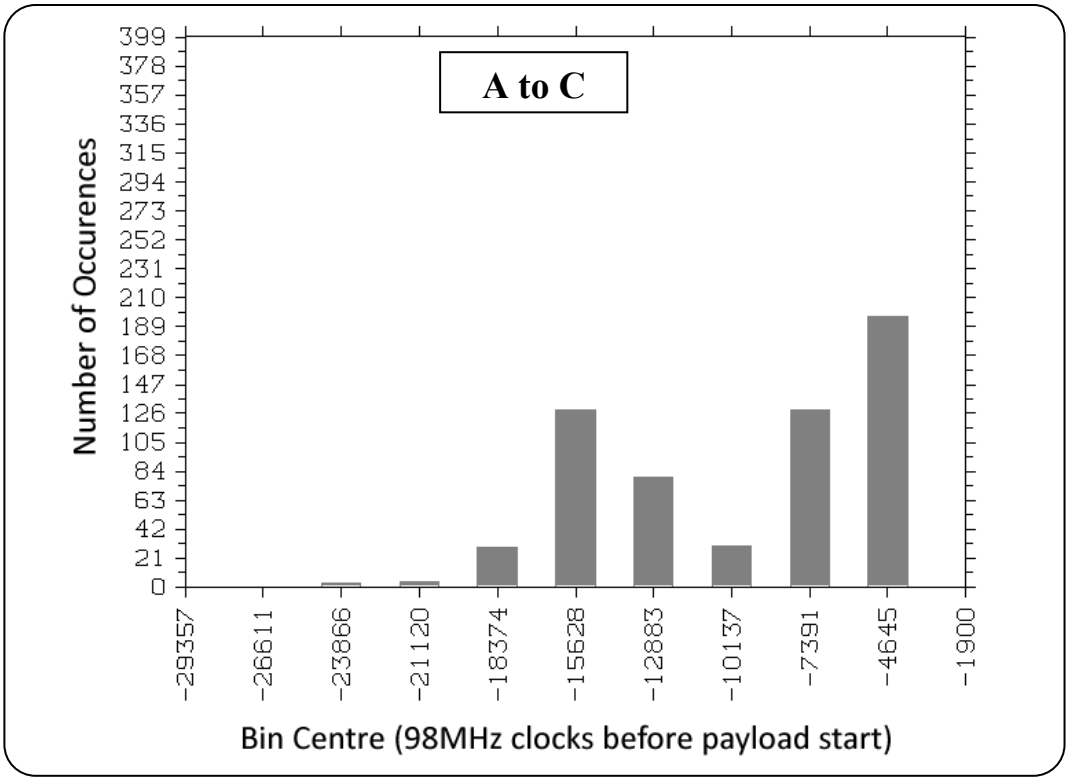


Figure 9: Histogram of AGC Gain Transition Centroid

The distributions of the main gain transition appear to vary systematically and in a characteristic fashion with the RF source, making discrimination between sources feasible in a specific receiving device. However, there is significant variation for each transmitter/receiver combination as seen in Figure 10. The histogram distributions remain relatively stable over time, but small changes are observed and the results appeared to vary periodically.

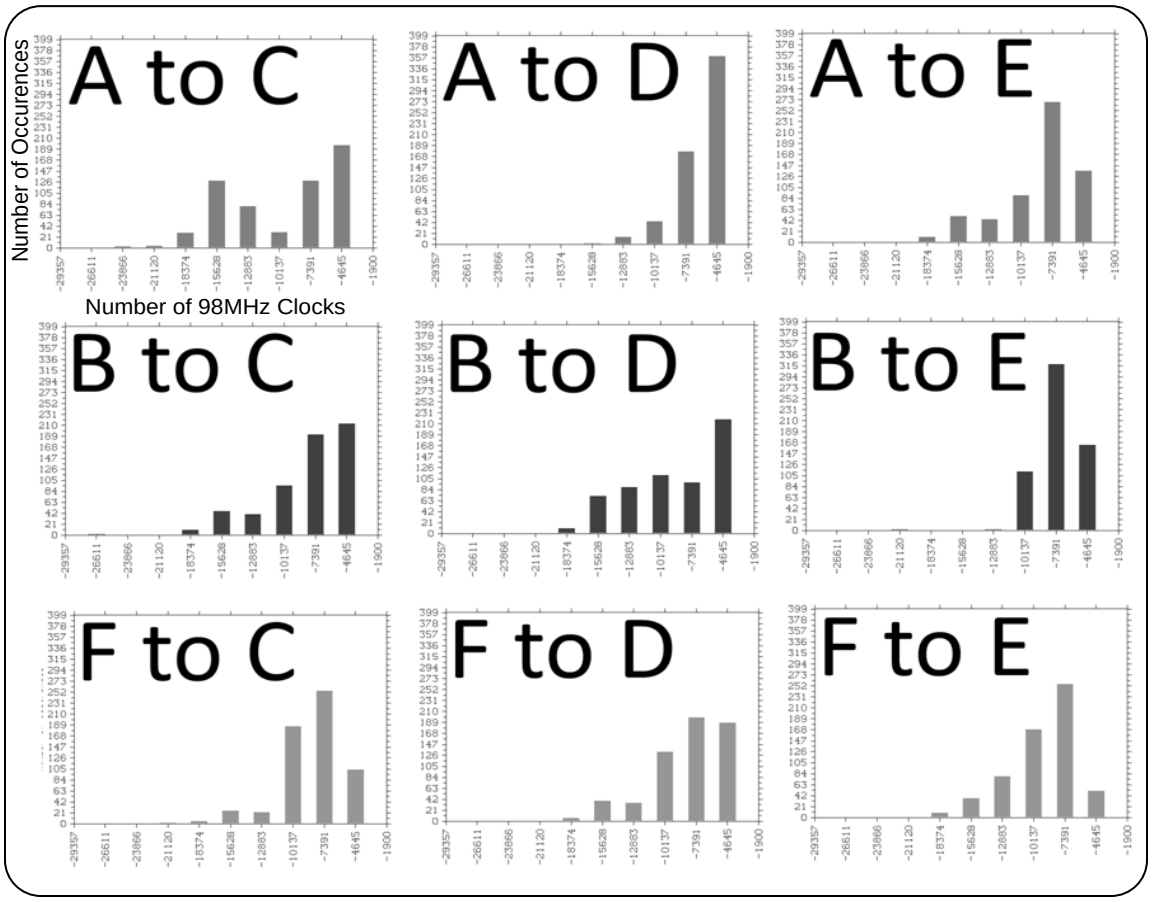


Figure 10: AGC Centroid Histograms for Transmitter/Receiver Pairs

Certain transmitter/receiver combinations have centroid histograms that 'resemble' each other. For example, transmissions from node B to E are quite similar to transmissions from node F to C in Figure 10. To improve the resolution of the histograms, we incorporate an estimate of the gradient of the gain change at the centroid into the histogram binning process. This gradient is calculated as the difference in the gain values divided by the difference in the time values at those gain values. We define a 'composite bin' as a specific combination of gradient and centroid time value (Figure 11).

The composite bin index value is determined as follows:

Minimum Centroid value $= C_{\min}$
Maximum Centroid value $= C_{\max}$
Maximum integer number of centroid bins $= n_C$ (n_C = 8 in our example)
Minimum Gradient value $= G_{\min}$
Maximum Gradient value $= G_{\max}$
Maximum integer number of gradient bins $= n_G$ (n_G = 6 in our example)

$$\text{bin index value} = n_G * \left\{ n_C * round\left[\frac{C - C_{\min}}{\left(C_{\max} - C_{\min}\right)} \right] \right\} + n_G * round\left[\frac{G - G_{\min}}{\left(G_{\max} - G_{\min}\right)} \right]$$

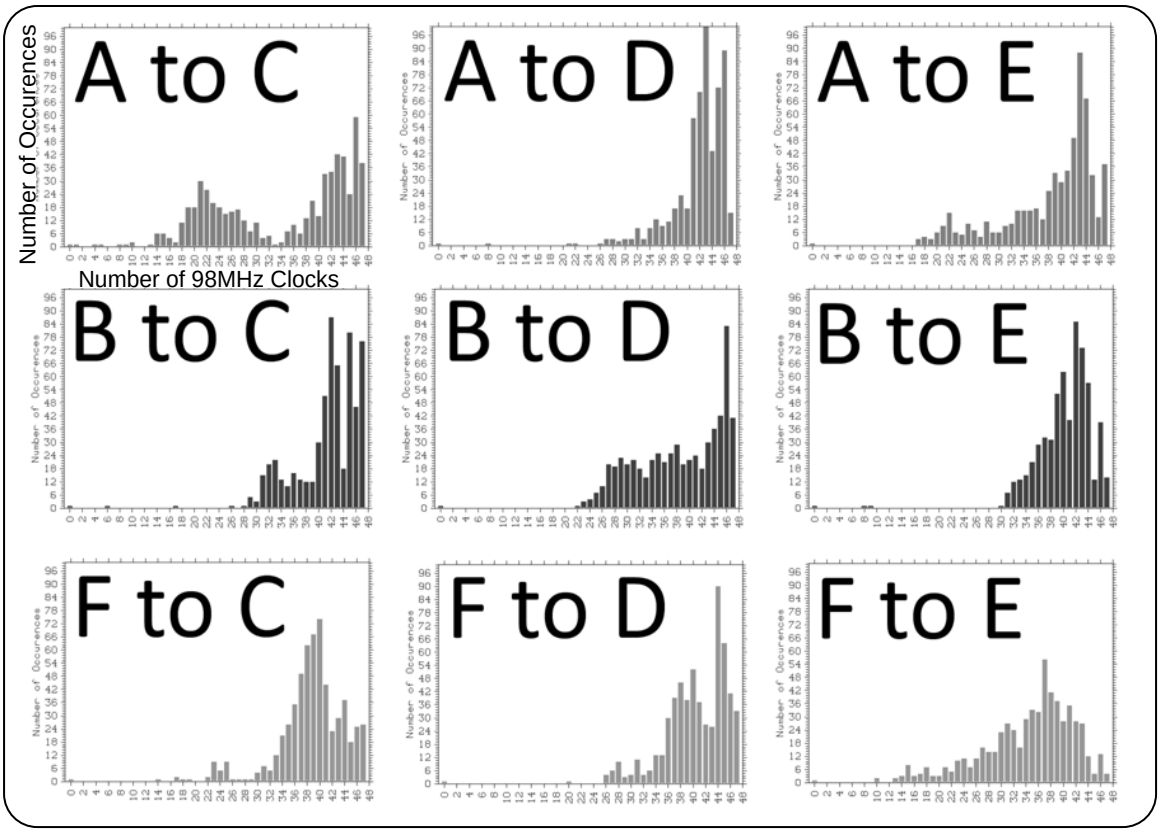


Figure 11: AGC Composite Histograms for Transmitter/Receiver Pairs

The composite bin histogram values are comparable to the centroid histogram values when the gradient values are at their minimum. At these points, they have a value equal to the centroid histogram value, but scaled by the total number of gradient bins.

We define six active gradient bins, giving a new total of 48 active composite bins. The

choice of n_C and n_G depends on the total number of observations being made and the number of nodes being classified. We chose n_G to be the same as the number of known nodes and kept n_C at the same granularity as was used for our previous centroid-only analysis. Our sample size of nodes being classified is small enough that better guidelines than this are difficult. Too many empty bins would indicate a poor choice of these parameters, but ours are good enough.

The composite binning algorithm can be used in the algorithm pseudo-code of Figure 5, instead of the centroid binning. This results in an increase of the number of total histogram bins and approximately squares the overall calculation and storage complexity, depending on the specific choices for the number of bins.

Comparing Figure 10 and Figure 11, the basic outline of the histogram shapes are similar for the same transmitter and receiver pairs. However, the composite histogram bins give a finer granularity of characterization than just using the centroid alone. Essentially, both parameters of the straight-line approximation have now been incorporated into the binning process. We could have used the estimated gradient and intercept of the straight-line approximation itself, but using our time axis is a more stable reference than the gain value mid-point, since it does not vary with the RF transmission range conditions.

We investigated the fitting of cubic splines to the AGC gain setting data points, but the results were not significantly better than a simple straight-line approximation. It is rare for more than two 'significant' points in the range of the transient to be available for accurate curve parameter estimation and so more than one cubic spline can be fitted to a pair of data points. We do not get much improvement, even if more points are used to

constrain the splines further. Extra points do not vary enough to influence the shape of the spline, appearing only as constant gain values either side of the transition.

In spite of the impediments introduced with the serial interface, our experiments show that characteristic responses for distinct transmitting devices are observable. However, these differences are only usable in the form of aggregate statistics, with individual measurements varying considerably about the average value. Ideally, we want to define a WFP that discriminates the source of individual messages correctly without having to group them in batches for identification.

5.4 Experimental Results –Stability Over Time

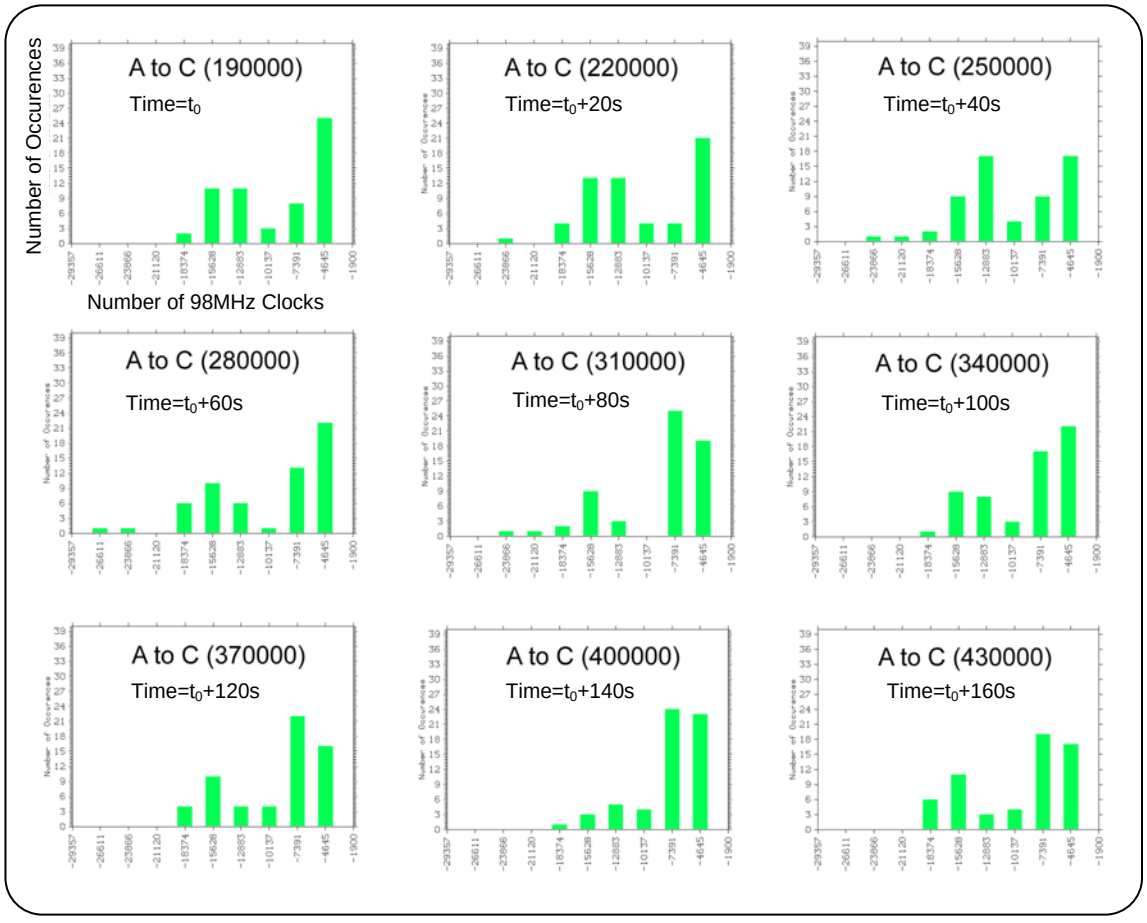


Figure 12: Time Sequence Histograms of AGC Gain Change Centroids

Next, we analyze the stability over time of the histograms used in the previous section

and show that they vary even over short time intervals. Figure 12 shows a time sequence of histograms for the centroids of the AGC gain transition for a series of transmissions from node 'A' to node 'C'. The axes all have the same scales, allowing qualitative comparison and reducing the importance of the exact numerical values.

Each graph in the sequence is separated by 30000 lines in the log file, which corresponds to a time offset of about 20 seconds. The total duration of the sequence shown is about three minutes. There is a noticeable change in the histogram shape during this time.

A scatter plot of the gradient (Y axis) and centroid (X axis) information and variation over time is shown in Figure 13. The colour (Z axis) corresponds to the Y intercept of a straight line with the specified gradient and centroid.

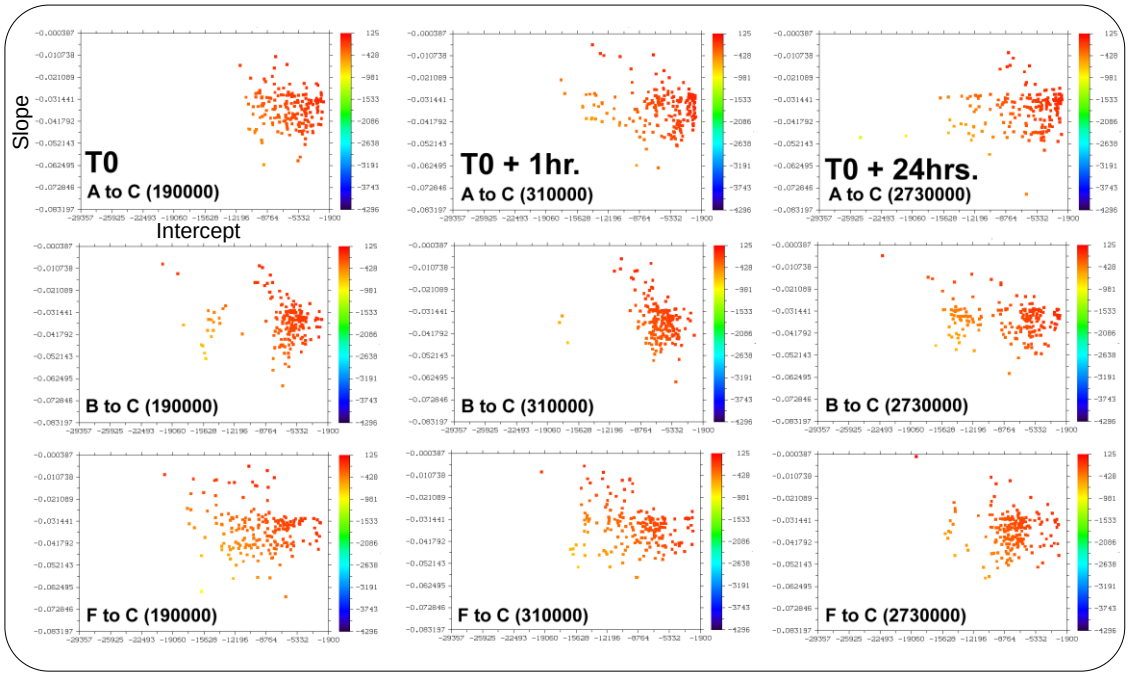


Figure 13: Scatter Plots of AGC Gain Change Gradients

Because the gradient and intercept are related with a straight-line approximation, a specific colour is always plotted in the same location on the graph, but is included to help with comparison. The data from Nodes 'A', 'B' and 'F' received by node 'C' are shown in

the first, second and third rows, respectively. Differences in the centroid and gradient values can be seen for the different transmitters.

The statistics of the gradient remain relatively stable over time for a given transmitter/receiver pair, but some variation is apparent, especially after 24 hours. We believe that the reason for this variation is related to the asynchronous nature of the receiving and transmitting clocks in the network and the relative frequency of the oscillators in question.

In our testing, we used the non-synchronous mode of operation specified in the IEEE 802.15.4 WSN standard, where the oscillators used for transmission in each WSN node run freely and independently of each other. In the IEEE standard, oscillators are also allowed to have slight differences in their nominal frequency (see Appendix B). Because the received RF input is re-timed using a clock derived from this same free-running oscillator, the relative clock phase of an input signal and the clock being used to sample that signal will change continuously. The gradual and seemingly periodic time variation that we observed in the centroid histograms is probably due to this changing relative phase difference. The phase difference varies very slowly during a message or between nearby messages but much more greatly between messages that are further apart in time. Another possible cause for the observed measurement variation is changes in the RF environment noise level over time. Without a more detailed knowledge of the AGC implementation, determining the precise root cause and improving WFP measurement consistency further is difficult.

We assume that the variation in AGC response for different transmitters is due to the variations of the sampled signal amplitude at the receiver, which includes contributions

from the RF channel and interaction between the noise performance of the receiver and the noise on the RF channel. These receiver and RF channel-induced variations are independent of the transmitter whose signal is being 'recognized'.

An AGC circuit is designed precisely to detect signals reliably in the presence of RF channel noise, while optimizing response time and stability during the signal detection process. Therefore, WFP consistency for different noise environments is expected to be automatically optimized during the AGC design process, which must strike a balance between maximizing sensitivity to signal amplitude variation while avoiding false triggering adjustments caused by noise.

From the experiments in this section, we show that different channel conditions and internal noise levels and timing affect a given receiver's AGC response over time. The response for a specific transmitter is not stable over time in a given receiver. This makes the use of our algorithm in a real network for authentication purposes less feasible, although the method still has value as part of an intrusion detection system. In an intrusion detection system, guarantees of authenticity are not required and evidence of authenticity is still useful.

We need a mechanism to provide access to the AGC state machine with less jitter than occurs with user software polling over an asynchronous serial interface. For example, direct access to the VGA gain setting via signals would be ideal, allowing use of the accurate processing timing resources. As an alternative, a time-stamped handoff buffer (similar to the one provided for data) would serve our purposes.

5.5 Determination of Parameters for Further Study

With the knowledge gained from our WSN node experiments, we switch our attention to the WFP algorithm for the USRP1 platform. We need a WFP to be more robust than our WSN implementation in order to classify individual messages. Based on the insights gained with our WSN platform, we now define the independent variables for our experiments on the USRP1: receiver stability, RF channel stability and time stability.

5.5.1 Receiver Stability

For networking purposes, we are interested in whether WFP measurement is affected by variations in the receiving device. If templates are the same at different receivers, then direct comparison at a network level is simplified. If not, then network-level techniques are required to reconcile them. If classification errors made with local templates are independent at different receivers, then there is an advantage when nodes collaborate to make a decision (see Section 8.5 and Section 8.3). If errors are highly correlated, using multiple nodes to reconcile and improve the reliability of WFP decisions is less effective. Filtering receiver-dependent noise requires an understanding of the noise processes inside an RF receiver will vary with the methods used for WFP generation. For example, we would like to reduce the amplitude noise in the receiver for our WSN node algorithm, but reduce the different sources of receiver phase noise for our USRP1 algorithm. Because the sources of receiver noise have probably been reduced to functional and practical levels by the RF device and PCB designers, further reduction is probably impractical. However, we can compare the results of two co-located SDR devices or use identical receiving locations, when measuring the same RF signals to quantify the noise effects.

5.5.2 RF Channel Stability

We are interested in the effects of RF channel variation and transmission range variation on the WFP measurement process. This has a direct effect on how WFPs can be used at a network level and on the restrictions for node mobility. We use the phase attributes of the wireless signal to derive our USRP1-based WFPs. Therefore, we are concerned with RF channels that can distort the phase information of the signal. In indoor environments, large reflecting surfaces cause strong multi-path signal propagation that can distort the RF signal phase and our WFP measurements. Doppler shifts can be created by transmitter movement or can result from the different components of the multipath propagation of the RF signal, as the RF signal interferes with itself [**52**]. Measurements made over smaller intervals of time should be consistent, however, since there is a smaller chance that reflecting objects in the environment move significantly.

The time-variance of RF channels has been studied and RF channels have been characterized in terms of their coherence time and their coherence bandwidth. An RF channel can introduce delay spread into received signals because of the different delays taken by the signal, causing signal time dispersion. Frequency dispersion also causes the received signal to have a larger bandwidth than that of the transmitted signal.

Wireless signals typically have a fast Rayleigh fading characteristic at the receiver, unless there is a LoS path, where the distribution changes to a Rician fading model. A Rician fading model is essentially the same as a Rayleigh fading model, with the addition of a strong propagation mode corresponding to the LoS transmission path.

5.5.3 Time Stability

Ideally, a WFP would not change over the entire lifetime of a WSN node, allowing it to be measured once and then recognized forever afterwards. However, this is unlikely with actual wireless signals. Short-term time stability is not only easier to characterize, but is a more basic requirement for WFPs, since long-term variation can be handled with WFP aging and refreshing protocols. Our WFP algorithm is based on circuitry that is functioning in its normal operating region. Performance here should be stable over time.

6 Chapter: USRP1 Wireless Fingerprint Measurement

In this chapter, we describe the basic architecture of the USRP1 and our WFP algorithm design. Unlike our previous work with the SiLabs WSN nodes, the use of a particular device like the USRP1 is not an essential requirement for the algorithm implementation. Other SDR platforms (or WSN node platforms that provide adequate contiguous sample access) can be used.

6.1 Ettus Research Inc. USRP1 Platform

The Ettus Research Universal Software Radio Peripheral (USRP1) [**43**] is a public Software-Defined Radio (SDR) hardware platform manufactured by Ettus Research that was designed to work with a large public Gnu Radio software base [**53**]. In combination with the Ettus Research RFX2400 daughter card [**54**], this platform has the same zero-IF type and speed of RF Receiver front-end design as the WSN nodes that we are using, making it a representative platform for future WSN nodes.

The USB interface has a maximum sample rate of 16MHz and is the main bottleneck for real-time radio signal information to and from the PC. Current WSN node technology limits our ability to process and store data above 2MHz rates, as discussed in the previous section. Therefore, we do not exceed the bandwidth of the USB interface with our algorithm and the platform does not limit our performance.

6.1.1 USRP1 Hardware Architecture

The USRP1 receiver hardware (Figure 14) consists of the following main blocks (see Appendix C for further details):

- **Antennas** and coaxial connectors and cables,
- **RF daughter card(s)**, containing all basic elements of a zero-IF RF interface,

except the A/D devices. Different daughter cards are manufactured for the USRP1; we use the RFX2400 [**55**] for compatibility with our WSN nodes. The AGC functionality is also implemented here.

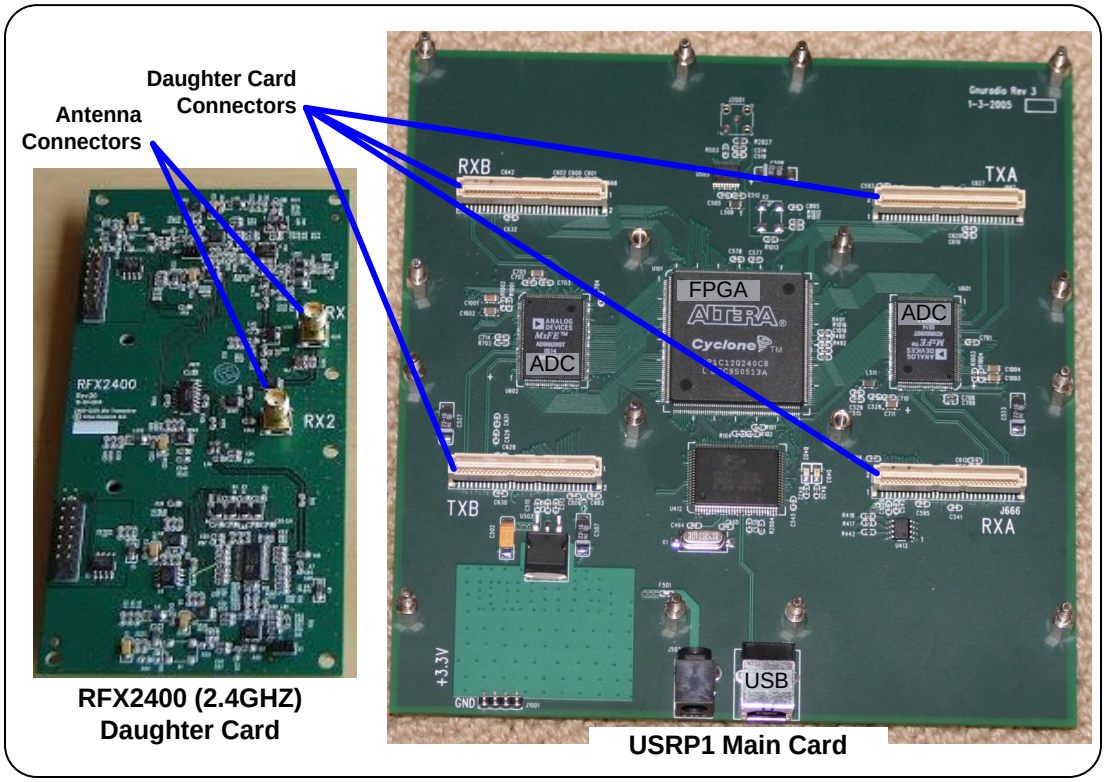


Figure 14: USRP1 Hardware Architecture

- **Analog/Digital (A/D) conversion devices**, which convert smoothly-varying real-world signals into ones that are usable by software algorithms.
- **User-programmable Field-Programmable Gate Array (FPGA)**, which is responsible primarily for connecting the correct antenna signals to the A/Ds and for converting the demodulated I/Q data samples into packets suitable for USB transfer.
- **USB interface**, providing the connection between the USRP1 and the PC.
- **Oscillator** for the FPGA and other digital logic.

- **Personal Computer (PC)**, which provides the digital signal processing power for processing the demodulated data signals. The personal computer is part of the USRP1 hardware, in that only limited amounts of signal processing are possible in the FPGA without it. The PC offers a flexible and powerful software development platform, but would be replaced by an embedded microprocessor, like the one we described in the previous sections, in a real WSN node.

We believe that this architecture is representative of a practical next generation of wireless hardware (and software) based on current levels of electronic integration.

6.1.2 USRP1 Software Architecture

A GNU Radio software base for the USRP1 has been created over the last ten years. Our version is based on an IEEE 802.15.4 implementation (started by UCLA [**56**] and further modified by researchers at CMU [**57**]). The CMU firmware load for the FPGA provides a time-stamping capability, which is useful for our experiments. Our USRP1 algorithm uses a filtered version of the decimated and time-stamped symbol-rate data (in I/Q format) from the FPGA. We have extended the CMU implementation C++ code to log the data samples into a file. Our USRP1 client task runs whenever data is available and ready for processing from the FPGA. To identify the specific demodulated data samples at the beginning of the RF signal, we buffer the data and timestamp information as it arrives, adding a sequence number tag to each data sample and we proceed to demodulate the stream of tagged data samples. The architecture of our software is shown in Figure 15.

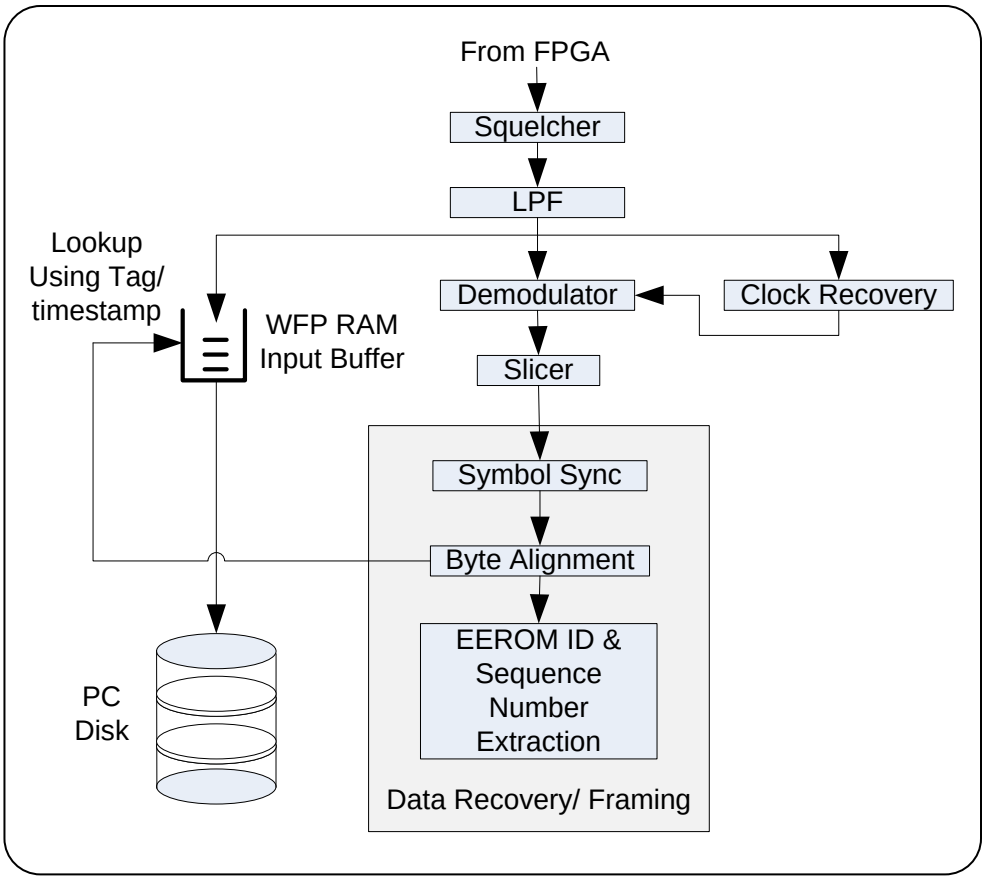


Figure 15: USRP1 Software Architecture

In the order that the data is processed, the functional blocks in the software receiving path are:

- **Squelcher**: The squelcher examines all input data, throwing away noise samples (i.e. ones not associated with messages). The simple magnitude-based method used in the existing CMU code is not adequate for long-range data transmissions in a DSSS wireless receiver. At longer ranges, the nature of the DSSS system requires a framing algorithm that uses knowledge of the IEEE 802.15.4 modulation scheme to distinguish signals from noise, so we have replaced the CMU implementation with one of our own. For consistency, we use the same phase-based framing algorithm for all of the tested transmission distances for the

USRP1-based WFP algorithm (see the 'Data Recovery and Framing' block). In our implementation, the squelcher disables reception of samples periodically (e.g. with a duty cycle of 20s idle time each minute) to allow signal processing of the previous set of samples to complete.

- **Low-Pass Channel Filter (LPF)**: This filter takes the squelcher output and performs digital filtering to extract only the baseband information associated with a single IEEE 802.15.4 channel from the samples received from the FPGA. These channels are spaced 5MHz apart in the 2.4GHz version of the specification. An identifying tag and timestamp is added to the filtered OQPSK data samples (which are still in IQ format) and they are buffered in PC RAM ready for use by the WFP algorithm after the message has been processed further.
- **Demodulator**: The Demodulator determines the phase difference between the successive samples out of the Low-Pass Channel filter, calculating the arc tangent of the product of the current sample with the complex conjugate of the preceding sample. The identifying tag and timestamp added in the earlier filtering stage is propagated through the block without modification.
- **Clock Recovery:** Because an IEEE 802.15.4 system is not necessarily synchronous, the transmitter clock and receiver clock alignment will change over time (see Appendix B). This block recovers the clock from the incoming signal and adjusts the receiver sampling clock to align with the data that is being received. This block is connected in parallel with the demodulator block to the output of the channel filter and adjusts the timing of the receiver in a feedback loop using an M/M (Mueller and Müller) algorithm [**48**].

- **Slicer**: The Slicer converts the variable positive and negative phase difference outputs of the demodulator into digital 1's and 0's, respectively.
- **Data Recovery and Framing**: This block extracts the higher-level data from the stream of digital bits out of the slicer and the corresponding tag and timestamp information. This is done as follows:
    - IEEE 802.15.4 Symbol Synchronization: A correlator searches for IEEE 802.15.4 preamble symbols, each of which is a specific 32-chip pattern of phase changes, corresponding to a 4-bit '0' data nibble (see Appendix A).
    - Byte alignment: A different correlator searches for the specific IEEE 802.15.4 SFD byte pattern to determine the byte boundaries for the recovered symbol nibbles. Our WFP training and classification algorithms uses the first few preamble symbol samples and uses the tag/timestamp information on samples after the IEEE 802.15.4 Symbol Synchronization stage to determine the corresponding samples in the RAM buffer to be copied onto the PC disk.
    - Electrically Erasable Read Only Memory Identifier (EEROM) ID and Sequence Number Byte Extraction: More of the data portion of the message is decoded for experimental reasons (i.e. to identify the specific sources of messages using the transmitted EEROM and Sequence Number information). The duplicated set of 8 EEROM ID bytes and the duplicated pair of sequence number bytes are extracted and both of their correct replications are verified. This information is appended to the copied sample information on the PC disk.

6.2 WFP Algorithm Fundamentals

Our WFP algorithm operates in parallel with the Clock Recovery loop. We also require no special changes to the standard parameter settings for the loop. Receiver clock adjustments are being made at the same time as the preamble of a message is being received. However, our WFP algorithm uses the samples in the first four of the eight preamble symbols for each message, after processing by the Low-Pass Channel Filter. Therefore, the input data for our WFP algorithm is being sampled by a received clock that is still in the process of being aligned with the transmitted data. We devise a simple feed-forward method that does not interfere with the dynamic behaviour of the Clock Recovery loop, but estimates the average phase offset between the transmitter and receiver clock. The WFP algorithm can use this estimated phase offset value in different ways. This section explains the method for phase estimation and other concepts required to understand the different variants and operation of our WFP training and classification algorithms.

6.2.1 'Phase Reversal' Chip Positions and the 'Reversal Mean'

Our WFP algorithm operates on the sampled OQPSK baseband data out of the Low-Pass Channel Filter. Symbols are transmitted at a rate of 2 million chips per second. By using samples collected at a higher rate of 4MHz, all of the transmitted symbols will be observed.

A symbol alignment procedure is required to determine the start of each message transmission, which operates by framing on the predefined and fixed preamble pattern that precedes each message as defined in the IEEE 802.15.4 standard (see Appendix A). In the absence of noise, consecutive phase shifts of the same polarity will yield a

recovered baseband data signal at the receiver with exactly the same phase changes (scaled by the sampling clock rate). For example, if the receiver rate is twice that of the OQPSK transmitter (i.e. using $\pm\pi/2$ phase changes) consecutive receiver phase changes of the same polarity will be observed as $\pm\pi/4$ phase changes.

IEEE 802.15.4 OQPSK symbols are made up of 32 pre-defined patterns of phase shifts, called chips. At chip positions where the phase change polarity reverses, the apparent sampled phase change can be smaller and this affects our WFP calculations. The amount of the reduction in phase change depends on the degree of alignment of the receive clock with the transmit clock as shown in Figure 16.

The figure shows five data bits (11001), encoded as $\pm\pi/2$ OQPSK phase shifts by the transmitter and then later sampled with a receiver clock that is exactly twice this shift rate. The receiver sampling clock edges are shown in dotted green and occur at exactly twice the rate of the RF transmitter's symbol/chipping clock edges, which are shown in dotted black. A data bit value of '1' is encoded as a positive $\pi/2$ phase shift and a data bit value of '0' is encoded as a negative $\pi/2$ phase shift.

Two different cases of transmitter and receiver clock alignment are shown. The upper clock alignment case shows a slightly diminished phase change because of a relatively small positive phase offset between the transmitter and receiver. The lower clock alignment case shows the worst-case misalignment corresponding to a relative phase offset of $\pm T/4$ in the receiver and transmitter clock phases. This results in a sampled phase change of zero at the 'reversal' locations. If the transmitting and receiving clocks are perfectly aligned, the interpolated sampled received data (shown in dotted red in the

figure) and the transmitted baseband data (shown in solid black in the figure) would be identical at these locations.

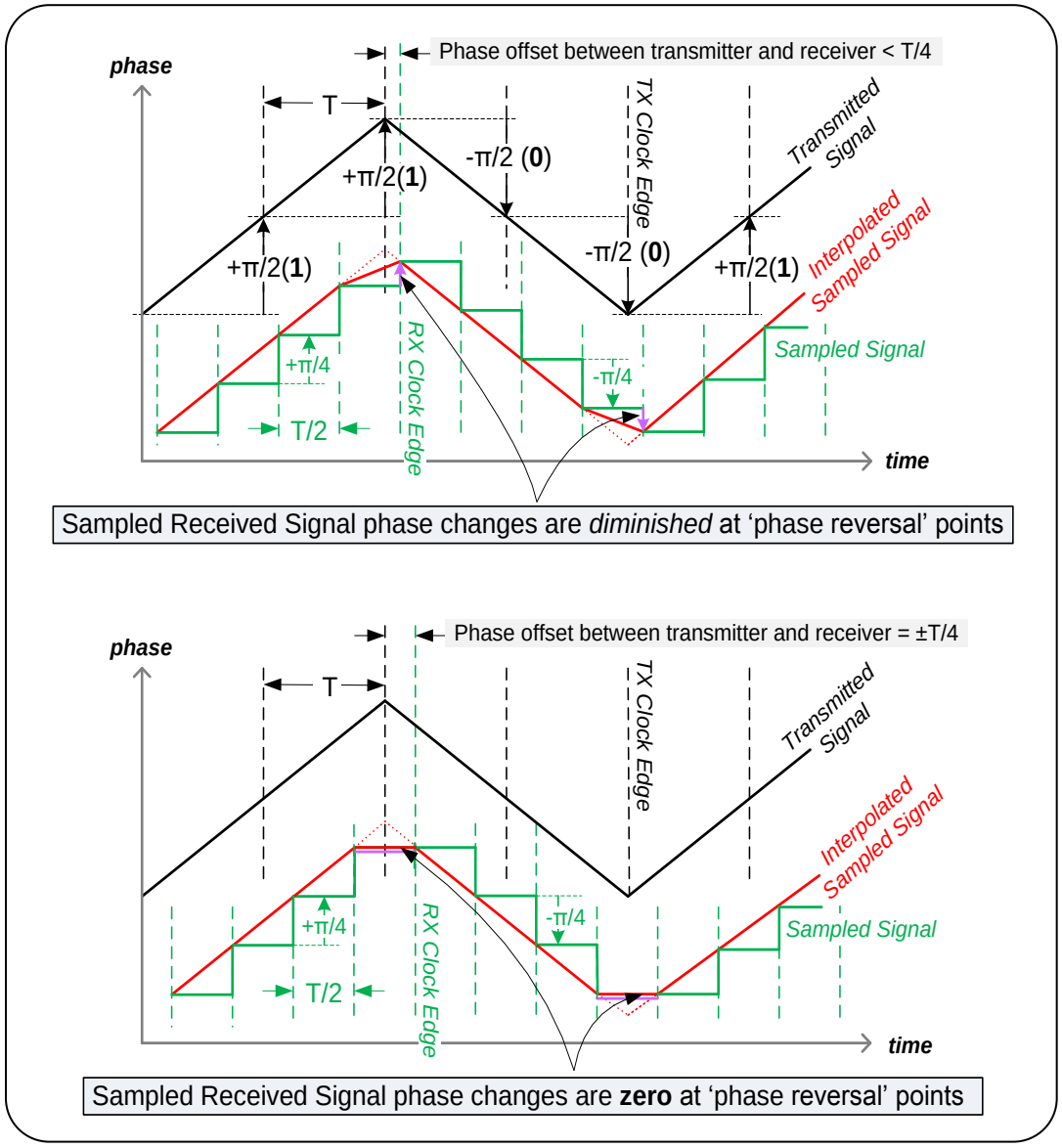


Figure 16: Sampled OQPSK Waveform

We term the portions of the preamble sequence, where the consecutive phase change polarity reverses, the 'phase reversal' chip positions. The 'phase reversal' chip positions can be used to give an estimate of the Transmitter/Receiver (Tx/Rx) phase offset. As shown in Figure 17, we can use the average value at the 'phase reversal' chip positions,

which we term the 'reversal mean' ($\Delta\phi$), to estimate the offset (ΔT) between the transmitting and receiving clock phases.

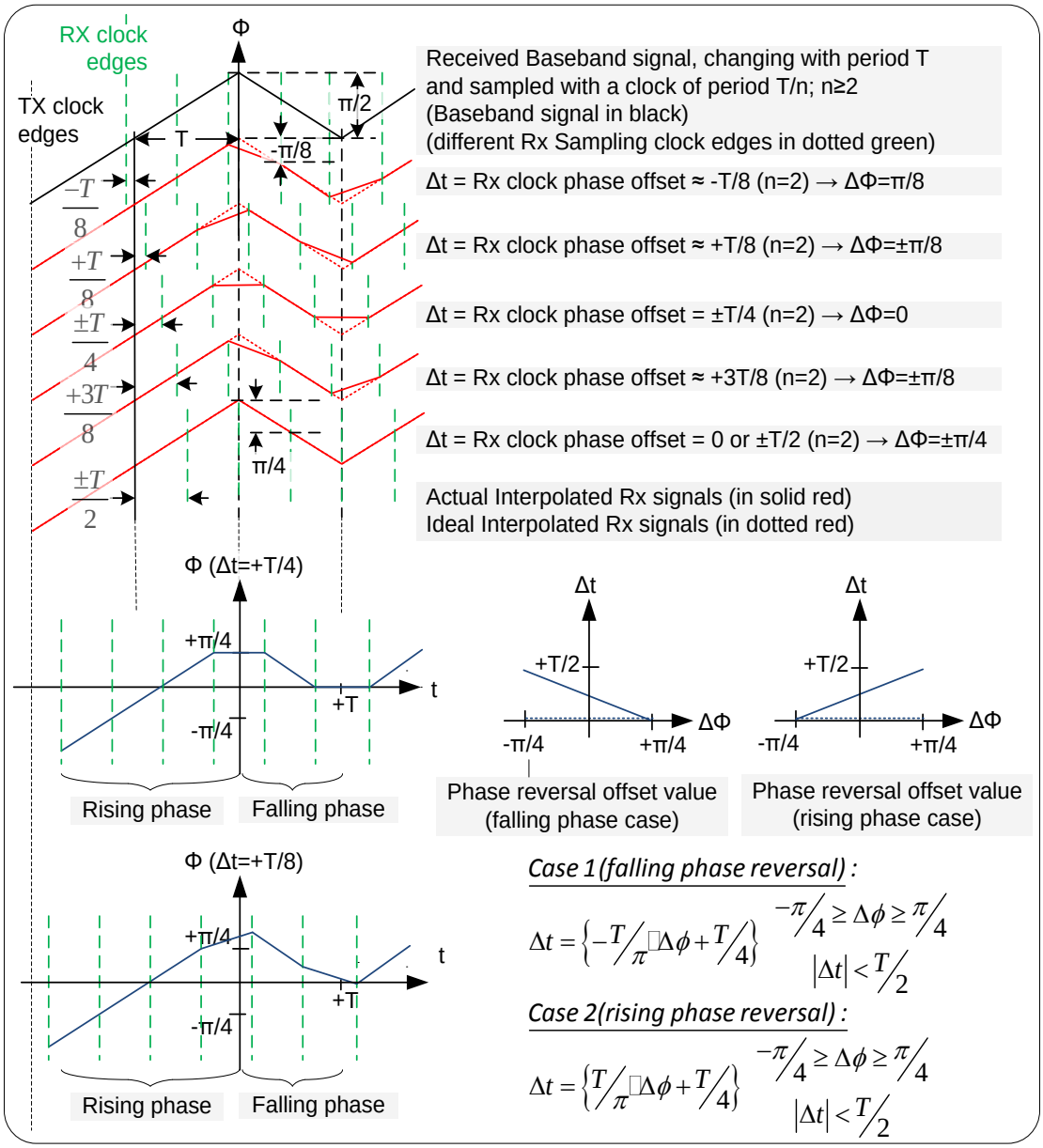


Figure 17: Tx/Rx Clock Phase Offset Estimation in Sampled OQPSK

6.2.2 'Internal' Chip Positions

In the absence of other phase noise sources, consecutive positive or negative OQPSK phase shifts, when sampled non-synchronously at twice the symbol rate, have an average

phase shift of $+\pi/4$ or $-\pi/4$, respectively. We use the term 'internal' chip positions to describe the samples at the chip positions in a sequence of consecutive phase changes of the same polarity. This does not include the samples at the edge of the sequence, where the polarity reverses.

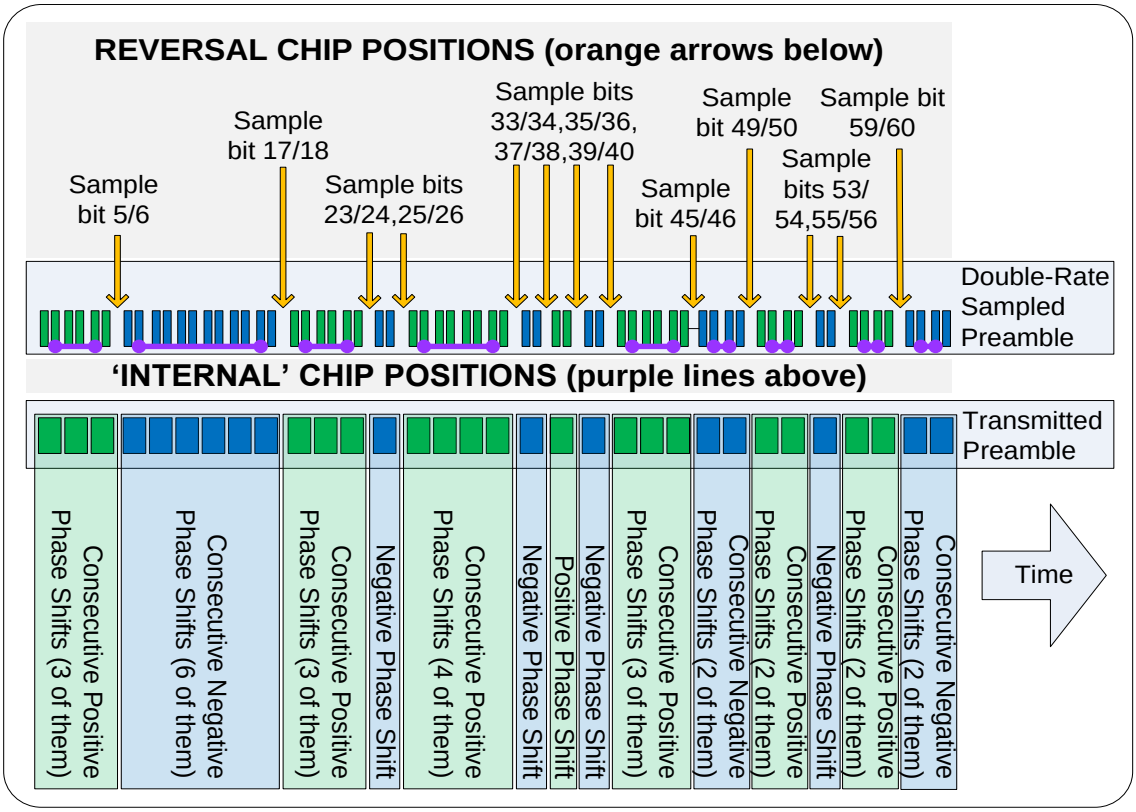


Figure 18: Reversal Chip Positions in First IEEE 802.15.4 Preamble 'Zero' Symbol

The IEEE 802.15.4 preamble zero symbol pattern and the locations of the 'phase reversal' and 'internal' chip positions are shown in detail in Figure 18 and Appendix A.

The 'internal' chip positions are indicated with connecting purple lines at their base in the figure. 'Internal' chip positions are used by our WFP algorithms because they are less affected by misalignment of the receiving and transmitting clocks. To identify the 'Internal' chip positions, we use the Data Recovery and Framing functional block to synchronize to the IEEE 802.15.4 symbols.

Phase differences are calculated by subtracting the phase values of successive samples from each other ($\Delta\phi = \phi_{current} - \phi_{previous}$ values). The closest matching symbol pattern corresponding to the sequence given in Appendix A is determined using the following metric:

$$Maximum\left[\frac{(1+number\_of\_chips\_with\_matching\_polarity)}{(1+number\_of\_chips\_with\_mismatching\_polarity)}\right]$$

This metric behaves well when the number of matching bits or mismatching bits is zero. Rather than merely selecting the longest string with matching bits, it also favours longer strings with only a few mismatches, trading the two off in a reasonable way. Even when there are no 'perfect' alignments possible for a given contiguous set of sampled RF values, WFPs can still be generated.

6.2.3 Distribution of Alignment Errors by Chip Position

Figure 19 shows measured phase difference values for 100 IEEE 802.15.4-compliant preamble sequences[3], transmitted from an RF source (node 'B'). The phase differences in the first 36 (or 37) chip positions for each distinct message from the node are shown as a column of coloured rectangles. The message columns are ordered by reversal mean along the X axis. The phase difference value, $\Delta\phi$, is shown on the Y axis as a proportionally-coloured rectangle for each chip position, with the earliest (chip position 0) at the bottom of the figure.

Blue rectangles indicate negative phase difference values (with a darker blue/purple indicating a more negative phase shift than expected). Green rectangles indicate positive phase difference values (with a yellower green indicating a phase difference that is more

[3]From left-to-right, the 'double-rate sampled' preamble pattern shown in Figure 18 and the rising vertical colour sequence in Figure 19 are identical.

positive than expected). Red/orange values indicate the wrong received polarity relative to the expected preamble symbol, given the best alignment, as selected by our metric.

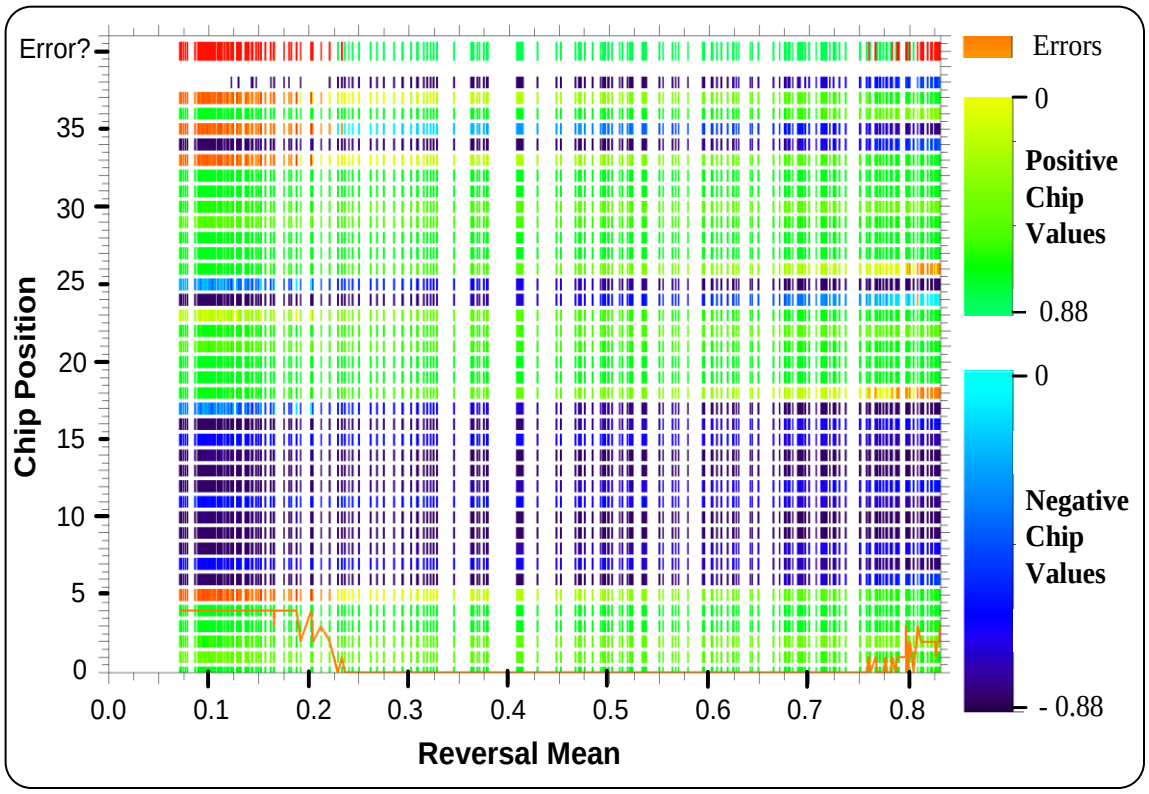


Figure 19: Phase Differences by Chip Position vs. Reversal Mean for Node 'B'

We define the 'reversal mean' as the average of the 'phase reversal' chip values, adjusted by sign. Messages in the figure are ordered horizontally in order of their reversal mean, The large gaps in the figure indicate that the distribution of those phase offsets is not completely uniform for the 100 received messages. Alignment errors tend to occur in the 'phase reversal' chip positions, where the phase margin is lowest.

The solid continuous orange line at the bottom of the figure gives the total number of alignment errors for each message. In the figure, there are a maximum of 4 alignment errors when the receiver's and transmitter's clocks are at their most misaligned (i.e. when

the 'reversal mean' is near zero). Errors also occur at the other alignment extreme with 'reversal mean values near $\pi/4$.

The expected phase shift value of $\pm\pi/4$ is represented by the average green or average blue colours in the rows corresponding to the 'internal' chip positions. Colour variation can be seen, indicating that the specific phase shift values are not the same at each chip position. The phase shift variation varies as a function of the reversal mean. The red or green rectangles at the very top of the figure (at a 'chip position' of about 40) indicate whether a particular message has alignment errors or not. A green colour indicates that there are no alignment errors and a red colour indicates the presence of one or more alignment errors.

Using these indicators, perfect alignments can be seen in the figure for reversal mean values between 0.24 and 0.76. The exact transition values vary for different RF sources. However, for the RF sources that we have observed, most but not all alignment errors occur for reversal mean values at the extremities of the range. The reductions in phase margin are also visible at the reversal chip positions, which is an expected result from our earlier analysis in this section.

The consistency of phase values observed for different 'internal' chip positions in Figure 19 is much better than for the other chip positions and depends less on the reversal mean values. To improve robustness and to allow classification of samples with a wider range of reversal mean values, we use the 'internal' chip positions in our WFP algorithms.

6.2.4 WFP Algorithm Handling of Symbol Alignment Errors

Our WFP training and classification processes must be resilient to symbol alignment errors that result from the receiver sampling clock phase offset with the transmitter's

clock. In sets of error-free samples with perfect alignments, there is no resulting chip position ambiguity since only one perfect alignment can exist without any phase mismatches using the IEEE 802.15.4 preamble pattern. Therefore, we examine the effects of ignoring samples with imperfect alignments in both the training and classification processes and the results are presented in Section 7.2.

The disadvantage of rejecting samples with alignment errors is the large potential number of rejected samples. For example, about 50% of the samples received are rejected with imperfect alignments for the set of messages received from node 'B' in the trial results shown in Figure 19 (e.g. for reversal means less than 0.25 and greater than 0.75). This percentage varies with the transmission conditions and the RF source and also depends on the relative phase offsets of the transmitter and receiver. For our limited set of five WSN nodes, the number of perfect alignments received varied from 50% through to 85%, depending on the RF source in question.

Messages could be split up into different smaller cryptographically-linked sub-messages, to increase the probability that at least one sub-message with perfect alignment exists. Power is wasted with this approach, since each sub-message will have a header that would not be required otherwise. Alternatively, trusted neighbour nodes can also work together to classify a node. If authentication of all messages at every node is a strict requirement, then re-transmission may be required or the zero-tolerance approach must be relaxed and any resulting reductions in classification accuracy must be accepted.

The ratio of perfect to imperfect samples can be used as an estimate of the degree of phase margin over all possible states of the receiver and transmitter. We assume that the initial relative alignments of the transmitter and receiver clocks are uniformly random

with respect to one another between messages. Based on the specified tolerances of oscillators in an IEEE 802.15.4 system, the relative alignment of the two clocks varies slightly during a message, but much more significantly between messages (Appendix B).

6.2.5 Residual Phase Vectors

A phase residual value is a type of error vector magnitude specific to a PSK system and is defined as the measured phase change of the received samples minus the expected phase change value (see Figure 20). This definition requires that received sampled values are framed with the preamble pattern so that the correct expected values are subtracted from the received data.

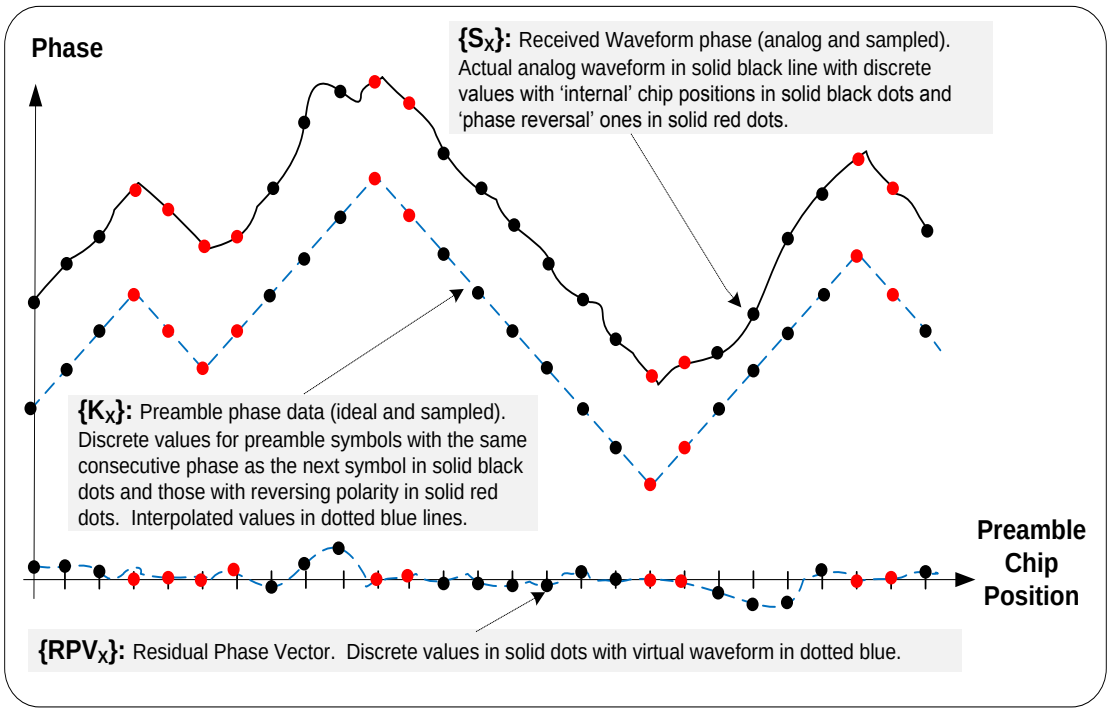


Figure 20: Phase Residual Data Calculation

Once the best alignment has been found, the definition of chip positions is complete and all of the 'phase reversal' and 'internal' points in the received message sample have been identified. As already explained, the residual phase calculation uses only the 'internal'

chip positions, since the phase residuals in the 'phase reversal' chip positions vary too much with different transmitter/receiver clock phase offsets.

We define a residual phase vector (generally termed an ‘error vector’) as the set of the residual phase values for all the 'internal' chip positions in a received message. The residual phase vector represents the phase deviation from a perfect signal and is independent of the relative phase offset of the receiver and transmitter. The WFP training templates for each distinct RF source are constructed using the residual phase vectors.

Mathematically:

residual phase at chip position $i = PR_i \equiv \hat{\phi}_i - \bar{\phi}_i$

$\hat{\phi}_i$ = measured (sampled) phase change at chip position i

$\bar{\phi}_i$ = expected (sampled) phase change at chip position i

residual phase vector = RPV = $\left\{ \begin{matrix} PR_i : \text{i} \in \text{'internal' chip positions} \\ 0 : \quad \text{i} \in \text{'phase reversal' chip positions} \end{matrix} \right\}$

6.3 WFP Training Algorithms

We now propose two algorithms for WFP training using the concepts explained in the previous section.

6.3.1 Fixed Templates (Global Method)

To form the WFP templates for the Global method, we calculate an average of the residual phase vectors for a set of training messages. To normalize comparison between messages of different lengths, the templates are calculated using the same number of 'internal' chip positions, d. All residual phase vectors in the set of training messages are averaged to produce a single average residual phase value in each chip position.

Mathematically:

$$\text{Wireless fingerprint template} = \left\{ \frac{\sum_m PR_i}{m} : \text{ i } \in d \text{ 'internal' chip positions} \right\}$$

$$d = \min(\text{largest valid 'internal' chip position index, with valid data, in } m \text{ messages})$$

The result is a single average template for each of the N known RF sources that has been calculated over all of the m received messages from that source. The same value of m is used for each RF source.

We calculate a Global template which is an average of all of the received samples for a particular RF source. Thus, the template will be at the centre of the total variability observed over all random alignments and over the duration of the training period. If the phase residual values do not vary with the receiver/transmitter phase alignment for the different chip positions, then the Global template is expected to perform well. Furthermore, only one value for each bit position needs to be stored for each known ID. If storage is a primary concern, Principal Component Analysis (PCA) can be used to reduce the dimensions of the stored training template database (see Section 6.3.4 for an explanation of PCA). In this case, all that needs to be stored are the first few PCA Principal Components for each known RF source and also the coefficients required to map the chip positions into Principal Components.

6.3.2 Variable Templates Organized by Reversal Mean (Local Method)

For the Local Method, we divide up the [0, π/4] reversal mean interval into J sub-intervals and attempt to find roughly k entries for each of those sub-intervals, yielding a total of kJ residual phase vectors, which are stored in a training database dictionary, organized in order of their 'reversal means'. The WFP training templates are determined by first calculating the 'reversal mean' of the new sample being classified. The template

in the training template dictionary with the closest 'reversal mean' value is selected for each known RF source and then used in the classification distance calculation. For better noise tolerance, an average of the k closest templates is used, rather than a single message sample.

The time alignment of the residual phase vector template being used for classification should be as close as possible to that of the received message. Higher values of k or more sub-intervals give a finer granularity for 'reversal mean' lookup, but require more storage and computation. After determining the relative clock offset between the transmitter and the receiver using the methods of Section 6.2.2, we select a set of samples that closely resemble the sample to be classified using that offset (i.e. are in the same 'zone' of phase marginality). These templates can be considered to be 'local' neighbours of the sample being classified. Note that multiple neighbours are required to gain some time-averaging advantage and also to avoid being overly sensitive to outliers in the training data set.

By increasing the training group size, k, we are less sensitive to outliers and will get better averaging. However, the cost is decreased locality, where samples that are further away (in terms of estimated phase offset) are included in the template. We determine that a neighbourhood size of k=5 messages yielded satisfactory performance in trials consisting of 100 (i.e. J=20) samples, using messages with the majority of the messages having similar reversal means in most of the trials. Adding more neighbours improved performance only marginally. With this method, if not enough messages are received for a particular range of reversal means, the training period must be extended until coverage is adequate.

The main disadvantage of this method is that many more samples may need to be processed than are used, while waiting for samples to fill 'gaps' in the range of 'reversal means' (see Figure 19). The training stage is assumed to be executed less often than the classification stage, so energy expended in the training stage has less of an impact on overall power consumption.

6.3.3 Extending WFP Algorithms with Residual Phase Noise Filtering

The WFP algorithm input can be pre-filtered to remove input samples which are deemed to be too noisy. We have already discussed the effects of sampling on the apparent phase change during the training process. The RF environment is also noisy (especially the 2.4GHz ISM band). Figure 21 shows the combined theoretical effects of RF noise and sampling noise on the digital representation of an analog RF signal. The linear superposition of the two sources of noise makes the overall resulting sampled waveform more noisy.

The noise has a component in both the horizontal and vertical directions. The horizontal noise component motivated the design of the Local training method, which uses the relative phase offsets between the transmitter and receiver as an input. The Global method compensates by averaging over a large enough sample of samples with different phase offsets.

Misalignment of the receiver's sampling clock with the incoming data signal translates to uncertainty on the precise time locations of the measured sample I and Q values, relative to the signal detected at time zero (corresponding to the dashed green lines for the different ϕ_i's in the upper part of the figure). We assume that the RF signal produced by an RF transmitter is either constant or changes very slowly, so that noise can be defined

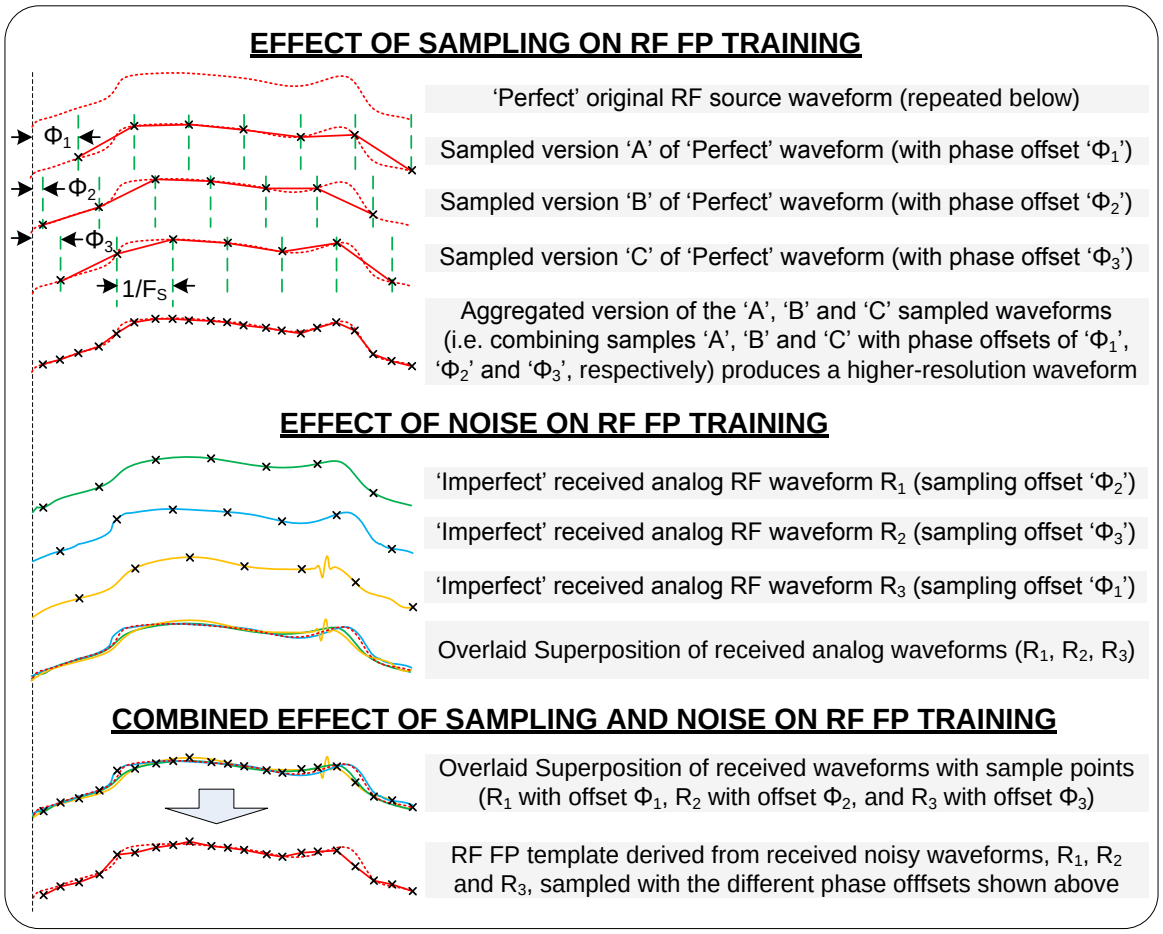


Figure 21: Combined Effects of Noise and Sampling During the Training Phase

as a transient phenomenon. 'Persistent' noise correlated or synchronous with the transmission of messages becomes part of the signal being characterized. Time-averaging the measurements made over a noisy RF channel will filter out the non-deterministic and non-constant noise.

We have implemented the practical noise-filtering approaches of other researchers [**58**], where samples that are too far away from the template mean value are discarded prior to being used for training. We determine a percentage of messages to be discarded, rejecting the corresponding number of the samples with the highest variance. Another approach would be to use a fixed threshold for the maximum variance above which

messages are discarded. This method is dependent on the stability of RF channel noise conditions. Finding a fixed common threshold for different nodes is also difficult, if certain nodes are inherently noisier than others.

6.3.4 Extending WFP Algorithms with Principal Components

Principal Component Analysis (PCA) can be used to reduce the size of the Residual Phase Vector. Using Principal Component Analysis (PCA), we can determine which preamble chip positions are the most significant in the WFP processes. The main advantage with the method is that less data needs to be stored and used for calculations. This is examined in more detail Section 7.5. PCA is also useful as a tool to visualize and compare discrimination performance of different WFP training and classification methods (see Section 7.4).

PCA automatically identifies and uses values with the maximum discriminating power, by transforming a set of random variables, which may or may not be correlated with each other, into a set of uncorrelated orthogonal random variables, called Principal Components (PrC). Each of these Principal Components can be viewed as an axis onto which the measured data is projected linearly. Each Principal Component is defined to explain as much variability as possible, with subsequent components used to explain the remaining variability, while also being orthogonal to the previous components.

The first Principal Component projection axis gives the maximum possible variance in the projected data values. To determine the second Principal Component, this variability is removed by subtraction. Using the resulting data, the second Principal Component is defined in a similar greedy fashion to the way that the first was defined. This procedure of identifying and removing the Principal Component dimension with the highest

variability is repeated a number of times up to a maximum number of Principal Components.

The maximum number of Principal Components will be no greater than the dimension of the original dataset (which is equal to the number of chip coefficients). However, in most cases, the variability in the original dataset can be captured by fewer Principal Components than this. This reduction in the overall dimension of the dataset means that only the first few Principal Components need to be calculated to characterize most of the observed variability in the data, resulting in computational savings. The exact amount of this reduction depends on the amount of noise in the system and can be expected to vary. The Principal Components have no direct correspondence with the physical world, but they can be expressed as a linear combination of the original variables. A loading matrix is defined that transforms the set of physically meaningful random variables (i.e. phase residual values by chip position) to the new set of Principal Components.

We extend our two WFP training algorithm variants using PCA as follows:

1. The specific training algorithm being analyzed is run to establish phase residual values in each preamble chip position for each of the distinct IDs.
2. The training data is centered by subtracting the mean value, taken over all symbols and standardized by scaling the values to have a standard deviation of one.
3. A principal component analysis is performed (using an 'R' [**59**] routine, called *prcomp*) and the loading matrix is determined, to derive the Principal Components from the phase residual values at each chip position.

To classify using Principal Components, the phase residual values are all converted into

their PCA representation using the loading matrix and the calculations then proceed in the same way as before. For the Global training algorithm, the template for each known RF source is converted into PrC-space. For the Local training method, the stored samples (still based on reversal mean) are each converted into PrC-space and then used in the same way as before (but with reduced dimensionality).

Messages to be classified are converted into their PCA representation using the loading matrix and a distance metric is calculated, but the distance calculation is performed over all of the Principal Components, rather than over each of the chip positions. The number of chip positions being used must still be standardized for template and message samples so that a complete and accurate PCA representation is produced.

6.4 WFP Classification Algorithm- Distance Calculation

Once the template database has been created using the training algorithms described in Section 6.3, new samples can be classified using the WFP classification algorithm. During this phase, the most likely RF source for a new sample is estimated, by calculating the classification distance to the template for each known RF source in the database. This distance metric is a mean squared error calculation between the residual phase vector of the sample that is to be classified and the residual phase vector for each training database template.

The distance metric is defined as:

$$\textbf{Classifier Distance} \equiv \underset{\text{X chip values}}{\text{Average}}\left(\sqrt{\sum_{X=1}^{\text{normalized number of chips}} \left(RPV_X - TRPV_X \right)^2} \right),$$

where *RPV* is the Residual Phase Vector of the received sample and *TRPV* is the Template Residual Phase Vector.

This calculation is performed over the normalized number of X 'internal' chip positions, which should be the same for all known RF sources, based on the proposed training algorithms. The basic classification method selects the template with the lowest classification distance to the newly-arrived sample as the most likely RF source.

To implement the Local training algorithm, we modify the classification algorithm as follows:

- The 'reversal mean' of a newly-arrived sample is determined
- The training sample with the closest 'reversal mean' is then found for each of the N known IDs and the corresponding template information from the template database is used in the distance calculation above. We modify this basic Local method to use the average of the k 'closest' training samples for the template of each known ID. This is shown in Figure 22, where we use k=3. The samples in the dotted red boxes are averaged to produce the template.

We assume that there are a total of $k \bullet J$ samples stored in the training database. J is the selected bin granularity value and represents the desired number of divisions of the total range for the 'reversal mean' indices that are being used for training sample lookup. Higher values of J increase the resolution for the Local training algorithm, reducing the average distance to the 'closest' k training samples. This increased resolution comes at the expense of increased training sample storage costs.

However, in a non-synchronous system, it is not possible to control the initial phase offset between the Receiver's and Transmitter's clocks. Therefore, the 'reversal mean' values might not be uniformly distributed across the entire range for certain RF sources

(e.g. node 'D' in the figure). To fill in these potential gaps, the training period can be extended until there are at least *k* samples in each of the *J* bins.

The noise filtering techniques used in the training phase may not be applicable in the classification phase. We might not want to refuse to classify a fixed percentage of noisy signals, as we did in the training process. Rather, we can *attempt* to classify each newly-arrived signal using our training template database and the classification distance that we calculate for each known RF source.

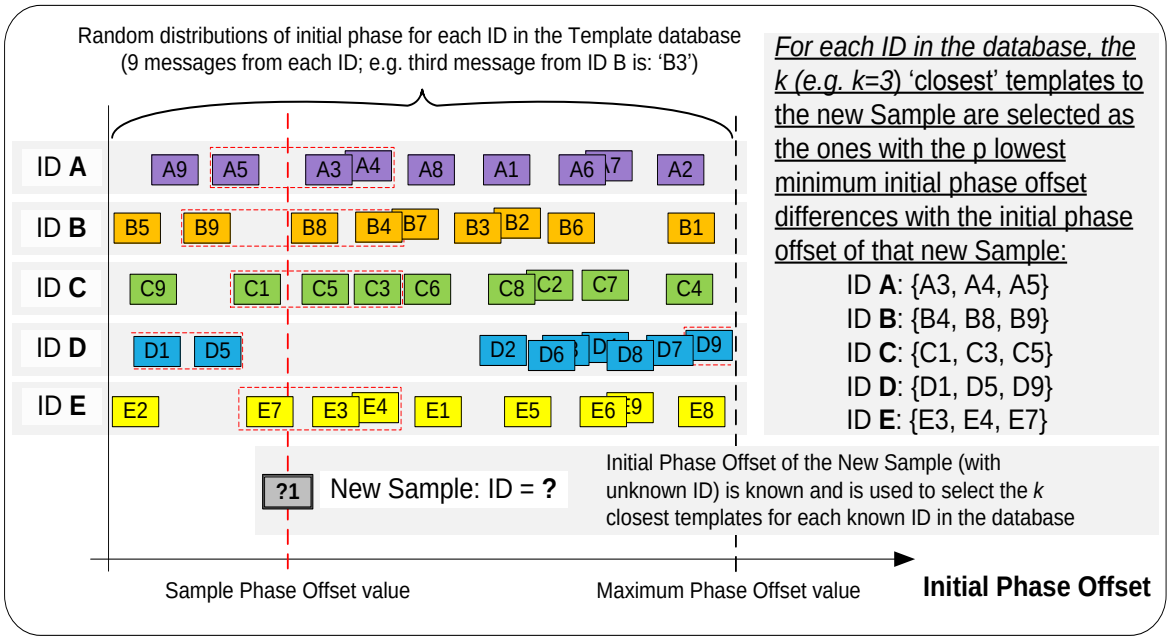


Figure 22: Method 1: Local Training Method Without Binning

If the two smallest classification distances are too close to each other (i.e. smaller than a *distinguishability* threshold), the classification is determined to be 'too close to call' and we fail in our attempt. If the smallest distance is too large (i.e. exceeds a *goodness-of-fit* threshold), the classification decision is that this ID is a new one that does not exist in the training template database yet.

Figure 23 shows the classification distances between all possible combinations (without comparisons of messages with themselves) of 50 messages received from each of a set of 5 different nodes. Black points on the scatter plot correspond to points from the same RF source, while coloured/shaded points correspond to the distance between samples from different RF sources. The classification distances between each message are plotted against the difference of the reversal means of those messages (i.e. samples with the same reversal mean have a reversal mean difference of zero).

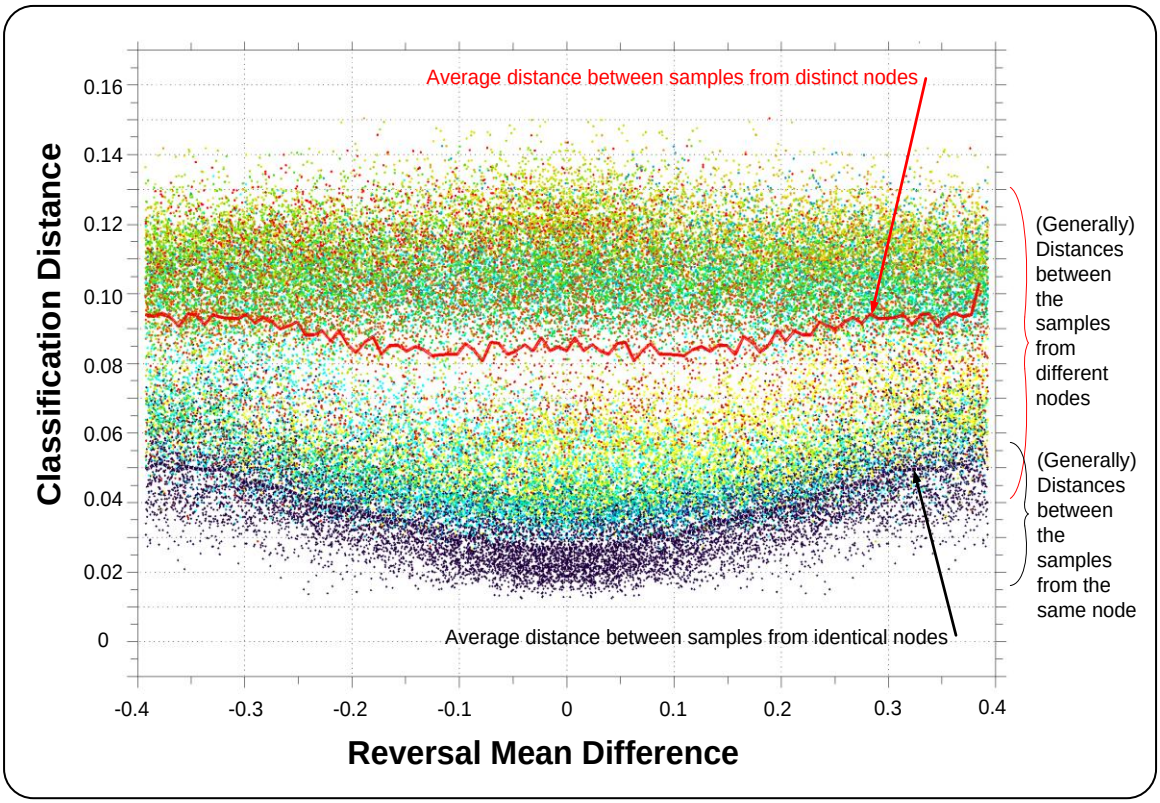


Figure 23: Classification Distances for 5 WSN Nodes

The dotted black line is the average classification distance for transmissions from the same RF source and the solid red line is the average of the distances between distinct RF sources. Samples with similar phase offset ($\Delta\phi$) values tend to have 'nearer' classification distances. This is more pronounced for samples transmitted from the same RF source

plotted against the difference of their $\Delta\phi$ values. This result provided the initial motivation for the Local training method based on using samples with similar reversal means in the training set.

6.5 Implementation Issues

There is only one Gnu Radio implementation of the IEEE 802.15.4 standard available for the USRP1. In its final form, our WFP algorithm disabled most of the higher-layer IEEE 802.15.4 functions in this software. We require only the IEEE 802.15.4 data extraction and the ability to link the recovered preamble data symbols with the original baseband samples that produced them (at the channel filter output). We now review other significant changes that we encountered when implementing and testing our WFP algorithms on the USRP1 platform.

6.5.1 Squelching Threshold Decision

When nodes are separated by large distances, the changes in the average power for received IEEE 802.15.4 signals are small, although still detectable. For our experiments, nuisance triggering (from neighbouring IEEE 802.11g/n sources as well as microwave ovens) proved bad enough that our initial simple amplitude-based triggering methods had to be migrated over to full-symbol decoding frame detection methods, requiring the radio to discard samples and be 'offline' for 20 seconds out of each minute while the PC completes the processing backlog.

Discarding samples in this way is acceptable, since the IEEE 802.15.4 protocol specifies contention-free/'beacon mode' mechanisms to ensure that nodes can operate in power-saving sleep modes most of the time, only waking up periodically at synchronized times to exchange information with each other. If not operating in 'beacon mode', then we

would require the same type of hardware buffers for WFP input samples that were already identified in Section 4.3, which are similar to the data buffers already provided on WSN node devices.

6.5.2 CPU/ RF Front End Bandwidth

Higher bandwidths for incoming demodulated data samples are almost certain to improve the quality of WFP measurements and also reduce the effects of phase misalignments between the RF source's transmission clock and the receiver's sampling clock. However, our central assumption is that WFP processing at near-RF bandwidths is not feasible on real nodes. This limits the increase in complexity and speed that we are prepared to tolerate on this interface.

An alternative to increasing the rate is to control the phase offset between the receiver's sampling clock and the transmitter's clock in a more deterministic fashion. A clock recovery function already exists in the receiver (on the WSN node or on the USRP1 platform) to lock the receiver to the transmitter's symbol frequency. Rather than adapting to the initial alignment of a particular message and discarding misaligned samples, as in our USRP1-based method, the processor could control the offset, varying it in a deterministic way to build up a composite WFP template with more resolution. We did not perform any experiments that included this proposal, given our objective of integrating WFPs unobtrusively into the normal operation of a WSN node in a network.

6.5.3 Noise Filtering

Environments have different levels of noise at different times, making it difficult to devise a general method for noise filtering. Our approach of discarding samples that are

too far from the WFP mean works well in different noise conditions, but adapts slowly to changes in those noise conditions.

The method described in the previous section, for varying the symbol clock phase offset, also allows a receiver to move the WFP detection location into a 'quieter' zone. This can be accomplished with a feedback loop that compares classification performance with the controlled phase offset. A special training period is required where the receiver tunes itself, moving the phase offset until WFP classification reaches an optimal value. Alternatively, a set of equally spaced candidate offsets can be used and multiple WFPs generated at each offset for each detected neighbour. In this work, we do not interfere with the operation of the clock recovery loop.

6.6 Summary

We reviewed the hardware and software architectures of the USRP1. The USRP1 is a flexible experimental wireless platform that allows full access to demodulated baseband data samples. However, the USRP1 is still representative of WSN hardware.

We presented the concepts required to understand the two variants and different parameters of our WFP algorithm. We presented the Global and Local variants and explained how to use Principal Components instead of chip coefficients to improve the efficiency of the implementation. In the next chapter, we present the results of our experiments to characterize the classification accuracy of these different variants.

7 Chapter: USRP1 Experimental Results

The performance results collected using the USRP1 are presented and analyzed in this chapter for two different implemented WFP algorithms. The analysis and design criteria from the previous sections and the measured performance are used in Subsection 7.5 to select the best classification and training algorithm, along with a complexity analysis (for both processing and memory requirements) to provide insight into the relative implementation difficulty of the different algorithms. Further experiments on the selected WFP algorithm are then presented to determine the performance: using different receivers, using different RF channels. The stability of the classification performance over time is also measured.

We adopt a common statistical method that has also been used by other researchers in the wireless fingerprinting area. Measured data samples are randomly sub-divided into mutually disjoint subsets of training and classification samples and then the discrimination performance is measured. This avoids favouring schemes that 'tune' to specific data sets.

The training process uses measurements which do not include the samples that are then going to be classified. Therefore, any rejection of samples as outliers is explicit in the definition of the training or classification algorithm. By varying the widths and start time of the time windows used for the selection of training and classification samples, the stability of the training and classification algorithms can be assessed over time.

7.1 USRP1 Experiment Design

The RF measurements are made using an Ettus Research USRP1 SDR platform [**43**] with an RFX2400 [**55**] daughter board and modified GNU Radio software [**53**] that runs on a

desktop PC running version 10 of the Ubuntu Linux software operating system (see Section 6.1.2). The results stored on the PC disk are post-processed using the different algorithms for training and/or classification. Graphs were produced using the Dislin [**60**] software package.

We accept the effects of small changes in the receiver and transmitter position variations on the WFP measurement process, since this is typical for a standard indoor residential environment. However, we try to keep the RF environment as stable as possible to simplify our analysis. RF measurements are made in a different indoor environment from the one used for the earlier WSN experiments. The house used for the USRP1 experiments has a cinder block construction and Gypsum walls. Short, medium and long transmission distances of 1.5m, 4m and 10m, respectively, are used for our testing.

In typical indoor conditions, long-distance testing implies the presence of intervening walls and floors, so we perform WFP classification using similar conditions. While it is inevitable that devices can be placed in slightly different transmission positions and/or subject to movement, the locations and orientations of transmitters and receivers are fixed using marked locations on floors and tables. The relative positions and orientations of the transmitters are kept constant as shown on the left side of Figure 24.

The receivers are located 0.5m away from the exterior wall of a room at the front of the house and are mounted on a wooden platform, standing 0.4m off the floor. The transmitters are mounted on plastic platforms at a distance of 0.9m above the floor and secured in place. The right side of Figure 24 shows the SDRs arranged in ‘Position 1’. If the SDR positions are reversed, the resulting arrangement is called 'Position 2', which is important later in Section 7.6.2.

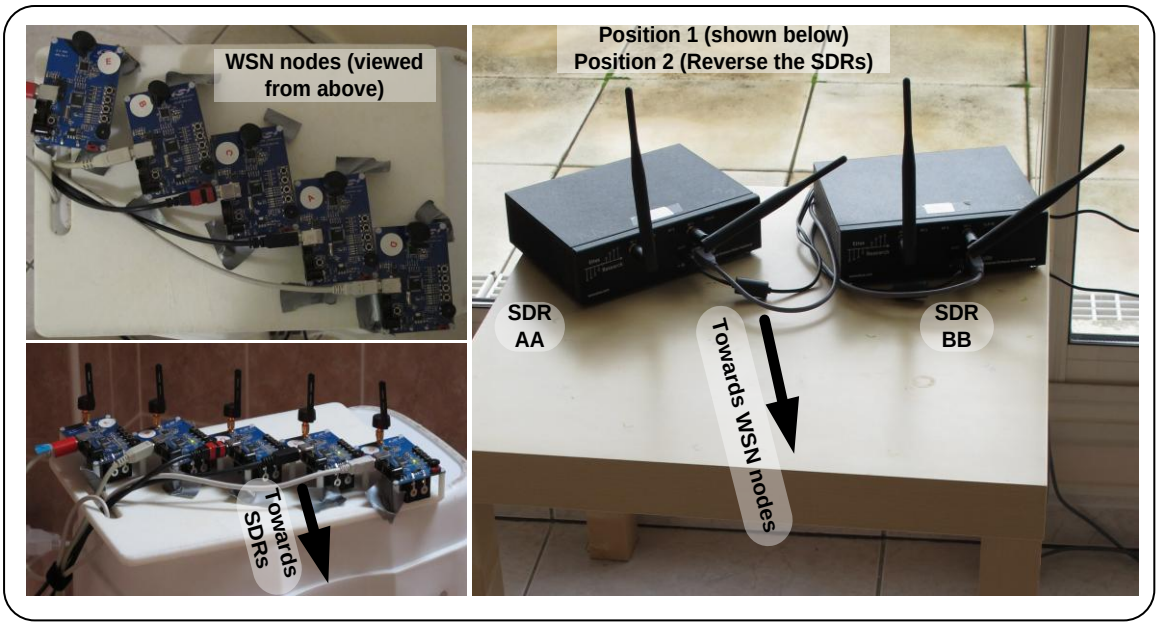


Figure 24: Physical Configuration of WSN Nodes and SDRs During Testing

To reduce the effects of deep fades, the Short and Medium testing uses LoS visibility between the transmitters and receivers. There are no intervening walls or other obstacles and testing takes place in a room with standard residential furnishings and construction. Transmissions are subject to interference from a variety of uncontrolled sources (e.g. IEEE 802.11g/n routers, microwave ovens). For the long-range testing, the five distinct WSN RF sources are located in a separate room with tiled construction at the back side of the house. This results in the transmitters being separated from the receivers by two intervening Gypsum walls, one of which is covered with ceramic tiles.

All of our algorithms discard messages that do not contain enough IEEE 802.15.4 PHysical LaYer (PHY) layer signal content to be considered valid signals. As already discussed, duplicated sequence numbers and duplicated node identification codes are included in the content of each message. Messages which do not have repeated identical versions of these data are discarded.

Each RF source generates a single RF message signal each second and is monitored for 24 hours, yielding a total of over 86,000 signal transmissions from each RF source in the period. With current WSN technology, such a transmission rate is consistent with a standard battery lifetime of a few years. While attempts have been made to maintain the different WSN nodes in identical transmitting positions during each experiment, the RF channel is not guaranteed to be stationary. We believe that these conditions are consistent with the conditions in a deployed WSN network. WFP algorithm stability over time is verified by comparing classification accuracy for five distinct 1-hour intervals spread throughout the 24-hour test period, using a 1-hour training interval at the start of the test.

7.2 Classification Performance

We measure classification performance using our two proposed methods: Global and Local. For both of these methods, we compare performance using samples both without symbol alignment errors and also permitting them (e.g. up to thirty errors in 118 chips). We also compare the performance using the modified extensions of both methods using residual phase noise filtering and using Residual Phase Vectors based on Principal Components instead of chips.

For each algorithm, we randomly select a set of 100 messages from each of two disjoint time intervals. The first time interval contains the training samples, and the subsequent time interval contains the classification samples. This selection process is done for each of 5 different RF WSN node sources (labeled, 'A', 'B', 'C', 'D' and 'E') using the same time intervals. For both algorithms, this selection process is continued in parallel with the filtering process for chip alignment errors. If filtering, we keep track of how many

messages are rejected and discarded during the selection process, until enough messages with acceptable alignment have been obtained.

We use a 1:1 training to classification ratio, giving 100 training samples and 100 classification samples. While we do not present the results here, the classification performance of the Global algorithm was relatively stable to changes for this ratio (e.g. similar results were obtained with a ratio of 1:10). However, the Local algorithm improved noticeably as more training samples were used.

For the classification distance calculations, we use a comparison difference significance threshold that is low enough to mitigate the effect of round-off errors. However, with such a low threshold value, the WFP algorithm rarely refuses to make a decision as being 'too close to call'. For this reason, we record such results as misclassification errors in our statistics and do not analyze them separately. A 'new' node is identified if a sample is too far away from all of the known templates. We did not test the WFP algorithm's ability to detect new nodes, given the relatively small size of our WSN sample set.

If we plot the distances from the classification samples (on the Y axis) with the training samples (on the X axis) as a proportional colour for each of our runs, we see an indicator of the overall classification performance at a glance (Figure 25).

The figure shows data collected at medium range from SDR 'BB' using the Global algorithm. The messages are ordered within each of the five source-specific cells on each axis by their estimated $\Delta\phi$ value. Tones towards the blue/dark end of the spectrum represent closer classification distances and yellow/red colours represent larger distances. A white colour represents a classification distance that is large enough to exceed the maximum shading scale value.

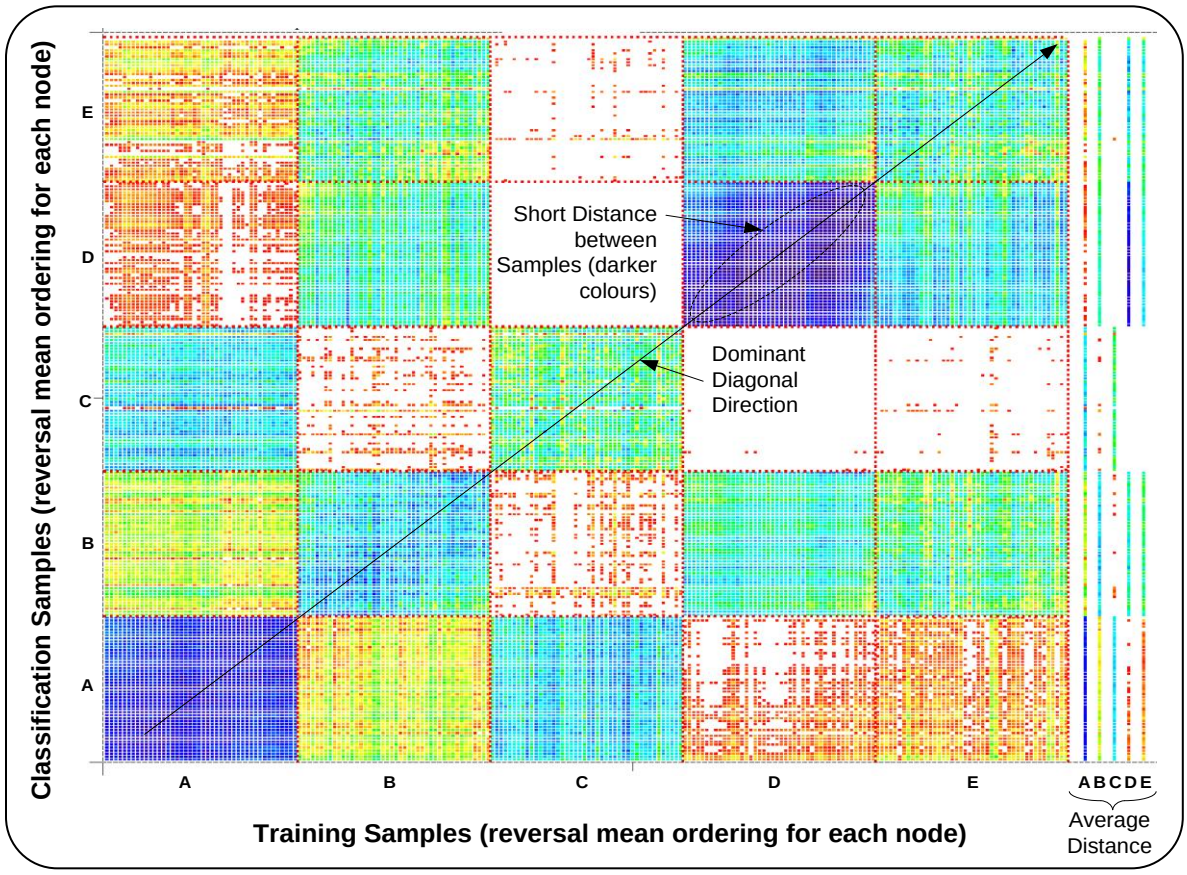


Figure 25: Classification Distances For 5 WSN Nodes (Reversal Mean Ordering)

The classification distance averages from each cell for each classified message (i.e. each cell row) are displayed on the right-hand side of the figure in vertical bands. For the Global algorithm, the best performance will be obtained when these vertical bands are uniformly dark for when training and classification samples come from identical nodes. A 'dominant diagonal' (transposed principal diagonal) of darker colour can be seen for the training and classification samples of the same nodes, indicating that the Local training algorithm might perform better than the Global training algorithm, since it uses the value of $\Delta\phi$. Certain nodes (e.g. nodes 'A' and 'D') have a more pronounced diagonal than others (e.g. node 'C'). This would imply less of a performance advantage with a Local training algorithm for node 'C'. Darker cells off the dominant diagonal indicate nodes

that are more easily confused with each other (e.g. node 'C' might be confused with node 'A' and node ‘E’ might be confused with node 'D').

7.2.1 Experimental Results - Training with Local Templates

Figure 26 and Figure 27 illustrate the classification performance using templates derived using the Local training process using measurements made at a range of 4m by SDR ‘BB’, discarding messages with detected alignment errors. Data for different RF sources are plotted in different colours and marker styles. The classification performance results from 100 trials are shown for each source.

Figure 26 shows the classification performance if alignment errors are tolerated in the classification samples. A ‘best’ preamble chip alignment is determined using the methods already described and the number of discrepancies calculated over all chip positions. We permit up to 30 alignment errors to be tolerated during the classification stage but no alignment errors to be tolerated during the training stage.

With this approach, the Local algorithm performance typically has an average classification accuracy between 43% and 47% for the worst-case node ‘C’. If we change the approach and tolerate no alignment errors during classification, the results improve by 1.4% to 9%, depending on the RF source, as shown in Figure 27 and Table 3. The table shows the average classification accuracy results (and the 95% Confidence Intervals, or CIs) for each RF source using the two Local algorithm variants. Average classification performance improves for all nodes, when samples with alignment errors are excluded. With rejection, classification performance is almost perfect for nodes ‘A’, ‘B’ and ‘D’.

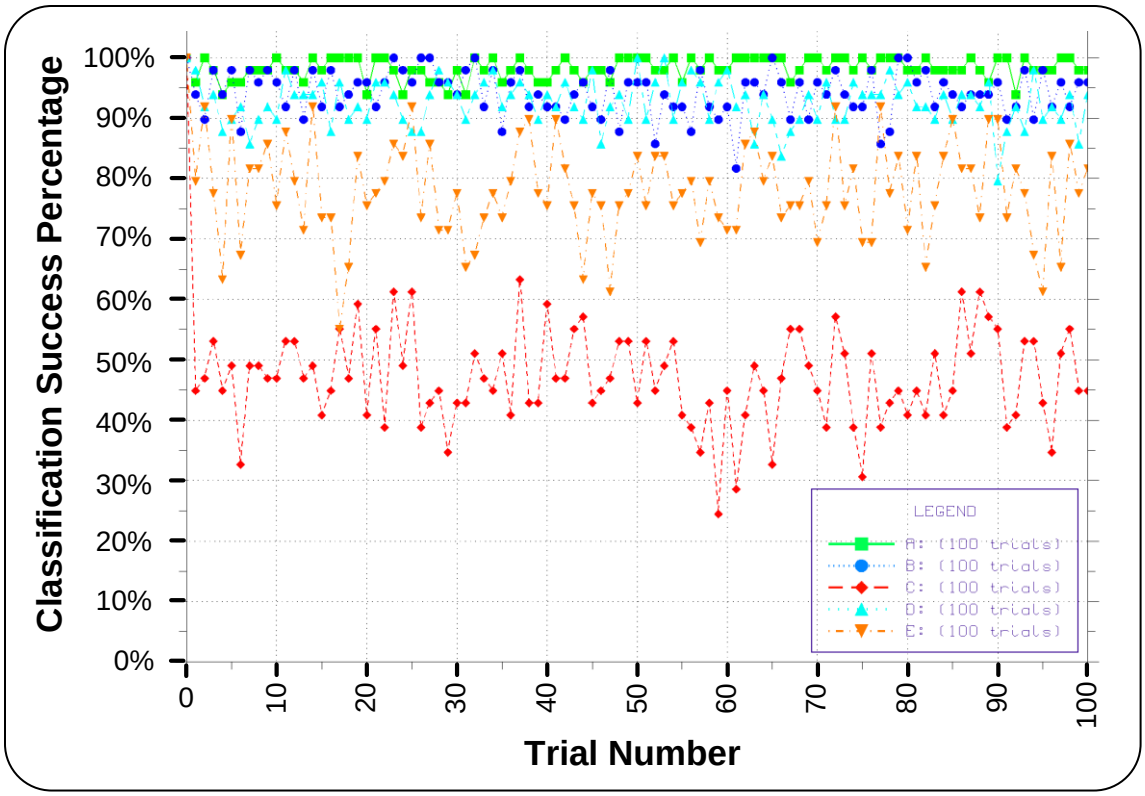


Figure 26: Local Classification - Alignment Errors Tolerated (Medium Range)

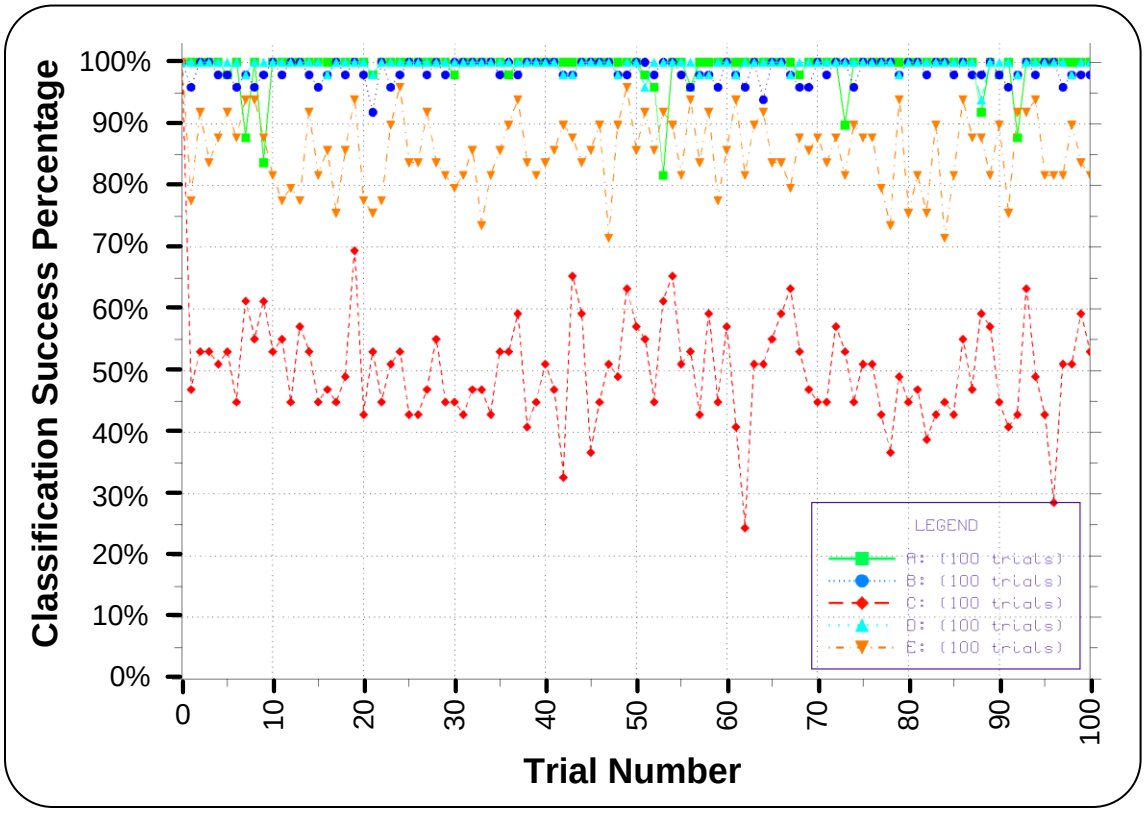


Figure 27: Local Classification – No Alignment Errors Tolerated (Medium Range)

Table 3: Mean Local Training Algorithm Classification Accuracy

Node ID:	***A***	***B***	***C***	***D***	***E***
Local Training Method - Including message samples with alignment errors during classification only	97.65% [97.03%, 98.27%]	93.86% [93.14%, 94.58%]	44.98% [43.28%, 46.68%]	90.96% [89.8%, 92.12%]	77.37% [75.8%, 78.94%]
Local Training Method - No message samples with alignment errors included during classification or training	99% [98.36%, 99.64%]	98.59% [98.26%, 98.92%]	49.69% [48.15%, 51.23%]	99.63% [99.44%, 99.82%]	85.27% [84.09%, 86.45%]

7.2.2 Experimental Results - Training with Global Templates

Figure 28 and Figure 29 illustrate the classification performance using templates that are derived using the Global training process. Figure 28 shows the performance if alignment errors are tolerated during classification. Figure 29 shows the performance if samples with one or more preamble chip alignment errors are rejected during classification. Table 4 shows the average classification accuracy results (along with 95% CIs) for each RF source using the Global algorithm.

Reviewing the tabulated data in Table 4, the Global algorithm average classification performance for nodes 'C' and 'E' improves when samples with alignment errors are rejected at the classification stage. The average classification accuracy performance improves for all RF sources. We now examine the effects of different rejections strategies on the two algorithms in more detail.

Table 4: Mean Global Training Algorithm Classification Accuracy

Node ID:	***A***	***B***	***C***	***D***	***E***
Global Method - Including message samples with alignment errors during classification only	97.23% [96.59%, 97.87%]	97.23% [96.71%, 97.75%]	82.92% [81.73%, 84.11%]	95.19% [94.37%, 96.01%]	87.51% [85.99%, 89.03%]
Global Method - No message samples with alignment errors included during classification or training	98.69% [98.13%, 99.25%]	99.57% [99.38%, 99.76%]	89.47% [88.56%, 90.38%]	99.78% [99.65%, 99.91%]	96.37% [95.87%, 96.87%]

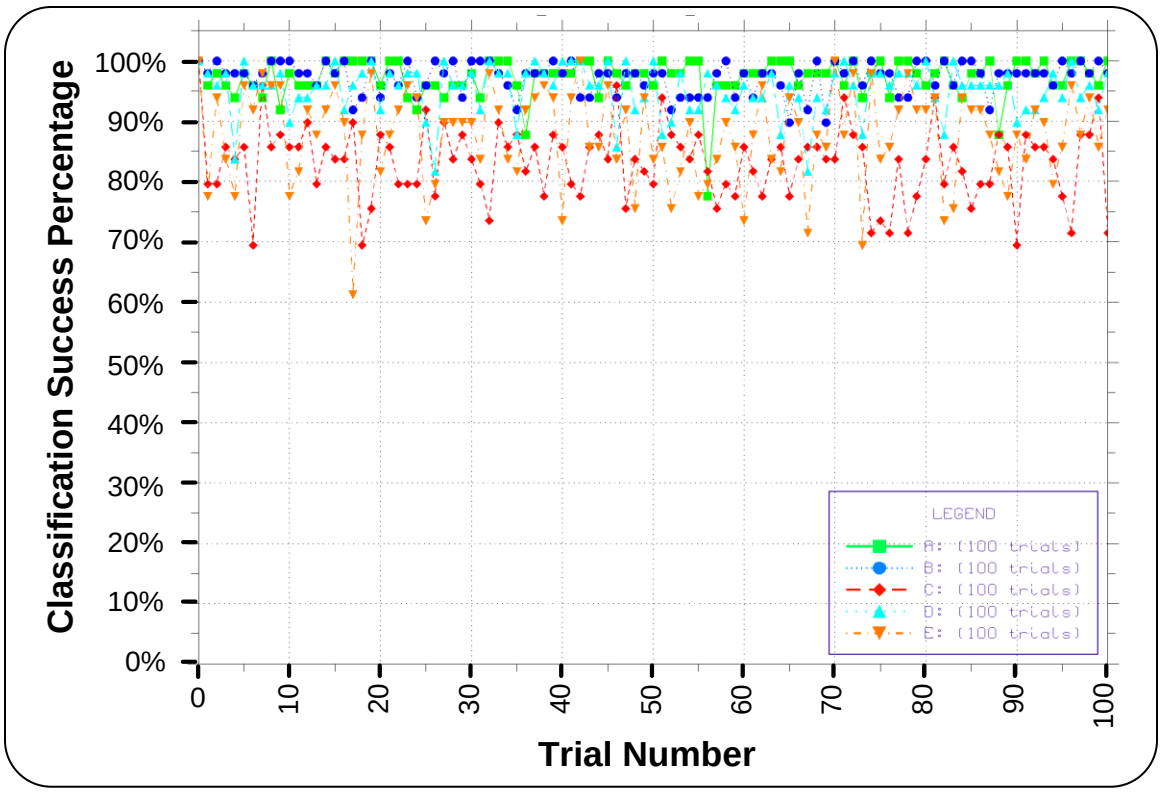


Figure 28: Global Classification – Alignment Errors Tolerated (Medium Range)

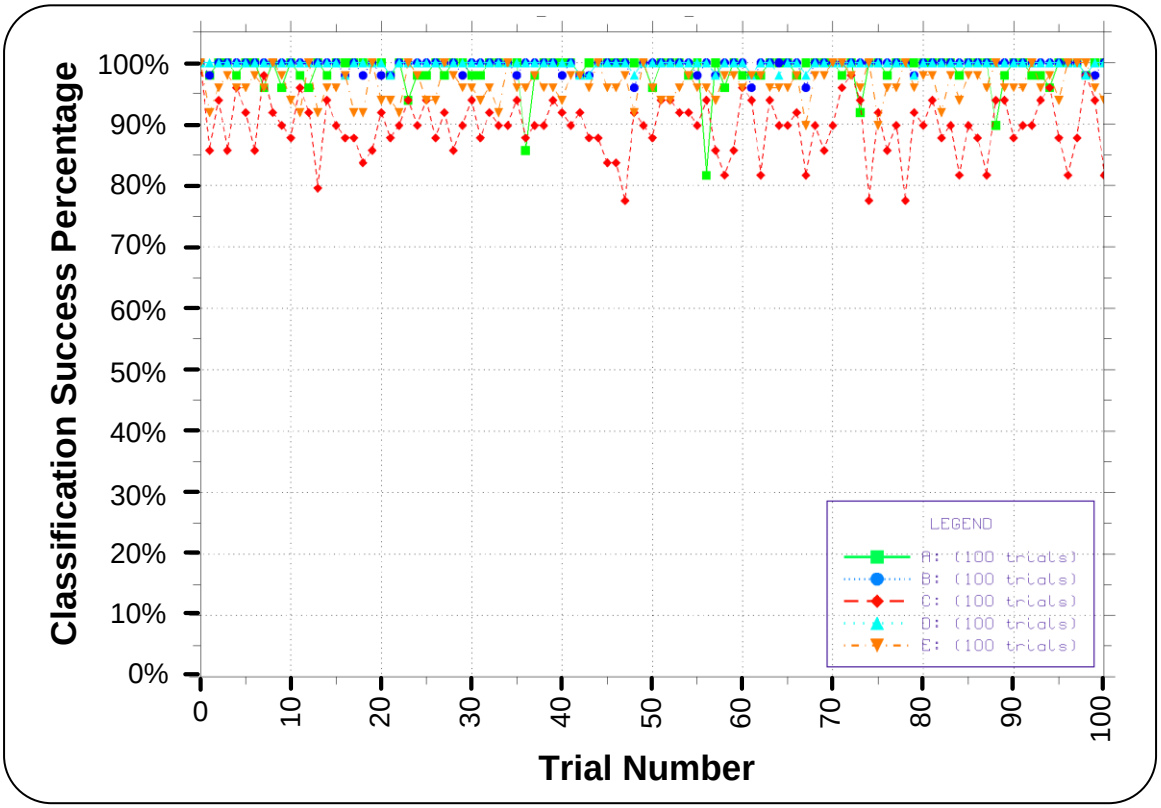


Figure 29: Global Classification – No Alignment Errors Tolerated (Medium Range)

7.2.3 Experimental Results- Performance Based on Rejection Strategy

Table 5 shows three cases for each of the WFP algorithm variants: training and classification using all received samples. Misaligned samples are discarded during classification only and discarding misaligned samples are discarded during both training and classification. With both of the algorithms, at medium range, RF sources 'C' and 'E' are most often confused with other nodes (nodes 'A' and 'D', respectively).

Table 5: Mean Classification Accuracy Comparison

Node ID:	***A***	***B***	***C***	***D***	***E***	***Overall Average***
Global Method - Including message samples with alignment errors during both classification and training	89.85% [86.77%, 92.93%]	90.93% [89.41%, 92.45%]	77.82% [74.96%, 80.68%]	92.8% [91.21%, 94.39%]	79.96% [77.51%, 82.41%]	86.27% [55.27%, 100%]
Global Method - Including message samples with alignment errors during classification only	97.23% [96.59%, 97.87%]	97.23% [96.71%, 97.75%]	82.92% [81.73%, 84.11%]	95.19% [94.37%, 96.01%]	87.51% [85.99%, 89.03%]	92.02% [78.93%, 100%]
Global Method - No message samples with alignment errors included during classification or training	98.69% [98.13%, 99.25%]	99.57% [99.38%, 99.76%]	89.47% [88.56%, 90.38%]	99.78% [99.65%, 99.91%]	96.37% [95.87%, 96.87%]	96.78% [89.8%, 100%]
Local Method - Including message samples with alignment errors during both classification and training	88.76% [87.06%, 90.46%]	88.25% [86.58%, 89.92%]	44.33% [42.35%, 46.31%]	93.93% [92.81%, 95.05%]	72.36% [70.55%, 74.17%]	77.53% [55.68%, 99.37%]
Local Method - Including message samples with alignment errors during classification only	97.65% [97.03%, 98.27%]	93.86% [93.14%, 94.58%]	44.98% [43.28%, 46.68%]	90.96% [89.8%, 92.12%]	77.37% [75.8%, 78.94%]	80.96% [64.95%, 96.98%]
Local Method - No message samples with alignment errors included during classification or training	99% [98.36%, 99.64%]	98.59% [98.26%, 98.92%]	49.69% [48.15%, 51.23%]	99.63% [99.44%, 99.82%]	85.27% [84.09%, 86.45%]	86.44% [74.36%, 98.51%]

Nodes 'A', 'B' and 'D' are rarely confused with other nodes. Using the Global method,

the average classification accuracy improves for all RF sources when misaligned samples are excluded only during classification. When misaligned samples are discarded during both training and classification, average classification accuracy performance improves again for all RF sources.

The mean classification performance for each RF source improves in a similar fashion with the Local method. However, the variability of classification accuracy does not change as much as with the Global algorithm, as can be seen from the width of the 95% confidence intervals for the overall average, taken over all RF sources. The classification for nodes 'C' and 'E' remains poor with the Local method, even after samples with alignment errors are discarded during both the training and the classification stages. This indicates that there is a shortage of 'close' error-free templates for the specific reversal mean values of the messages which are misclassified.

Although not apparent from the figures, misclassification errors occur more often at certain reversal mean values for specific RF sources, where symbol alignment errors are also occurring. This does not mean that *all* messages from an RF source with a specific reversal mean will be misclassified. However, errors for specific reversal means tend to occur 'nearby', in terms of the relative phase offset between the transmitter and receiver. We theorize that this is due to specific points in the sampled waveform where there is more phase ambiguity or phase marginality in the receiver for a particular RF source. Samples received with phase offsets at the extremities where wrap-around occurs (i.e. with a relative phase difference of 0 or $\pi/4$) are most often misclassified, as might be expected. The Local method is much more sensitive to this deterioration, which might explain why the expected performance benefits for an algorithm that selects training

samples based on reversal means did not materialize.

Increasing the number of neighbours or changing their spacing might improve the performance of the Local method. This was attempted with groups of 10 or 15 neighbours, but did not improve the classification accuracy results and is not presented here. There is no functional difference between the Global and Local methods, if the number of neighbour nodes is increased to a very large number, so very large groups were not analyzed.

The Local method could be modified to prune out samples that are in the range of the reversal means exhibiting error 'locality' using the rate of occurrence of alignment errors nearby. However, we have also not investigated this further, given our theorized cause for the behaviour. Complexity issues aside, tuning a WFP algorithm to avoid suspected areas of phase marginality is problematic, if these areas vary over time or with changes in the RF channel.

7.3 Experimental Results- Performance With Noise-Filtering

Rejection of outlier samples with high variability may also reject samples with symbol alignment errors. The 2.4GHz version of the IEEE 802.15.4 standard uses Direct Sequence Spread Spectrum (DSSS) which reduces the effects of narrow-band noise by spreading signals across a wide frequency range. However, if signals are noisy, we want the WFP training and classification algorithms to reject the corresponding samples as explained in Section 6.3.3.

Using a programmable percentage threshold, we discard the most noisy received samples in a given time period for each given RF source before starting the training process. In cases where there is little or no noise, this means that potentially valid signals are

discarded. However, the advantage is that such a scheme adapts to noisier channel conditions automatically, provided that the programmable percentage used corresponds roughly to the actual percentage of noisy traffic.

Table 6 shows the change in classification performance when noisy samples (i.e. solely based on the variance of the phase residual about the mean) are discarded. The noisiest 10% samples are discarded during the training process for each ID. There is a statistically significant improvement for the average classification accuracy of node 'A' with the Local method, but the results are less conclusive for the other RF sources, with overlapping CIs. The Global training method performance is identical for all RF sources with the two variants.

Table 6: Mean Classification Accuracy Discarding Noisy Samples (Medium Range)

Node ID:	***A***	***B***	***C***	***D***	***E***
Local Method - No training or classification alignment errors and no discarding of noisy samples	99% [98.36%, 99.64%]	98.59% [98.26%, 98.92%]	49.69% [48.15%, 51.23%]	99.63% [99.44%, 99.82%]	85.27% [84.09%, 86.45%]
Local Method - No training or classification alignment errors and discarding 10% noisiest samples	99.78% [99.64%, 99.92%]	98.8% [98.5%, 99.1%]	51.76% [50.2%, 53.32%]	99.8% [99.68%, 99.92%]	87.12% [85.95%, 88.29%]
Global Method – No training or classification alignment errors and no discarding of noisy samples	98.69% [98.13%, 99.25%]	99.57% [99.38%, 99.76%]	89.47% [88.56%, 90.38%]	99.78% [99.65%, 99.91%]	96.37% [95.87%, 96.87%]
Global Method - No training or classification alignment errors and discarding 10% noisiest samples	98.69% [98.13%, 99.25%]	99.57% [99.38%, 99.76%]	89.47% [88.56%, 90.38%]	99.78% [99.65%, 99.91%]	96.37% [95.87%, 96.87%]

Discarding of noisy samples is performed before their rejection based on alignment, so we conclude that the noisiest 10% also have alignment errors for the Global algorithm. From these medium-range experiments, the extra calculations to reject samples based on

excessive variance improve performance but may not be worth the required power on a WSN node for the Global algorithm. However, this cost/benefit tradeoff depends on the application, since discarding based on noise is cheaper computationally than discarding based on alignment errors. We choose to discard noisy samples, so that the Local and Global algorithms both perform at their highest level before we select the better of the two.

7.4 Principal Component Analysis-based WFP Algorithm Variant

Figure 30 shows the first three calculated Principal Component values for randomly selected messages received from five different RF sources using the Global training algorithm (with up to 30 alignment errors tolerated during the classification process and none tolerated during the classification process). The text labels next to each data point on the plot indicate the actual RF source ID letter followed by a unique numerical message ID (whose numerical value has no significance). The colour of the label is chosen to simplify the viewer's discrimination between the different RF sources.

The first Principal Component (PrC1) is displayed on the X axis and the second Principal Component (PrC2) is displayed on the Y axis, with the third Principal Component (PrC3) being shown as the colour (Z axis) of the square data point. The figure includes an inset bar chart showing the variability that is explained by the different Principal Components. The differences in the phase residual data for each ID can be seen as distinct clusters using the first two components alone. The third Principal Components is indicated using the colour of the data point in the figure. Therefore, geometrically-overlapping points on the graph are still distinguishable, provided the square colour is different.

The proportion of the variability that is being explained by each Principal Component

(and all of its predecessors) is shown using dark purple bars and the remaining variance that is still to be explained is indicated using adjacent red bars. The proportion of the variability converges to a value of 100% with all of the Principal Components and the remaining variance converges to a value of zero.

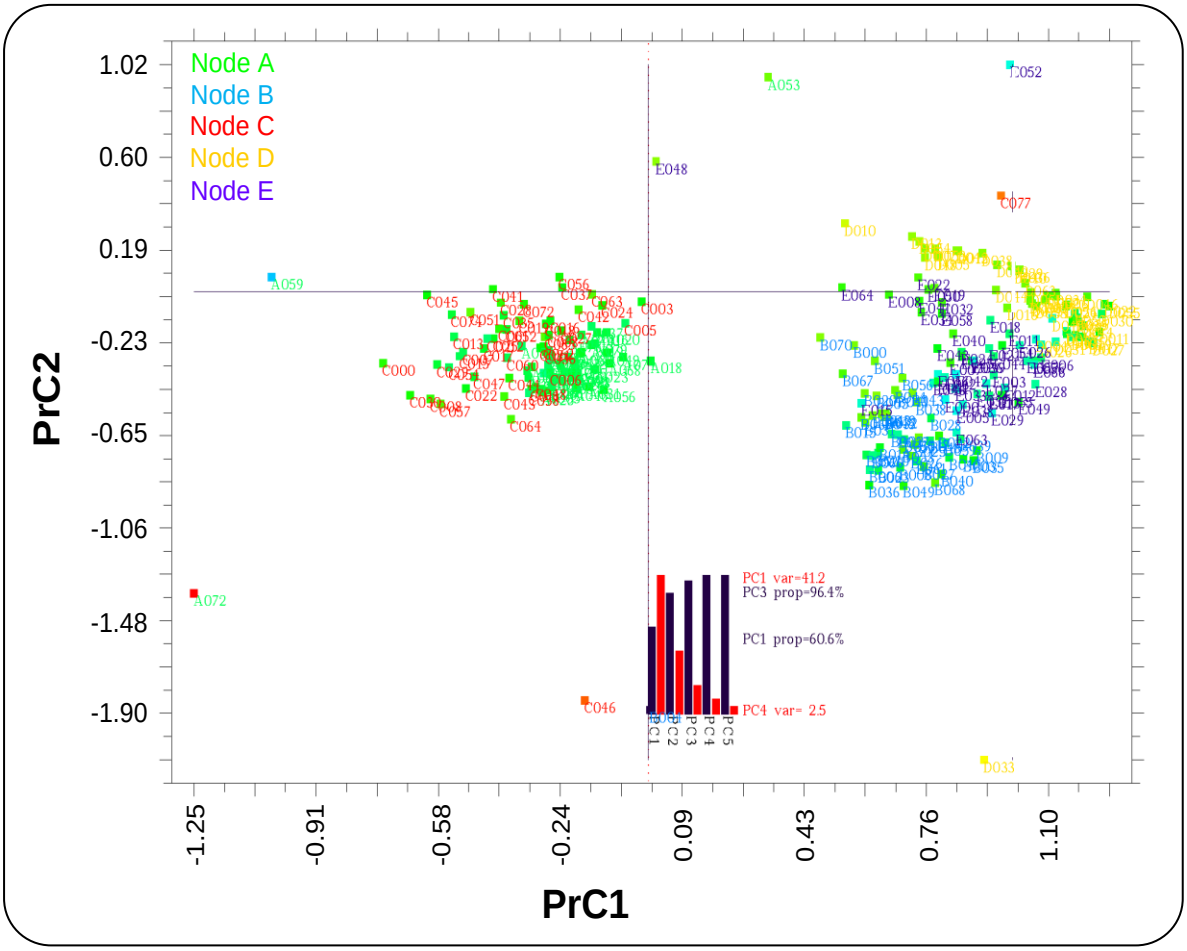


Figure 30: PCA Representation of Samples Received From Five Different RF Sources

In our classification experiments at medium range, for both SDRs, the first and second Principal Components explained over 65% of the variability for node 'C', and over 90% of the variability or higher for the other nodes. From Figure 30, node 'C' is most often confused with node 'A' and this was also apparent from Figure 25. In the figure, the first three Principal Components explain an average of 96.4% of the variability of the sampled data, giving a good representation of the classification performance for the five nodes

although the classification accuracy for node ‘C’ is still bad.

The messages that overlap in PrC-space are typically the messages that resemble each other and that are misclassified. The messages visible in the figure that are located outside the five main clusters all disappear if no errors are tolerated during classification.

The classification accuracy performance obtained using PCA-based classifiers for both training algorithms is summarized in Table 7.

The table is based on medium range data and no alignment errors are tolerated in any of the training methods. Note that PrC5 classification accuracy is obtained when using all of the first five Principal Components. As a reference, we include the results obtained earlier using the data taken directly from all chip positions.

The classification performance results with PCA are generally poorer than the results obtained using all of the phase residual values in each chip position, especially when extending the Global method. The differences in classification performance are most obvious when the sample variance is highest (e.g. with nodes 'C' and 'E'), which might be expected given that the Principal Component dimensions are aligned with maximum variability of the dataset.

With both WFP algorithm variants, the classification accuracy mean does not always continue to improve as more Principal Components are added, which may indicate calculation round-off error for the Global method. The performance deterioration for node ‘C’ with more Principal Components with the Local method is not understood. To optimize the worst-case performance, over all RF sources, the Global training method (with no tolerance for alignment errors during the training and classification stages) gives the best classification accuracy being over 89% for all WSN RF sources.

Table 7: Mean Classification Accuracy Using Principal Components

Method \ *Node ID*	*Principal Component number*	*A*	*B*	*C*	*D*	*E*
Local Training Method - No Alignment Errors using Principal Components	**PrC1**	93.68% [93.0%, 94.4%]	84.2% [83.1%, 85.3%]	74.63% [73.5%, 75.7%]	97.8% [97.4%, 98.2%]	67.76% [66.3%, 69.3%]
	PrC2	95.96% [95.4%, 96.6%]	96.16% [95.7%, 96.7%]	73.35% [72.4%, 74.4%]	99.14% [98.8%, 99.4%]	92.68% [92.0%, 93.4%]
	PrC3	97.1% [96.6%, 97.6%]	98.84% [98.5%, 99.1%]	69.92% [68.6%, 71.2%]	99.8% [99.7%, 99.9%]	88% [87%, 89%]
	PrC4	99.59% [99.4%, 99.8%]	99.08% [98.8%, 99.4%]	56.47% [54.9%, 58.1%]	99.8% [99.7%, 99.9%]	86.78% [85.8%, 87.8%]
	PrC5	99.59% [99.4%, 99.8%]	99.06% [98.8%, 99.4%]	56.31% [54.7%, 57.9%]	99.8% [99.7%, 99.9%]	86.74% [85.7%, 87.7%]
Local Training Method - No Alignment Errors using Chip Positions	**(N/A)**	99.78% [99.6%, 99.9%]	98.8% [98.5%, 99.1%]	51.76% [50.2%, 53.3%]	99.8% [99.7%, 99.9%]	87.12% [86.0%, 88.3%]
Global Training Method - No Alignment Errors using Principal Components	**PrC1**	88.59% [87.7%, 89.5%]	83.08% [81.8%, 84.3%]	74.53% [73.3%, 75.8%]	92.1% [91.3%, 92.9%]	70.51% [69.0%, 72.0%]
	PrC2	93.41% [92.7%, 94.1%]	98.9% [98.6%, 99.2%]	78.82% [77.8%, 79.9%]	99.76% [99.6%, 99.9%]	91.9% [91.2%, 92.6%]
	PrC3	95.84% [95.1%, 96.5%]	99.47% [99.3%, 99.7%]	81.73% [80.7%, 82.8%]	99.59% [99.4%, 99.8%]	93.17% [92.5%, 93.8%]
	PrC4	97.27% [96.5%, 98.1%]	99.37% [99.1%, 99.6%]	88% [87%, 89%]	99.53% [99.3%, 99.7%]	92.78% [92.1%, 93.4%]
	PrC5	97.33% [96.5%, 98.1%]	99.39% [99.2%, 99.6%]	87.92% [87.0%, 88.9%]	99.51% [99.3%, 99.7%]	92.74% [92.1%, 93.4%]
Global Training Method - No Alignment Errors using Chip Positions	**(N/A)**	98.69% [98.1%, 99.3%]	99.57% [99.4%, 99.8%]	89.47% [88.6%, 90.4%]	99.78% [99.7%, 99.9%]	96.37% [95.9%, 96.9%]

7.5 WFP Training/Classification Algorithm Complexity Analysis

We now present a brief complexity analysis for each algorithm considered. We estimate the number of processing operations required for classification and the amount of storage necessary for each training algorithm and summarize the results in Table 8 using a 'big O' nomenclature with the following parameters:

- T = number of training samples stored for each RF source ID. A larger value of T gives more lookup granularity. For the Local training algorithm, T could be viewed as a product $k \bullet J$, where k is the number of 'close' neighbour values to be used in the calculation (stored in an ordered list in memory) and J is the effective number of reversal mean bins used for the lookup. For the Global training algorithm, T has a value of 1, since a single average template is stored for each RF source ID.
- N = number of 'known'/encountered RF source IDs
- b = number of chip positions used for residual phase calculations
- A=number of processing cycles used to estimate the phase offset using the reversal mean. A is less than $O(b)$, since the reversal mean calculation is performed with a subset of the total number of chip samples (i.e. the ones at the positive and negative reversal points of the preamble phase shift sequence)
- C = number of processing cycles used to estimate the phase residual values used in a template. This is assumed to be equivalent to an average calculation based on evaluated phase residuals.

The Global training algorithm requires considerably less storage for templates, since only a single set of parameters is required for each known ID compared with the requirement for an indexed and comprehensive database of samples for the Local method. The Global training algorithm also requires fewer lookups and computations during the classification process. For the Local algorithm, multiple 'neighbouring' samples are averaged dynamically for each known ID, based on the input sample's reversal mean value, to derive a classification distance calculation template. The Global algorithm requires no such calculation, merely retrieving the single stored template value for each known ID.

Table 8: Complexity Analysis for Training Algorithms

Training Algorithm	***Training Template Storage***	***Classification Computations***
Local (chip-based)	O (TNb)	O (T^2) for list lookup, O (A) for reversal mean calculations and O($kNbC$) for N distance calculations
Global (chip-based)	O (Nb)	O (N) for list lookup, O (NbC) for N distance calculations
Local (PCA-based)	O (TN^2)	O (T^2) for list lookup, O (A) for reversal mean calculations, O (Nb) for PrC-space conversion and O (kN^2C) for N distance calculations
Global (PCA-based)	O (N^2)	O (k) for list lookup O (N^2C) for N distance calculations

The PCA-based versions of the algorithms are more efficient in terms of classification computation complexity and training storage complexity, when the number of Principal Components is significantly less than the number of chip positions being used for training and classification. However, the computation of the templates during training is more complicated than the non-PCA extensions of the algorithms since extra calculations are required to move each template into PrC-space. This may be acceptable, given our assumption that training is not performed as often as classification. The calculation of

these Principal Components are based on Eigenvalue decomposition, which can be reduced to a complexity of $O(N^2)$ [**61**] with $O(Nb)$ storage size complexity for the loading matrix.

From Table 5, it can be seen that the Global training method has better discrimination accuracy than the Local algorithm and is more feasible for implementation on a WSN. This implementation cost is measured both in terms of the number of classification CPU cycles and the amount of WSN node memory that is required to store the template information. Based on all of these results, we select the Global algorithm with no tolerance for alignment errors during training or classification as our method for further analysis. In the next subsection, we analyze the variations in performance for that algorithm alone.

7.6 Performance Variation

In this section, the selected Global WFP algorithm classification performance is studied using different receiver devices and under different RF channel conditions, and is also examined for stability over time.

7.6.1 Experimental Results- Receiver Effects

Five transmitting WSN nodes (labeled ‘A’, ‘B’, ‘C’, ‘D’, ‘E’) and two USRP1 SDR receiving devices (labeled ‘AA’ and ‘BB’) are used. The results presented so far use SDR 'BB' as the receiving device. The differences in classification accuracy, when using both SDR 'AA' and 'BB', are now analyzed in more detail.

We measure classification performance differences due to variations in the two receivers over different distances. The Global algorithm is used for classification, rejecting messages with alignment errors at both the classification and training stages. The

training and classification samples are randomly selected from data collected over a 24-hour test period (with the training samples always being selected to occur before the classification samples).

To achieve this, logged data is processed to delete individual transmissions not observed by both SDRs. Since transmissions are not synchronized, collisions can also occur during the testing process. Messages with invalid WSN node EEROM codes or non-identical sequence numbers are not included in the results. Classification accuracy generally deteriorates with increasing transmission distance for both SDRs.

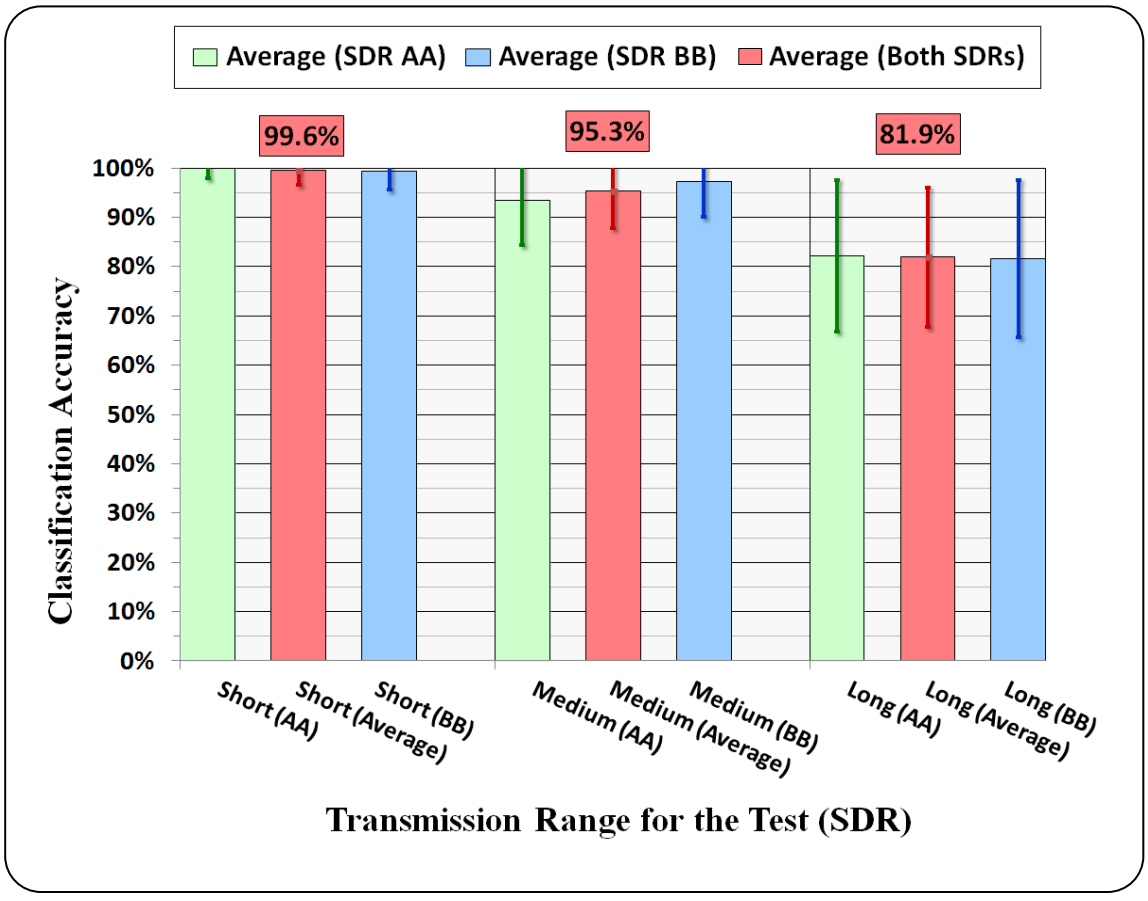


Figure 31: Mean Classification Accuracy for Two Receivers- Three RF Channels

Figure 31 shows the classification accuracy, averaged over all 5 RF sources, for both SDRs for the three different transmission distances used in our experiments. From the previous results with SDR 'BB' at medium range, we saw that WSN nodes 'C' and 'E' are classified more poorly. The average classification accuracy performance over all 100 trials for each RF source for both receivers is summarized in Table 9 over the different transmission distances.

Table 9: Mean Classification Accuracy (Simultaneous Reception)

Transmission Range	***WSN Node***	***SDR 'AA' Average Classification Accuracy (std. dev.)***	***SDR 'BB' Average Classification Accuracy (std. dev.)***
Short (1.5m LoS)	***A***	99.88% [99.78%, 99.98%]	97.43% [96.93%, 97.93%]
	B	99.84% [99.72%, 99.96%]	99.53% [99.27%, 99.79%]
	C	99.94% [99.87%, 100.01%]	99.96% [99.90%, 100.02%]
	D	100% [100%, 100%]	99.87% [99.77%, 99.97%]
	E	99.61% [99.34%, 99.88%]	99.62% [99.35%, 99.89%]
	Average	99.85% [97.98%, 100%]	99.28% [95.61%, 100%]
Medium (4m LoS)	***A***	99% [98.73%, 99.27%]	98.69% [98.13%, 99.25%]
	B	99.96% [99.90%, 100.02%]	99.57% [99.38%, 99.76%]
	C	67.49% [66.13%, 68.85%]	89.47% [88.56%, 90.38%]
	D	99.67% [99.52%, 99.82%]	99.78% [99.65%, 99.91%]
	E	95.57% [94.88%, 96.26%]	96.37% [95.87%, 96.87%]
	Average	92.34% [83.32%, 100%]	96.78% [89.8%, 100%]
Long (10m with intervening walls)	***A***	75.89% [74.41%, 77.37%]	66.14% [64.42%, 67.86%]
	B	92.34% [91.60%, 93.08%]	98.29% [97.90%, 98.68%]
	C	74.56% [73.21%, 75.91%]	66.08% [64.46%, 67.70%]
	D	77.63% [76.36%, 78.90%]	94.13% [93.27%, 94.99%]
	E	90.64% [89.72%, 91.56%]	83.51% [82.49%, 84.53%]
	Average	82.21% [66.81%, 97.62%]	81.63% [65.69%, 97.57%]

The table shows the variation in WFP classification accuracy, for the two different receivers, for three different 25-hour tests (each conducted over a different transmission distance). Tests were conducted over the same 25-hour daily interval (from 18:00 on the first day to 19:00 on the following day). We randomly select training samples between

18:00 and 23:00 on the first day and randomly select classification samples between 23:00 on the first day and 4:00 on the following day. We do this at the same time of day for each of the three distances.

Between receivers, there are differences in the measured samples for the same WSN node transmissions and there are also differences between sources. At medium range, WSN node 'C' is classified better at SDR 'BB' than at SDR 'AA'. At long range, there are significant differences between the two SDRs for all WSN nodes. Strong statements about differences in the classification accuracy average, taken over all RF sources, are not possible, because of the large 95% CI for the two SDRs. Figure 32 and Figure 33 show the classification accuracy results for each individual trial for the medium-range and long-range experiments, respectively.

At short range, classification is close to perfect for all RF sources, with the exception of node A at SDR BB with an average classification accuracy of 97%. There is also larger variance about that mean than for any other source node/ SDR combination at this transmission range. In general, larger classification accuracy variability is associated with poorer classification accuracy mean performance, which deteriorates as transmission distances are increased.

At medium range, WSN nodes ‘A’, ‘B’ and 'D' are classified almost perfectly by both SDRs and both SDRs have more difficulty classifying nodes ‘C’ and ‘E’. At the longer transmission distances, node 'C' classification accuracy is the poorest and the most variable for both SDRs. However, this is not the case at short range.

The same SDR is not always better at identifying the same source as can be seen by comparing the data for the medium and long transmission ranges for node ‘C’ in Table 9.

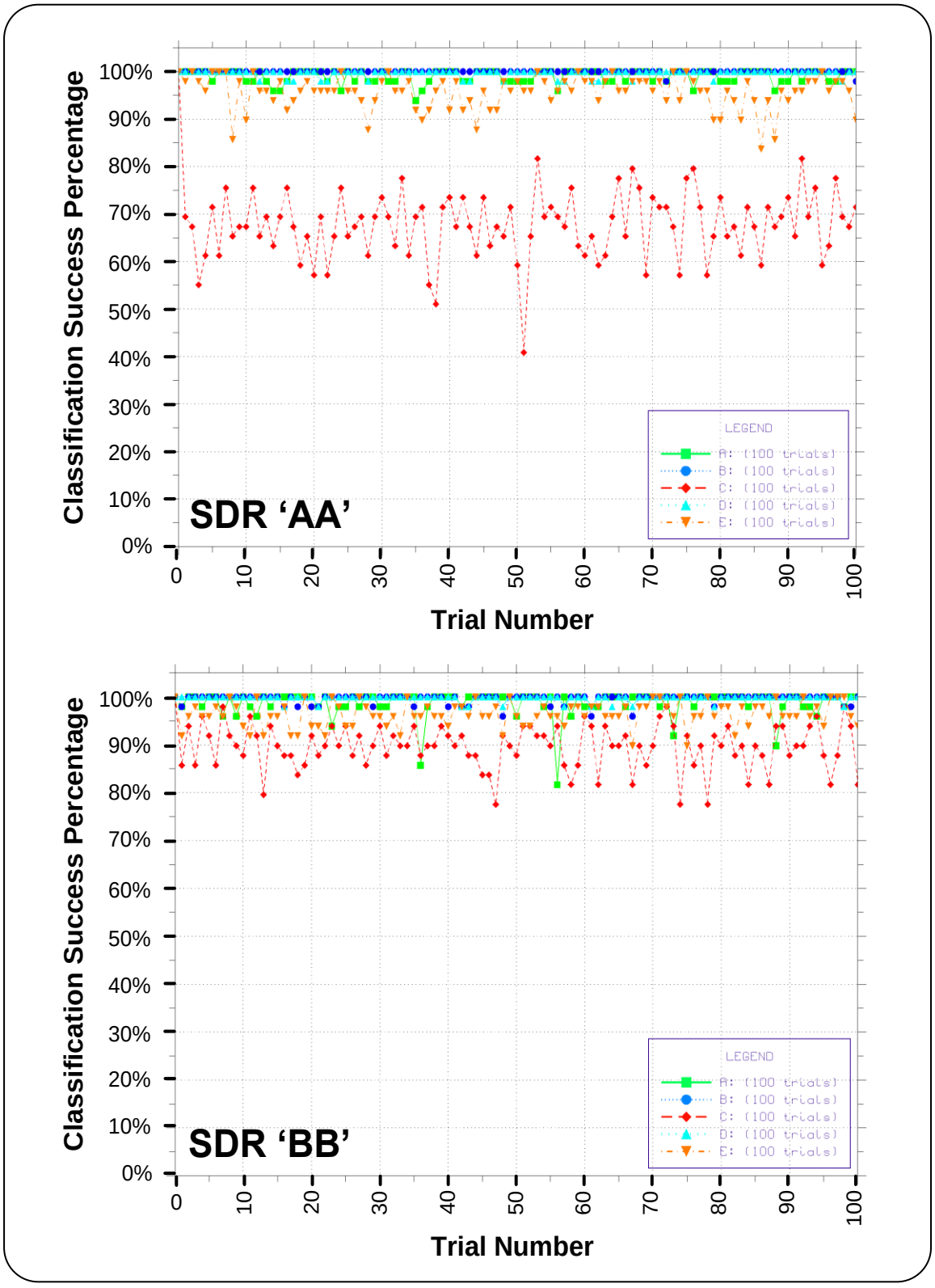


Figure 32: Simultaneous Classification Performance for Two SDRs (Medium Range)

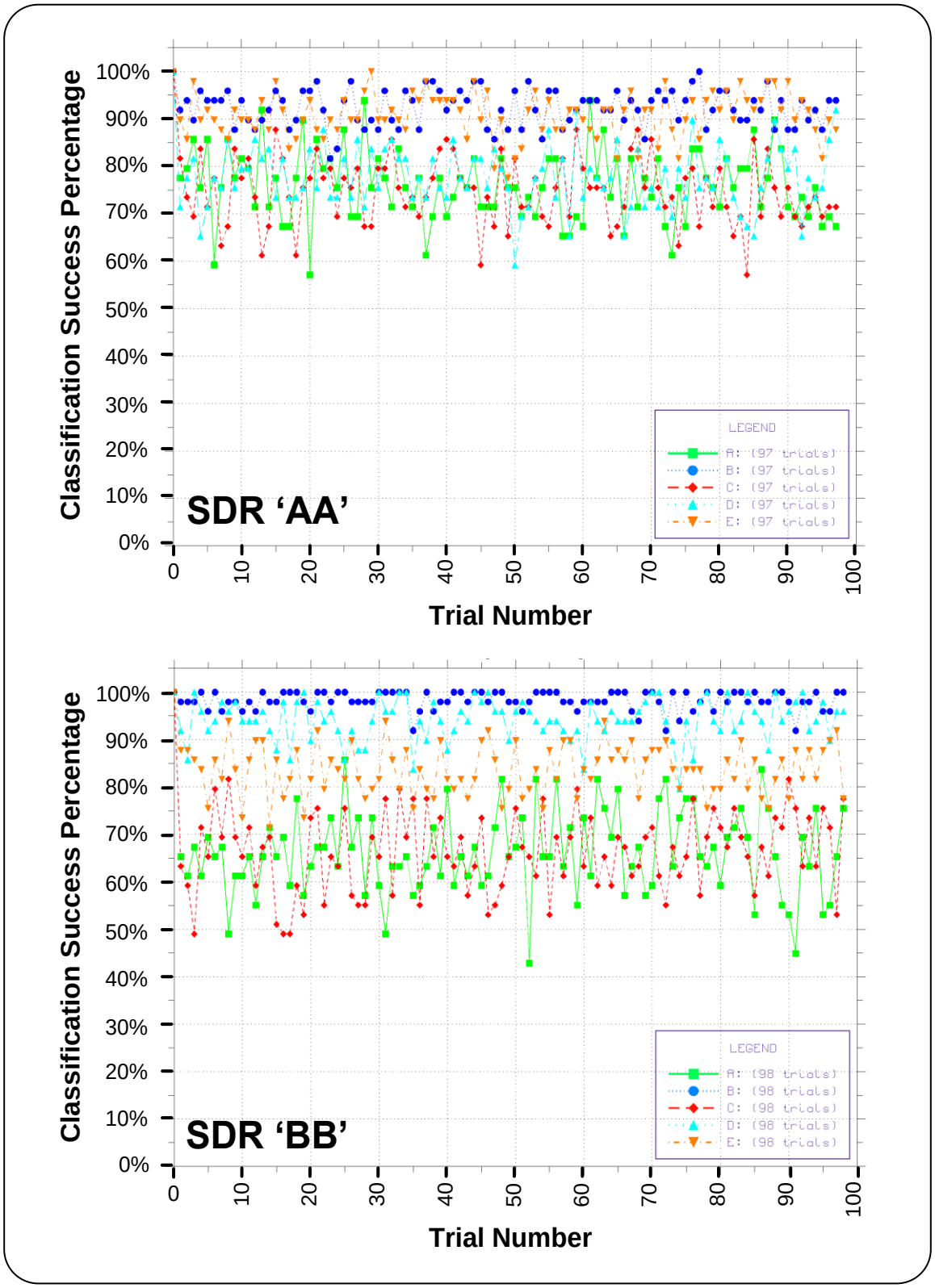


Figure 33: Simultaneous Classification Performance for Two SDRs (Long range)

Different gain settings in the USRP1 give similar results; we also used different settings of 65dB and 85dB from the nominal 45dB value. The difference in SDR classification is probably due to different levels of phase noise present in the receive signal path of the receivers or might be caused by RF channel discrepancies.

Our knowledge of the hardware and the sample size of two USRP1 receivers is too small to make any definitive statements about the first source of variability, but we study the differences from the RF channel in the next section. The RF channel cannot be assumed to be identical for the three different tests, since they were each held a day apart. However, daily variations from Internet activity or temperature changes are likely to be similar.

Figure 34 shows the results from six random trials, viewed in PrC-space, taken at the three different transmission ranges, using both SDRs. Classification accuracy is similar for the two SDRs for each of the different RF sources over distance. The relative classification distances between the different RF sources are not identical at each SDR and this can be seen using PCA. For example, the tighter clustering for node 'D' at SDR 'BB' at long range is apparent and the classification performance variation for node 'C' at medium and long range is also visible for both SDRs.

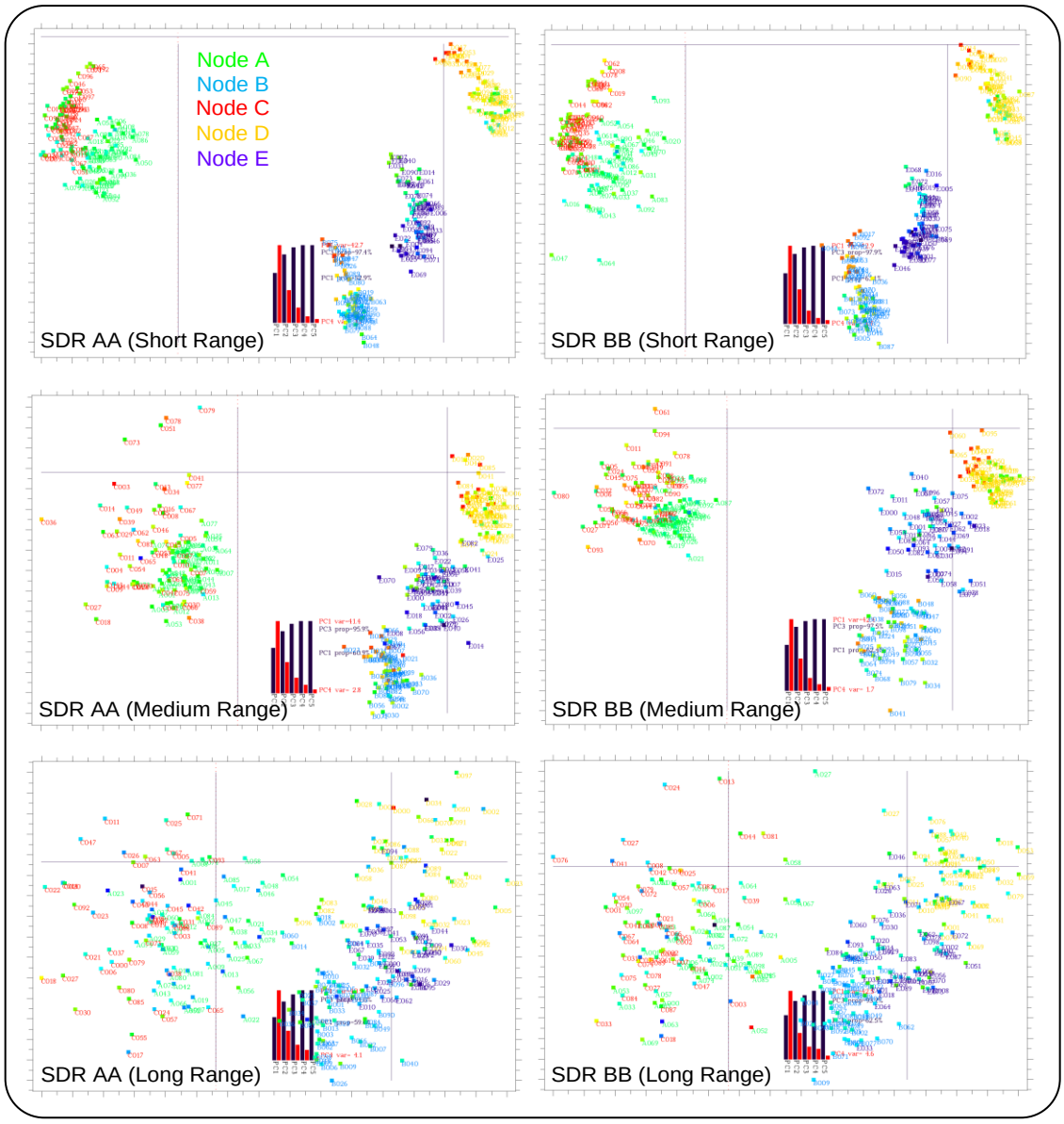


Figure 34: Principal Component Analysis (Classification Accuracy Variation)

7.6.2 Experimental Results- RF Channel Effects

The receiving antennas of the two SDRs are arranged with a separation of 26 cm between them, corresponding to two 2.4GHZ wavelengths. This yields uncorrelated measurements for the two receivers [**62**]. The antennas of the WSN nodes are set in a vertical position and aligned to be in the same orientations as each other (flat sides oriented towards the receivers).

During the experiment, small movements in the positions of the WSN nodes and the SDR antenna affected the numbers of transmissions that were simultaneously observed by both SDRs. An attempt was made to make transmission conditions as similar as possible for each node, but variability was observed in spite of these efforts. We accept this variation as a practical constraint and assume that such conditions could also exist in a real network.

To determine whether the classification accuracy is affected more by the receiver or by the RF channel, we repeat the long-range 25-hour experiment using the spatially-diverse SDR antenna spacing. Rather than co-locate the antennas to create an identical RF channel, each SDR was positioned in the same location previously occupied by the other SDR, which we term 'Position 2'. We attempt to keep wiring and all other surrounding objects in the same positions as with the original 'Position 1' (see Figure 24). The same nominal gain setting (45 dB) is used for all transmission ranges.

While it is unlikely that the RF environments are exactly identical for the two positions, they should be very similar. The two SDRs have the same metal enclosure and their antenna positions were set up to be as similar as possible. The RF sources were also maintained in their previous position, although a slight lateral movement of about 1cm was required for the RF source mounting platform to ensure approximately the same arrival rate of messages from all RF sources at both SDR receivers.

Table 10 shows the average classification accuracy for each SDR in each of the two locations for each RF source. Classification accuracy deteriorates for nodes ‘B’ and ‘E’ on both SDRs in Position 2. Node ‘C’ is recognized more consistently by both SDRs in Position 2. Node ‘A’ and ‘D’ classification are less affected by the change in position.

The upper portion of Figure 35 plots the same long-range data in bar chart form. The lower portion of Figure 35 plots the differences between the classification accuracies, grouped in two different ways. One set of bars (in green and red) plots the differences in average classification accuracy between the two SDRs at a specific position. The other set of bars (in blue and yellow) plots the difference in average classification accuracy between the two positions, for each SDR.

Table 10: Mean Classification Accuracy for Different Locations (Long Range)

Original Position (Position 1)			***Node***	***New Reversed Position (Position 2)***		
SDR AA Classification	***SDR BB Classification***	***Mean***		***SDR AA Classification***	***SDR BB Classification***	***Mean***
75.89% [74.41%, 77.37%]	66.14% [64.42%, 67.86%]	71.02% [64.10%, 77.94%]	***A***	74.10% [72.63%, 75.57%]	59.08% [57.35%, 60.81%]	66.59% [59.70%, 73.48%]
92.34% [91.60%, 93.08%]	98.29% [97.90%, 98.68%]	95.32% [92.77%, 97.87%]	***B***	76.55% [75.23%, 77.87%]	90.08% [89.13%, 91.03%]	83.32% [78.37%, 88.27%]
74.56% [73.21%, 75.91%]	66.08% [64.46%, 67.70%]	70.32% [63.90%, 76.74%]	***C***	98.61% [98.31%, 98.91%]	97.92% [97.51%, 98.33%]	98.27% [96.73%, 99.81%]
77.63% [76.36%, 78.90%]	94.13% [93.27%, 94.99%]	85.88% [81.22%, 90.54%]	***D***	69.30% [68.01%, 70.59%]	97.35% [96.80%, 97.90%]	83.33% [79.06%, 87.60%]
90.64% [89.72%, 91.56%]	83.51% [82.49%, 84.53%]	87.08% [82.9%, 91.26%]	***E***	55.43% [53.76%, 57.10%]	46.57% [45.26%, 47.88%]	51.00% [44.53%, 57.47%]

Looking at the lower portion of the figure, we see that the receiver and the RF channel both have an effect on classification performance. The difference in classification accuracy, either for a different receiver or at a different location, depends on the RF source in question. There are still certain RF sources which are classified better by a specific SDR, most notably node 'D', which is classified between 17% and 28% better by SDR BB than SDR AA, depending on the RF channel being used.

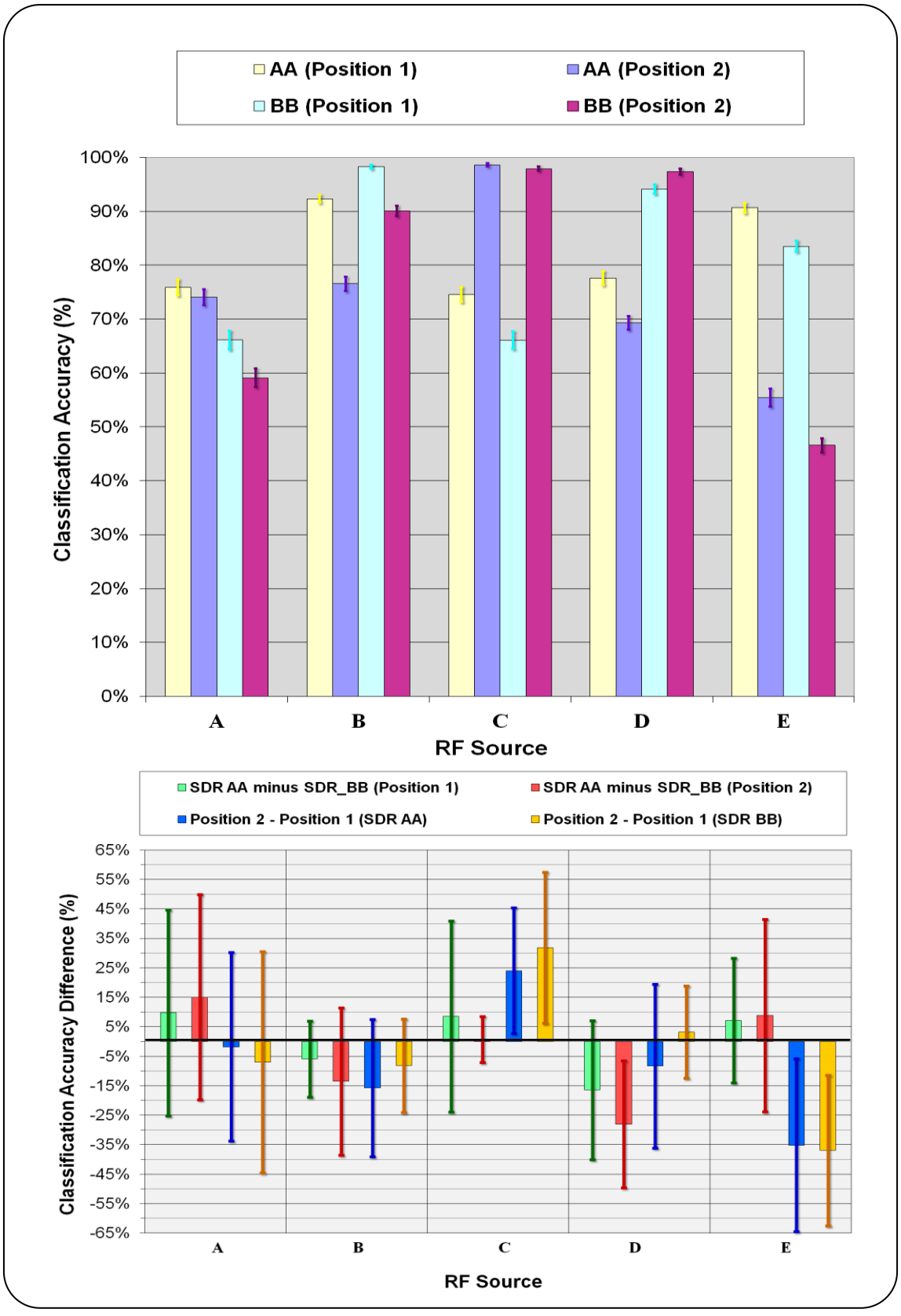


Figure 35: Mean Classification Accuracy by Source- Two Receiver Positions

For a given SDR, Nodes 'A', 'B' and 'D' are the least sensitive to changes in the RF channel. The 95% confidence intervals include zero for the yellow and blue bars for these nodes. Node ‘C’ and node 'E' classification accuracy are affected by the RF channel, with node 'C' almost always being classified better in Position 2 and node 'E' almost always being classified better in Position 1. This is consistent between SDRs, indicating that the RF channel is a more significant factor than the SDR for these two RF sources.

7.6.3 Experimental Results- Performance Stability over Time

The WFP generated by an RF device could vary over time. For example, the electrical oscillators used in these devices often increase their frequency as the crystal inside the oscillator gets lighter over time. Generally, collecting data over months or even years is required to observe such changes in electronic performance. Because of the impracticality of these timescales, we do not measure the performance stability for electronic device changes that are due to aging. Subjecting devices to large voltage and/or temperature extremes or mechanical shaking can sometimes be used to accelerate changes in device performance over time. However, we cannot afford the potential device casualty risks associated with these methods, even if the performance changes resulting from them are representative of the ones arising from normal aging.

RF device performance is also affected by fluctuations in the power supply voltage. WSN nodes are often powered by a single battery, whose voltage can change over time. The characteristics of this deterioration vary with different battery technologies. Device operation is guaranteed within a specified range of voltages and the RF IC devices used on WSN nodes often have internal voltage regulation circuits, to generate stable internal

voltages, even when battery voltages vary. For example, the voltage regulator used in the RF device on our WSN node accepts an input supply range of 2.1V to 3.6V. Therefore, performance variation over time with different battery voltages is unlikely.

In a practical network, the temperature will vary as a function of the time of day. For example, temperatures go up as the sun rises. Also, there are seasonal changes as the average height of the sun changes and as the heating and ventilation patterns in buildings change with these conditions. Temperature changes can affect electronic device performance. For example, colder CMOS technology parts tend to have faster signal rise times and shorter propagation delays. Scientific measurement of the performance of devices over different temperature is not straight-forward, especially one with industrial temperature ranges (e.g. -40 to 85 degrees Celsius) like ours. Calibrating the actual temperature of the die of an integrated circuit requires the use of thermocouples attached directly to the device package and detailed knowledge of the thermal characteristics of that package.

Without measuring classification performance using more controlled conditions, we do not make strong claims about the stability of WFPs with temperature variations. However, we attempt to reduce such effects while measuring the performance variation over time. We have already analyzed WFP variation with receiver changes in Section 7.6.1 and with the RF environment in Section 7.6.2.

The experiments in this section were conducted using the same indoor locations for transmitters and receivers over a 25-hour period and we have identified the night time as a ‘quiet’ period of relative stability in the RF environment. While average outdoor

temperatures decreased by about five to ten degrees Celsius from their initial value during the daylight hours, they remained relatively stable overnight.

We examine the stability of the WFP classification process as a function of time and use the quieter period of the day for the training interval. Measurements for the five different RF sources that we tested were made over three consecutive nights using the three transmission distances described already. All five RF sources were transmitting simultaneously. Classification was studied in successive intervals (Figure 36).

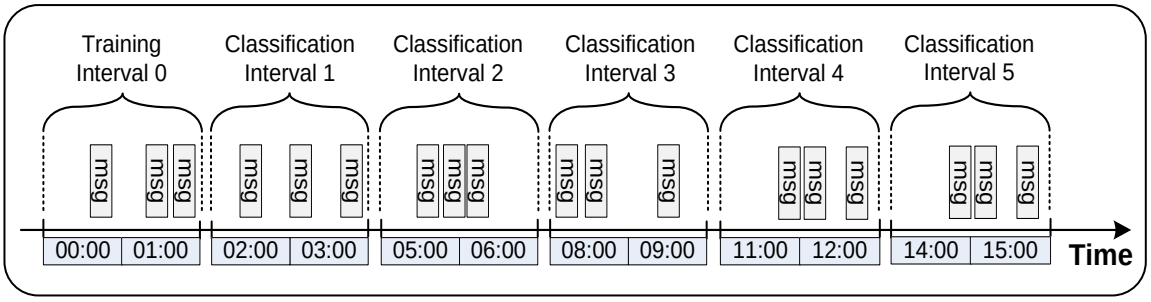


Figure 36: Disjoint Training and Classification Intervals

We use the same statistical cross-validation technique to measure classification performance as before, but use disjoint time intervals for the selection of the training and classification samples as shown. We randomly select the training samples from the relatively quiet one-hour time period (interval 0) which starts at midnight. We then randomly select classification samples from one of the five subsequent disjoint one-hour time intervals and the results are shown in Figure 37.

The training interval has the same duration as each of the different classification intervals, but the classification intervals are offset by two hours from each other. The classification accuracy averages for each RF source in each time interval are shown in Figure 38. The Y axis displays the classification success percentage rate average, calculated over the 100 trials and the X axis is broken into subsections for each RF source

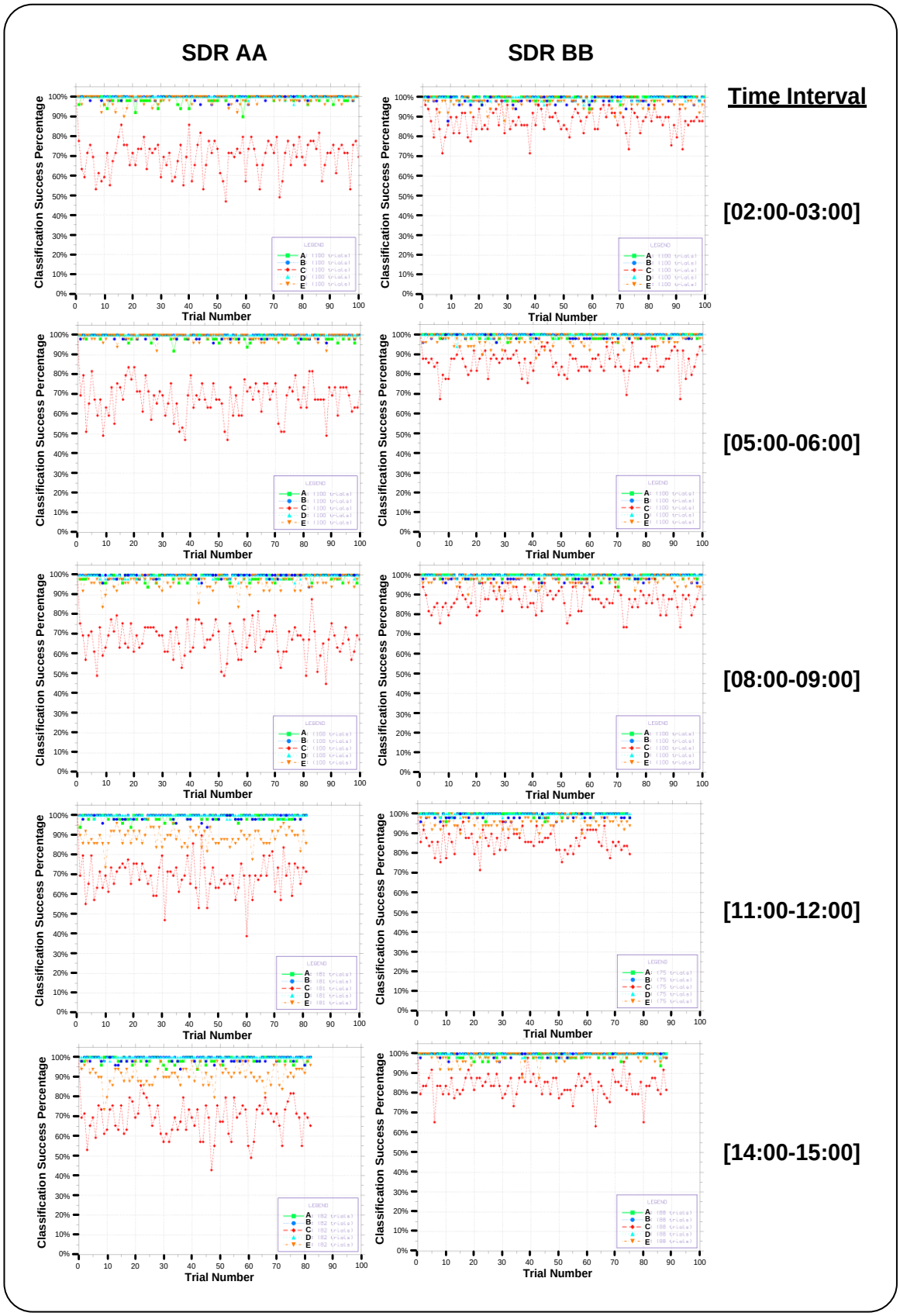


Figure 37: Classification Accuracy During Different Time Intervals (Medium Range)

with the results from each classification interval shown for both receiving SDRs. SDR AA is shown with the five blue/green bars on the left in each cell. SDR BB is shown with the five pink/orange bars on the right. The vertical line at the top of each bar is a 95% confidence interval centered at the mean value.

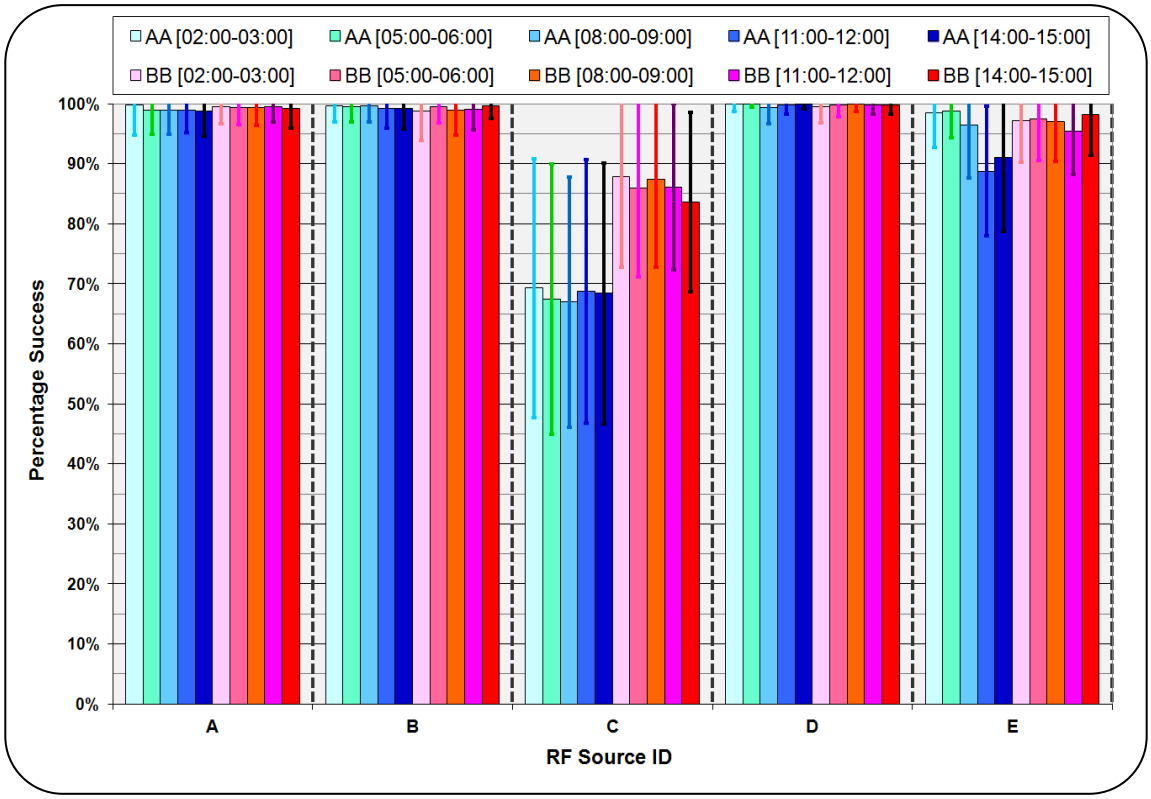


Figure 38: Mean WFP Classification Accuracy Over Time (Medium Range)

Average classification accuracy varies less than 1% for all nodes during the measurement period, except for nodes 'C' and 'E', with worst case degradations of 2.5% and 7.7%, respectively. For node 'E', the 7.7% degradation occurs during the fourth interval at SDR AA. SDR BB is also affected at the same time, but to a smaller degree, which indicates some kind of change in the RF source or RF channel that affects that RF source (the classification of other nodes is stable). In the final interval, average classification accuracy recovers at both SDRs for node ‘E’. In general, the variation in classification accuracy for any given node/SDR combination is very consistent.

7.7 Summary of Results

In this chapter, we presented the classification accuracy results for our USRP1-based WFP algorithm. We determined that the best performance is obtained using the Global algorithm, rejecting samples with alignment errors during both training and classification. The Local WFP algorithm has generally poorer performance. Computations and storage are both more expensive.

In our testing at medium range, the noisiest 10% of samples also contain alignment errors. Our selected WFP algorithm rejects these same samples, so removing them has no effect on the classification performance. Noise filtering is a computationally-inexpensive pre-processing step before the WFP algorithm. Therefore, it can still be used to reduce the number of samples being processed for training or classification.

We used Principal Component Analysis as a basis for WFP classification, with poorer results than either of the other two methods. However, this technique is useful for the visualization of classification results. PCA can reduce WFP algorithm calculation complexity, when large numbers of chip coefficients are being used with small numbers of templates.

Specific RF sources are classified more accurately on specific receivers and others are classified more accurately over specific RF channels. Generally, the difference in the relative classification performance of receivers for a specific RF source remains stable over different RF channels, implying that this receiver performance advantage is affected more by an interaction between the receiver and the transmitter than by variations in the RF channel. We observed other nodes where the receiver relative performance advantage changed more strongly with the RF channel.

For a given source/receiver pair and RF channel, our WFP algorithm classifies consistently over time, with an average classification accuracy variation which is always less than 8% (and usually less than 2%) in a 12-hour measurement period. Note that the RF channel propagation characteristics during our experiment are controlled (i.e. no intentional transmitter or receiver movement), but radio interference conditions are not controlled at all. We conclude that our WFP algorithm has stable classification performance for a given pair of stationary transmitter and receiver devices, communicating over a realistic RF channel.

8 Chapter: Wireless Fingerprints at the Network Layer

With human fingerprints, it is useful to be able to identify the fingerprint of an individual based on measurements made in different countries under different conditions of temperature, humidity, etc. by different police forces. Similarly, in a wireless network, if a WFP is stable or changes predictably under different RF conditions and with different receiving devices, identification using WFPs is facilitated.

We present the results of an experimental study that uses templates generated by a different node than the one which is classifying a particular RF source. We show that classification performance is comparable to the case when using templates that are generated directly on the classifying node. If the templates generated by the non-classifying node are generated over a shorter distance to the RF source, classification performance is shown to be better than when using templates that are generated locally over a longer range.

Provided node classification results are significantly different and independent of each other, WFP classification decisions can be aggregated over multiple nodes (perhaps weighted by proximity to the source node) to improve overall classification accuracy. Further, the WFP templates themselves might be aggregate-able, although investigation of detailed algorithms to achieve this is left for future work.

We present a statistical analysis of the relative WFP misclassification error performance between different USRP1 receivers, when classifying the same messages from each of the five WSN RF sources. We analyze the USRP1 WFP classification errors for independence. We also present the results of a statistical analysis of paired message data to measure the value of producing WFPs on different nodes.

We can use WFPs to establish security in a network. A small amount of high-level research has been performed showing how RF Fingerprints might be used in Intrusion Detection systems [**10**][**36**]. In Section 8.6, we present a new method to derive a shared secret key that is valid for a group of nodes in a neighbourhood that contain both honest and malicious nodes [**6**]. The WFP is used to provide an authentication service and is incorporated tightly into the execution of the protocol. In this way, the message source and the message contents are tied together during protocol execution.

8.1 Experimental Results- WFP Template Alternative (Different Receiver)

Table 11: Mean Classification Accuracy With Partner Training Data (Long Range)

Training Data	***AA***	***BB***	***BB***	***AA***
Classification Data	***AA***	***BB***	***AA***	***BB***
A	75.89% [74.41%, 77.37%]	66.14% [64.42%, 67.86%]	71.64% [70%, 73.28%]	70.95% [69.69%, 72.21%]
B	92.34% [91.6%, 93.08%]	98.29% [97.9%, 98.68%]	94.79% [94.06%, 95.52%]	97.22% [96.75%, 97.69%]
C	74.56% [73.21%, 75.91%]	66.08% [64.46%, 67.7%]	75.04% [73.59%, 76.49%]	67.05% [65.55%, 68.55%]
D	77.63% [76.36%, 78.9%]	94.13% [93.27%, 94.99%]	78.68% [77.53%, 79.83%]	91.78% [90.9%, 92.66%]
E	90.64% [89.72%, 91.56%]	83.51% 82.49%, 84.53%]	87.95% [86.89%, 89.01%]	87.23% [86.22%, 88.24%]
Overall Average	82.21% [66.81%, 97.62%]	81.63% [65.69%, 97.57%]	81.62% [65.44%, 97.80%]	82.85% [68.78%, 96.91%]

Table 11 tabulates the average classification accuracy results for a 25-hour long-range experiment, in which each SDR classifies incoming samples using the templates generated by another SDR. This experiment indicates that different receiving nodes may be able to share WFP template information or collaborate using their WFP template information across a network. The same time intervals are used for training and

classification by both SDRs with the receivers arranged in Position 1. The same methods are used for random sample selection as in the previous experiments.

The upper portion of Figure 39 shows the tabulated results and the lower portion of the figure shows some pertinent sets of differences in the different cells, along with their 95% confidence intervals. The red/green bar set in the lower portion shows the difference between SDR AA and SDR BB classification accuracy when both SDRs are using templates generated by SDR AA or SDR BB, respectively. The blue/yellow bar set shows the change in classification accuracy at the same SDR (either SDR AA or SDR BB, respectively), when switching from templates generated by SDR AA to using templates generated by SDR BB.

Each SDR is better at recognizing specific RF sources, but overall average classification accuracy performance is similar, whether templates are generated by the same SDR or not. Confidence intervals for some RF sources overlap when the same classification data is used. Because there are only two SDRs, the 95% CIs in the lower portion of the figure are large, making comparison harder. The CIs for all the different classifier/trainer combinations include zero.

Classification accuracy for a specific RF source is similar for each SDR, regardless of which training data is used. The average values of the red/green and the blue/yellow bars do not have the same polarity for all RF sources, but have the same polarity and similar magnitude for a particular RF source. No polarity difference is statistically significant, with the 95% confidence intervals including the zero difference value in all cases, although Node ‘D' is often classified more accurately (average of 15% better) by SDR BB than by SDR AA (significant at a 90% confidence level)..

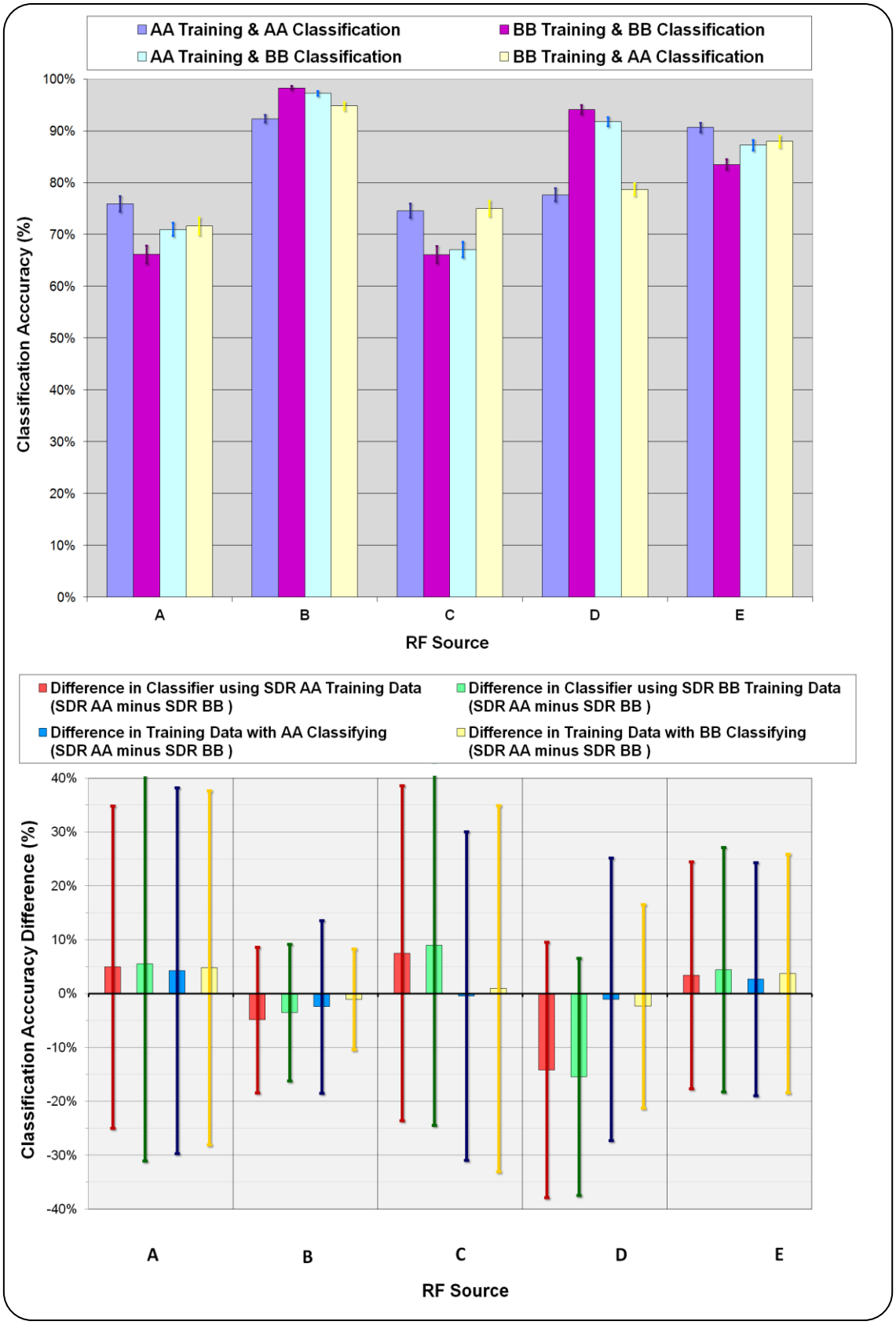


Figure 39: Classification Using Training Templates from Another Receiver (Long Range)

8.2 Experimental Results- WFP Template Alternative (Different Transmission Distances)

We have shown that WFP templates from another node at a similar range can be used for classification with little change in performance. We have also shown that average WFP classification accuracy deteriorates with distance. We now demonstrate improvement of average classification accuracy at long range by using WFP templates generated at short range. We also show that these short-range templates can be generated by a receiver which is different from the one doing the classification.

Using the previous data generated during the short range and long range testing, we measure the average classification accuracy for messages received over long range using templates generated at both short and long range. There are 8 possible permutations: training range (as short or long), training SDR (as AA or BB), classification SDR (as AA or BB).

The average classification accuracy for each of these 8 permutations for each RF source is graphed in Figure 40 for each RF source. The average of the average classification accuracy for all RF sources is also shown in the figure along with a 95% CI for each of the 8 permutations. The left-most group of four bars, for each RF source, corresponds to the use of long-range training data. The right-most group of four bars corresponds to the use of short-range training data. The left-most green/blue pair of bars and the right-most green/blue pair of bars corresponds to SDR AA classification, with long-range and short-range training data, respectively.

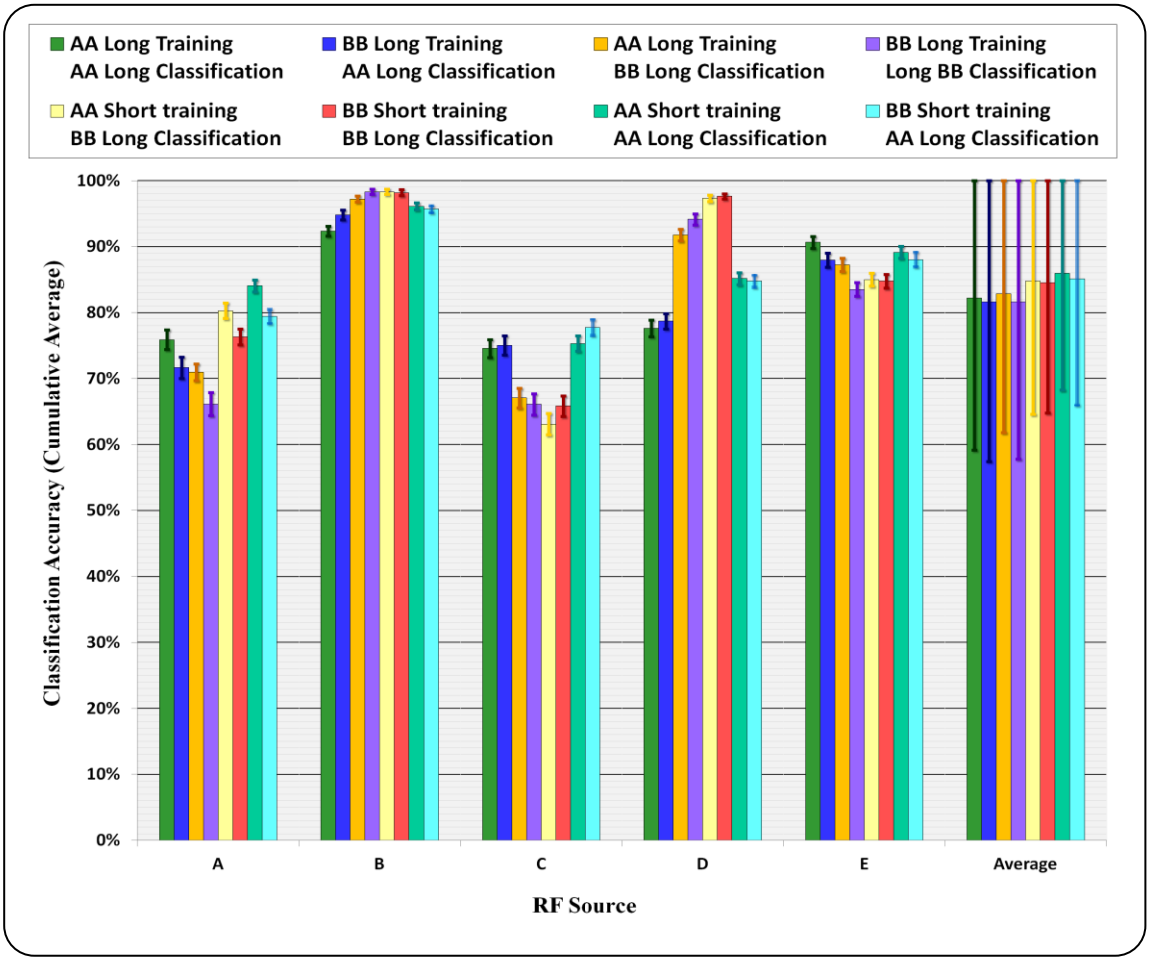


Figure 40: Mean Classification Accuracy (Template Variations)

For all RF sources, with both SDR AA and SDR BB, short-range templates either maintain or improve classification accuracy, most notably with nodes 'A', 'B' and 'D'. The classification accuracy for nodes 'C' and 'E' is approximately the same or slightly worse for the same classifying SDR, when short-range templates are used to classify the long-range data. SDR BB is significantly better than SDR AA at classifying node 'D', which was already observed, but classification improves at both SDRs when using short-range training data from either SDR.

The results from the previous two sections show that a particular RF source can be more accurately classified by a particular SDR and/or can have an RF channel that is more

conducive to accurate classification. Variations in classification accuracy over time could also explain these differences, since the experiments for different transmission distances were performed on different days. However, classification accuracy results are generally stable over time (see Section 7.6.3), so we consider this less likely.

All of the previous results increase our confidence that generating and using WFPs collaboratively in a network is both feasible and that the resulting classification accuracy is potentially stable enough to be useful. For example, nodes could be characterized at close range by a selection of other existing nodes during the deployment process. In the next section, we use statistical analysis techniques to provide further evidence of the feasibility of using WFPs at a network level.

8.3 USRP1 Classification Error Independence

A WFP algorithm which discriminates well between incoming signals, on a single node, can be extended to allow collaboration with other nodes in a network. Perhaps the WFP that is generated can be improved by collaboration or perhaps WFP classification results can be shared to improve accuracy. This chapter analyzes two basic characteristics that allow our WFP algorithm to be improved by executing it in a broader networking context.

So far, we have shown the feasibility of our training and classification algorithms for WFPs. We have shown that WFP templates for an RF source can be used by another node, to give similar classification performance. Using such a template, we have also shown that the classification performance can be improved with a template that was established over a shorter transmission distance from the RF source.

The network-level aspects of WFPs that we consider are:

- (Spatial consistency): WFP measurements should be consistent across the network, if we want to aggregate them directly using multiple nodes. This means that different receivers using different RF channels should be able to make WFP measurements for an RF signal that can be reconciled as being from the same source. Changes in the RF channel over time or with interfering RF sources can distort the WFP, but the RF measurement system must be able to filter out such noise or else operate properly in the presence of such noise. As an alternative to providing fully spatially consistent WFPs for aggregation, nodes can aggregate their link-layer classification decisions.
- (Temporal consistency): WFP measurements should be stable across the network over time. This means that measurements made at different times for the same RF source by the same receiver over the same RF channel can be reconciled as being from that source. This is important because of the time scales involved and potential forwarding delays for traffic at the network level. WFPs that ‘wander' slowly as a function of time might be acceptable, provided that they wander slowly enough to allow a networked database to evolve and keep track of the changes.
- (Error Independence): WFP classification errors should be independent across the network, if multiple nodes are collaborating to derive better aggregate WFP classification decisions. If errors at different WSN nodes are strongly correlated with each other, then the incremental improvement in WFP classification accuracy using a group of WSNs node will not outweigh the communication and co-ordination costs associated with the required sharing of information between

them. If the requirements of spatial and temporal consistency are met, the WFP itself can be aggregated across a network by multiple nodes and error independence is less important than the quality or accuracy of the WFPs being generated by each node. If errors are independent for different receivers, measurements made at different sites could be combined to improve the overall reliability of the WFP using group voting schemes (ignoring the presence of malicious parties in those groups) or other ensemble classification methods; see [**63**] for a recent review of different ensemble learning methods.

We have presented results for spatial and temporal consistency gathered on the USRP1 platform. We next study the independence of classification decision errors. The WFPs themselves might be aggregate-able, where multiple trusted nodes could combine their results derive a better template than possible from measurements made at a single site. Although we have not investigated this further, one method might be to create an average or a median of WFP templates taken from the group of trusted sites, which are closest to the RF source in question.

To measure the classification error independence of WFPs, we need a definition of independence and a formal test for independence. Probability theory defines events as being independent if the occurrence of one event does not affect the probability of the occurrence of the other event (i.e. the events have no influence on each other). More formally, the joint probability of two independent events can be found by multiplying the individual probabilities of the events together.

If different wireless nodes make classification errors independently of one another and authenticate a specific RF source with the same probability over time for a particular

transmission range, then the probability that m out of N nodes successfully authenticate that source is given by the binomial distribution. If a simple voting scheme is used and we assume all participants are honest, the probability tends to a value of 1 and does so very quickly for larger values of N and for better individual WFP classification performance (i.e. greater than 0.5).

In WSNs, the number of neighbours is of the order of 10 nodes (or less). Using an example of a group of 7 nodes, the nodes will fail to reach the correct group consensus 13% of the time if nodes have individual probabilities for correct WFP authentication of 0.7 each. However, the probability of the same group size failure drops to 0.3% if nodes can distinguish with an individual probability of 0.9. Larger group sizes do not require individual performance to be as high (see Table 12) but are less likely in WSNs.

Table 12: Probability of Correct Authentication by a Group (Independent Error Case)

Probability (correct WFP) / *Group size*	*0.6*	*0.65*	*0.7*	*0.75*	*0.8*	*0.85*	*0.9*	*0.95*
3	0.648	0.718	0.784	0.844	0.896	0.939	0.972	0.993
5	0.683	0.765	0.837	0.896	0.942	0.973	0.991	0.999
7	0.710	0.800	0.874	0.929	0.967	0.988	0.997	1.000
9	0.733	0.828	0.901	0.951	0.980	0.994	0.999	1.000
11	0.753	0.851	0.922	0.966	0.988	0.997	1.000	1.000

If a message is rejected because of poor alignment, the result is a similar binomial distribution, but using a smaller set of collaborating nodes. This reduces the probability of successful classification using node majority, by reducing the group size in Table 12 by a factor equal to the rejection probability.

If all nodes are required to participate in the aggregation algorithm (e.g. when the trustworthiness of participants is unknown), the probability of successful classification of

a message is very low. In this case, the probability of a correct WFP in Table 12 becomes the product of the probability of the message being accepted for classification multiplied by the probability of correct classification at a node. Both of these two probabilities must be greater than 0.5 to get an accurate aggregate classification decision with overall probability of greater than 0.5.

If we use the version of the Global algorithm that does not reject misaligned messages during classification, we observe an average degradation of 5% in classification accuracy for a transmission distance of 4m (see Table 5). The effect of a 5% level of degradation is to shift the location being used in Table 12 by one column to the left.

Table 13: Example- Classification Accuracy Using Decision Collaboration

Parameters / *Method*			*Global Method (Alignment Errors Tolerated)*	*Global Method (No Alignment Errors Tolerated)*
Sample Rejection Rate	***Group Size***	***Node WFP Classification Probability***		
0.45	***7***	***0.65***	71.8%	71%
0.45	***7***	***0.75***	84.4%	87.4%
0.45	***11***	***0.65***	76.5%	75.3%
0.45	***11***	***0.75***	89.6%	92.2%

For example, consider the cases shown in Table 13 of a network with either 7 or 11 nodes, all transmitting over medium range, with an acceptance rate of 45% for messages at each node and a probability of accurate classification at a node of either 0.65 or 0.75. Depending on the values chosen for the group size and WFP classification probability, the Global method that tolerates alignment errors is less or more accurate than the version which does not tolerate any alignment errors during classification. The selection of the

best variant depends on the specific network parameters. Similarly, if WFP degradation is lower than 5% or rejection rates are higher, this selection will also change.

Another alternative is to split messages up into sub-messages, provided that the increased messaging overhead can be tolerated in the network. The probability that a message with perfect alignment is received using such a scheme is also binomial and shown in Table 14, assuming that message segment acceptance decisions can be considered to be independent and random. For example, with the same assumptions, if a message is broken up into 7 segments and the probability of message acceptance on each trial is 45%, then the chance of an individual node accepting a message for classification is 98.5%, regardless of the variant of the Global algorithm which is used.

Table 14: Single Message Acceptance Probability Using Independent Message Segments

Probability (message acceptance) / ***Number of message segments***	***0.25***	***0.35***	***0.45***	***0.55***	***0.65***	***0.75***	***0.85***
3	0.578	0.725	0.834	0.909	0.957	0.984	0.997
5	0.763	0.884	0.950	0.982	0.995	0.999	1.000
7	0.867	0.951	0.985	0.996	0.999	1.000	1.000
9	0.920	0.980	1.000	1.000	1.000	1.000	1.000

Determining the degree of independence between WFP measurements made at different sites can be viewed as a type of inter-rater reliability problem. Such problems are common in the social sciences like medicine and psychology, where the objective is to measure the degree of agreement among raters for the same subjects taking the same tests. We use these statistical analysis techniques for our own work.

To determine independence, given that we have only two SDR 'raters', we measure the misclassification accuracy of two receivers for common messages and create contingency tables for the four different cases of correct/ incorrect classification of messages at each SDR receiver as shown in Table 15.

Table 15: WFP Misclassification Contingency Table

Cell #	***Event***	***Event Description***	***SDR 'AA' misclassifies***	***SDR 'BB' misclassifies***	***Joint Misclassification***
C0	{AA, BB}	Both SDRs classify correctly	No	No	No
C1	{AA, ***BB*** }	Only SDR 'AA' classifies properly	No	Yes	No
C2	{***AA***, BB}	Only SDR 'BB' classifies properly	Yes	No	No
C3	{***AA***, ***BB*** }	Both SDRs misclassify	Yes	Yes	Yes

We estimate the marginal probabilities of the events defined in the contingency table using the measured occurrence rates of each of the four different cases (cells C0, C1, C2 and C3 in the table). A message can only contribute, statistically, to one cell of the table. If misclassification events are independent, the product of the estimated marginal probabilities of misclassification by each of the SDRs will equal the estimate of the marginal probability of the joint misclassification event, which is the bottom cell on the right in Table 15.

8.4 Experimental Results- USRP1 Classification Decision Error Independence

This section describes the results of the experiments performed to measure the independence of errors at multiple receivers. We study the degree of independence for

two different receivers over medium-range and long-range transmission distances, using multiple trials. Short-range data is not presented in graphical form nor analyzed statistically, since very few classification errors are made by either SDR at this range. The medium-range statistics have misclassifications for all RF sources and the long-range statistics have the most misclassifications, making analysis worthwhile for these transmission distances.

In each trial, 100 training samples and 100 classification samples are randomly selected from specific disjoint intervals in time and a contingency table (Table 15) is created for the classification results for each message. We classify a set of five WSN nodes, as before, keeping the RF channels for the two SDR receivers as similar as possible to each other during the tests but arranging their antennas in a spatially diverse configuration with 26 cm separation.

We estimate the probabilities of misclassification events by measuring the observed relative frequencies of the cells in the contingency table for each message. For example, the probability that SDR 'AA' misclassifies a given message is estimated by calculating an average of the lighter-shaded cell frequencies within a trial. The estimate for SDR 'BB' is derived the same way with the darker-shaded cells. This averaging is done over all of the messages that are received from a distinct RF source in a trial and then also over all received messages.

The messages that are used for this averaging must be common to both SDRs (i.e. both SDRs must observe the message and also attempt to classify that message in a specific trial). To achieve this, we randomly select training and classification messages for the

first SDR from specific training and classification intervals. These selections are then duplicated on the second SDR.

On the second SDR, we verify that each message meets the same criteria for alignment mismatch errors in the WFP classification algorithm as used by the first SDR, excluding messages where this is not the case. The templates for each trial on the second SDR are also established using the same set of training samples as used by the first SDR, unless such samples are rejected by the training algorithm as being unfit. If a training sample is not acceptable, training sample replacements are selected randomly from the same training interval that was used by the first SDR.

The probabilities of the joint misclassification event and the product of the individual probabilities of miscalculation by each SDR is summarized in Table 16. The average value for all RF sources, over all trials, is also shown for both transmission distances.

Table 16: Marginal Misclassification Probabilities

Transmission Range	***WSN Source***	***Estimated marginal joint probability that both SDRs misclassify a message from an RF source***	***Products of estimated individual marginal misclassification probabilities for each message by each SDR***
Medium	***A***	0.0003	0
	B	0	0
	C	0.0264	0.0212
	D	0	0
	E	0.0007	0.0003
	Average	0.0055	0.0043
Long	***A***	0.0766	0.0754
	B	0.0090	0.0060
	C	0.0506	0.0596
	D	0	0.0033
	E	0.0151	0.0110
	Average	0.028	0.030

Examining the table, we see that the average probabilities for misclassification of specific RF sources are not the same at each SDR. However, for any given source, the product of the marginal probabilities of misclassification from each SDR is very similar to the joint probability of misclassification by both SDRs. This is true for both medium-range and long-range test results. Therefore, the two SDRs misclassify messages from any of our tested RF sources independently of each other, over both medium and long transmission distances.

8.5 Significance of SDR Classification Differences

We have seen that distinct RF sources are classified with different classification accuracies. We have also seen that different SDRs are better at classifying particular RF sources. In the previous section, we showed that misclassification decisions made by two receivers are independent of each other.

If the two receivers make independent errors, then this begs the question of whether independent WFP classification decisions made on a single SDR are significantly different from those expected by chance alone. If not, then WFP classification decisions made by different SDRs are not expected to be any different from those made using multiple messages for a specific RF source using the same SDR. In this case, the overhead of using network-level aggregation techniques for classification decisions from multiple receivers is not worthwhile.

If there is a significant difference, then it is worth considering whether the communication overhead and other implementation costs associated with aggregation of WFPs in a network are merited for a particular application. We show that there are statistically significant differences between the two SDRs when classifying messages

from the same RF source. The WFP templates themselves may also be aggregate-able, if different SDRs have perspectives which differ enough statistically.

8.5.1 Method- Simultaneously Classifying Messages With Two SDRs

We analyze the classification decisions for messages that are received simultaneously by two SDRs over the same RF transmission distance. Statistically, this data is categorical (non-ordinal and non-interval) and takes the form of matched samples, whose distributions may or may not be normal. In this case, McNemar's test is the most appropriate non-parametric[4] statistical test to determine whether classification decisions made by distinct SDRs differ from each other significantly or not (i.e. more than would be expected by chance alone) [**64**].

McNemar's test is frequently used for the analysis of paired data collected on the same subject taken at different times (i.e. before or after a specific treatment). McNemar's test is also often used to evaluate the differences between different testers of the same subject, which corresponds closely to our case. We wish to measure the similarity between the observations made by different SDR 'testers' on the same message 'subjects'.

The null hypothesis, in our statistical analysis, measures the likelihood that the classification decisions made by different SDRs have the same relative proportion of misclassifications as would be expected by chance alone. If we reject the null hypothesis, then the classification errors made by different SDRs are significantly different from each other in a statistical sense. In other words, it is possible for different SDRs to add value in aggregation schemes (either pooling classification decision or even comparing the WFP templates themselves), even when classifying the same received messages. If the

[4] Non-parametric statistical tests do not assume a particular distribution (e.g. like the normal distribution).

null hypothesis cannot be rejected, then collecting more samples at the same SDR is just as effective as using data collected by a different SDR and any aggregation overheads are not worthwhile.

However, McNemar's test has been criticized [**65**][**66**] as being inaccurate for cases where there is clustering between the matched observation pairs. For example, clustering might be expected in a dental study, where the analyzed treatment data includes the effects of a specific treatment on different teeth that come from the same patient mouth. Such clustering is a fundamental assumption in our work (i.e. we count on there being similarity between WFP samples from the same RF source). Therefore, we use the modified statistic proposed by Obuchowski in [**66**].

Referring to Table 15 for the meaning of the cell numbers, C1 and C2, in the contingency table, McNemar's basic test is calculated as a Chi squared random variable with one degree of freedom:

$$\chi_1^2 = \frac{(C1 - C2)^2}{(C1 + C2)}$$

In this statistic, only cases where a single SDR makes an error are included as being significant. Obuchowski's modified version of the statistic makes no assumptions about the correlation of measurements and performs more accurately when there is higher correlation between the statistical distributions of the paired clusters of data. This is the case with messages generated by the same RF source.

Defining p_i as the probability that SDR*i* makes a classification error on a randomly-selected message and p_i as an estimator for p_i, Obuchowski's modified statistic is defined as:

$$\chi_1^2 = \frac{\left(p_1 - p_2\right)^2}{\text{var}\left(p_1 - p_2\right)_{\overline{P}}}$$

where,

$$\text{var}(p_1 - p_2)_{\overline{P}} = \text{var}\left(p_1\right)_{\overline{P}} + \text{var}\left(p_2\right)_{\overline{P}} - 2\,\text{cov}\left(p_1, p_2\right)_{\overline{P}}$$
$$\overline{P} = \left(p_1 + p_2\right)\Big/2$$

The detailed steps for the calculation of this statistic are given in [**67**].

8.5.2 Statistical Analysis- Messages Classified by Two SDRs

The probability of the Obuchowski χ_1^2 statistic is calculated for each of the 100 trials and is shown in Figure 41. Values above the significance level of 0.05 indicate that there is evidence to reject the null hypothesis. In both the long-range and medium-range experiments, the probability of an invalid null hypothesis is always greater than 0.2. Therefore, there is strong evidence to support the claim of significant differences between the misclassification errors of the two SDRs.

Based on these results and our earlier results in Section 8.4, we conclude that WFP classification errors are independent and also significantly different at the two SDRs using our Global WFP algorithm. Therefore, we now consider methods for using WFPs in a network setting. For example, multiple SDRs can collaborate to establish a shared

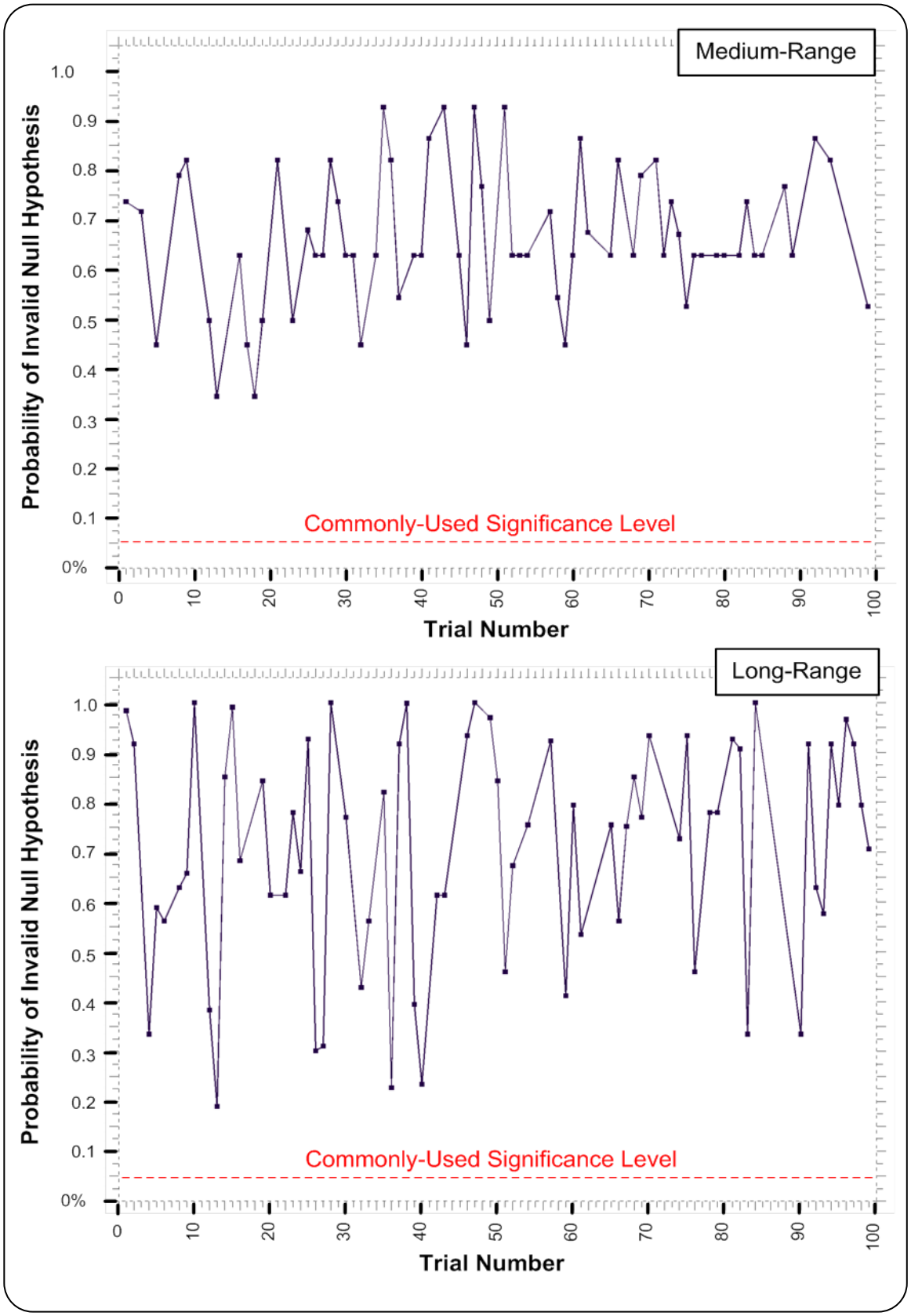


Figure 41: Obuchowski's Modified Statistic (Medium and Long Range)

secret communications channel by producing a shared secret key using WFPs for authentication and we present a method for doing this in the next section.

8.6 Establishing a Group Secret Key in the Presence of Malicious Nodes

We have determined a practical method to capture a WFP for an RF source. We have also shown that there is value in nodes collaborating to reconcile WFP classification decisions with each other or even to derive a better WFP template and that the sharing of templates from different nodes has advantages for improving classification accuracy. We now show how a WFP can be used in a network setting by modifying an existing protocol to establish a group key in the presence of a minority of malicious nodes.

8.6.1 Network Architecture

For fixed architecture networks with centralized and powerful base station nodes, nodes can communicate WFP information to those base stations, receiving authentication messages in return. They need only the ability to authenticate transmissions from the base station themselves, requiring multiple base stations to be available for the certification of other base stations (assuming that we do not allow such devices to certify themselves).

Different methods for key establishment, key distribution and key revocation can be used for authentication. However, if WFPs can be used to classify other nodes from inside a network, they are no longer required to be within transmission range of a central base station and distributed authentication options become possible. This reduces the power requirements in the network, since nodes no longer need to send/receive signals to/from a distant base station.

We assume that the transmission range is such that bidirectional communication is possible between those neighbours. We do not require full interconnection between all nodes, but assume that subsets of nodes are fully connected. Therefore, we end up with sub-networks of fully connected nodes. To use our method across an entire network, an honest majority of nodes are required at both ends of the RF links that form the gateway connections between these fully-connected sub-networks.

Our requirement for full interconnection can be relaxed, provided that malicious nodes are not in the majority in any of the sub-network partitions. Practically, defining all of these sub-cases is harder than just requiring the network to be fully connected. Basically, there cannot be a way for malicious nodes to corrupt an honest node without a majority of honest nodes overhearing the communication in both directions.

8.6.2 Protocol Incorporating WFPs

New protocols are required that incorporate WFPs into the authentication protocols as early as the neighbour discovery stage. WFPs work at the link layer for the set of neighbours around a node that are within transmission range, but can also be aggregated as templates. In Section 8.1 and Section 8.2, we showed that this is feasible. The protocol we select is based on previous work [**68**], which we term the 'base protocol'. The protocol allows nodes to establish a shared private group key, with the assumptions of Section 8.6.1.

In our version of the protocol, we add WFP functionality as an additional condition in the protocol. We require that the WFP for a given node is distinguishable from the other nodes in the sub-network by a majority of the nodes in that sub-network and also that

there is a comparable version of that WFP at each of those nodes. We have already shown that this is feasible in Section 8.1 and Section 8.2.

Our authentication method uses WFPs 'as far as they can go' and integrates them with other established security mechanisms when the WFPs are ambiguous, while still providing a secure authentication process. We accomplish this by binding the measured WFP physical layer parameters of the radio signal with the data being carried in that signal. We 'average out' RF channel effects by combining measurements from multiple locations in a secure fashion.

The 'readers' of a WFP are the immediate neighbours of the node in question. A connection of the WSN to other networks also relies on intermediate nodes to relay data to and from remote nodes in the network. Therefore, we need a reputation system and use trust mechanisms such as those listed in Section 2. We do not need to use auxiliary 'secure' authentication channels for this purpose.

In our version of the protocol [**6**], we add WFP functionality as an additional condition in the protocol and we assume that the transmission range is such that bidirectional communication is possible between those neighbours. However, we do not require full interconnection between all nodes. If we assume that these subsets of nodes are connected, we end up with sub-networks of fully connected nodes. For use of our method across an entire network, an honest majority of nodes are required at both ends of the RF links that form the gateway connections between these fully-connected sub-networks.

The objective of our method is to establish a shared and private security key amongst a specific group of nodes. We show how WFPs can be used to augment the operation of

the base protocol to include the authentication of WSN nodes. The basic steps for the initialization for authentication in the WSN sub-network using WFPs are as follows:

1. Nodes execute a neighbour discovery protocol.
2. Nodes create a secure private group of fully connected neighbours, with a shared conference key.
3. The group members exchange WFP information gathered during the previous two stages. The measurements may be aggregated. Nodes create a shared group credential for each other using this (potentially aggregated) data, provided that the WFP template data is close enough or that WFP classification decisions are close enough to their own WFP data/decisions[5].

After these steps have been completed, nodes have authenticated their immediate neighbours and have local certificates from them produced in the group setting. These local certificates can be grown into chains of certificates by 'showing' the digital credentials [**69**] (along with the authentication information for the referring nodes that issued them) to other neighbours which were not in the original fully-connected set. Credentials are verified interactively, but require the presence of one of the participants in the conference group originally used to establish the credential. 'Showing' requires the owner to demonstrate knowledge of all WFP data and all 'data layer' identifier data in the credential but does not require the value to be revealed.

While different group sizes are possible, impossibility results indicate that the group size must be three or more nodes to handle the presence of a single adversary. The private group is created by establishing a conference key. This means that adversaries need to

[5] The WFP template itself can be compared or the classification decision that is making use of that template can be compared with the local results.

participate actively in the group creation process to learn the WFPs of nodes at the end of different RF channels other than their own. This, in turn, requires them to 'reveal' their own WFP to everyone in the group. The steps listed earlier are now broken down in more detail in the following sections, to show more clearly how the existing protocol has been augmented with WFPs.

8.6.3 Neighbour Discovery

The originating node (which we will call '*A*') sends a request and an appropriate 'pseudo-corroboration' (e.g. a signature on a nonce, using the value of a as a secret key):

REQ: $ID = Z_A = g^a; a \in_R \mathbb{Z}_q$

This (and the WFP) shows that '*A*' is the actual sender. This is required for all transmissions and verifying it is required for all receptions. Therefore, it is implicitly included in the following steps, but omitted for reasons of brevity. Neighbours note the WFP of '*A*' as:

$WFP_{A \to i}$

They then reply in kind (using CSMA mechanisms to avoid collisions) using a message with the following format:

ACK: $ID = Z_i = g^i; i \in_R \mathbb{Z}_q$

Nodes record the WFPs of each other, recording also the corresponding cryptographic ID as the index. Once an adequate number of neighbours have responded with unique cryptographic keys, Node '*A*' starts the Conference Key Establishment protocol or aborts with an 'ABORT' message. We choose to permit the protocol to continue now to tolerate some 'noise', even if WFPs are not unique. A more energy-conserving approach would be to abort instead.

8.6.4 Conference Key Establishment

A' sends the list of group members, identifying and ordering them using their cryptographic Identifiers. This is an arbitrary ordering, but corresponds to the ring arrangement of the base protocol.

'*A*' calculates a partial key value derived from the published keys of the appropriate neighbours. These neighbours are determined using the cryptographic keys that lie above and below '*A*'s own value in the previously defined order[6]. This value is calculated as:

$$X_A = \left(Z_{i+1}/Z_{i-1}\right)^a,$$

where Z_{i+1} is the value immediately below '*A*'s value and Z_{i+1} is the value immediately above '*A*'s value in the ring-style ordering.

'*A*' then broadcasts the X_A value along with a hash of each of the WFPs that '*A*' measured in the Neighbour Discovery protocol round. The hash serves as a commitment to a WFP value, which can be revealed in the next round, but prevents an adversary from cheating using the WFP estimates from other nodes. The other nodes then record '*A*'s own WFP (and can compare it with their earlier estimate). They also note '*A*'s cryptographic identifier and partial key as well as his list of hashed WFPs (and associated tagging information).

Once this has been notes, the other nodes can then reply in kind with their own X_i values:

$$X_B = \left(Z_A/Z_C\right)^b$$

---

[6] An adversary can manipulate these values by waiting as long as possible (i.e. before 'A' times out and then replying with specific cryptographic values to put him in a specific ring position. We handle this problem using WFPs in the next stages.

They can include the hashes of the WFPs that they measured in their Neighbour Discovery round. As before, all nodes (including '*A*') record the WFPs for each of their neighbour's broadcasts. 'A' waits until all nodes have responded or a timeout period has elapsed. If the protocol has not aborted, '*A*' and all other nodes can then calculate the same shared group secret conference key using the previous broadcast results available to them.

8.6.5 WFP Exchange and Aggregation

Nodes in the secure conference group now encrypt further communications using their shared conference key. They each broadcast their Neighbour Discovery round WFP data. Receiving nodes can check to make sure each hashed value received in the Conference Key Establishment round is consistent. Nodes can record the WFPs of their neighbours in this round too. This gives them a total of three WFP measurements for each of their conference group neighbours. This is also a crude way for nodes to determine if their first WFP measurement is accurate.

The first-round WFP data is aggregated by '*A*' using an aggregation algorithm, as referenced to in Section 2. '*A*' compiles a credential with the resulting WFPs for each node in the conference group along with their data layer identifiers and the specified 'order' used to calculate the $X_i's$ used earlier. '*A*' then adds a threshold value and then signs the resulting credential, broadcasting this signed result.

Each node runs the same aggregation algorithm as 'A' and determines whether their WFP measurements are 'close enough' using the threshold value published by '*A*'. If all measurements are within tolerance, nodes sign 'A's (signed) broadcast in the order

specified in the credential. If not, they abort, suggesting a new threshold value (for which they could sign).

'A' can re-run the previous two steps until he gets a credential signed by all parties or he can terminate with the set of signatures that he has received. Thus, a *better* credential has more signing parties and a tighter tolerance value. To verify this credential, a node attempts to verify the signatures in the order specified in the credential. This credential would only be shared in a similar secure conference setting (i.e. the adversary has to participate in at least one session to learn WFP data).

The authentication credential with both FP and cryptographic identification information can now be used in the network. Credentials are verified interactively, but require the presence of one of the participants in the conference group originally used to establish the credential. They have the same form and are shown in the same way as the digital credentials in Section 2. Specifically, showing requires the owner to demonstrate knowledge of all WFP data and all 'data layer' identifier data in the credential but does not require the value to be revealed, necessarily.

8.6.6 Summary

Our modification of the base protocol integrates a physical authentication mechanism based on WFPs with existing digital credential technology. We do not rely on trusted third parties for this. The resulting protocol is efficient and has been shown to be secure, if the discrete logarithm problem is mathematically hard [**68**]. Our method can accept imperfect WFPs (i.e. cases where a single WFP alone is not adequate for distinguishing between nodes). This is done by aggregating different results (that were measured over different RF channels) from multiple nodes using resilient aggregation techniques.

Furthermore, we can also incorporate other physical authentication mechanisms as part of the credential for a node too.

Secure aggregation techniques such as those described in [**70**] would also allow our method to be extended across a network that was not fully connected. Similarly, trust-based reputation systems like those of [**71**] would work well with WFPs, allowing security to be bootstrapped into existence without requiring the provisioning of key information. Digital credentials [**69**] that include WFP information can be constructed in a shared group setting, as discussed in [**72**].

9 Chapter: Conclusions and Future Work

We summarize the main conclusions drawn from our work in Section 9.1 and then suggest areas for future work in Section 9.2.

9.1 Conclusions

Authentication of Wireless Sensor Network (WSN) nodes in a network is feasible using current technology. We present an authentication method using Wireless Fingerprints (WFPs) on both a real WSN node and a USRP1 software-defined radio.

On the WSN node, we use the characteristic responses of Automatic Gain Control (AGC) circuitry on a WSN node. The implementation is limited by the platform architecture. However, we successfully demonstrate the feasibility of discrimination between different RF sources. Basic discrimination between RF sources on real WSN node hardware is achieved near baseband data rates and on more than just the transient portion of the signal, without changing the WSN node design.

Better practical WFP classification performance is achieved on the more advanced software-defined radio USRP1 platform. We learned that more reliable access to received data samples is required than is provided on our SiLabs WSN node. Software-defined radios, like the USRP1, can provide that access and classify WFPs accurately. We achieved average classification accuracies of 99.6% at short range, 95.3% at medium range and 81.9% at long range for a sample of five SiLabs devices. While they provide full visibility of the demodulated data samples, and indeed all other (non-RF) radio functions, they use the same zero-IF architecture as many current WSN nodes. Therefore, the USRP1 is both a representative experimental platform for future WSN nodes and an acceptable stand-in for current WSN node technology.

WFPs are analog signals and so distance is an important variable, affecting performance. Short-range classification works well and is relatively immune to WSN node positioning. Medium-range classification is less dependable and the results are generally poorer than the short-range data we gathered. In turn, long-range classification is less consistent than medium-range classification, with much higher variability. Classification accuracy is directly related to the signal quality (i.e. Signal to Noise ratio).

Specific RF sources are classified more accurately by specific USRP1 devices. The relative classification performance of our two devices is consistent for a specific RF channel. However, the WFP classification accuracy of different RF sources changes for different RF Channel conditions. WFP classification performance generally deteriorates for all WSN nodes at USRP1 receiving devices, as the transmission distance increases, although certain nodes exhibit more variation than others and are more poorly classified. Small changes in node position cause changes in the number of error-free (valid WSN node code and valid sequence numbers) messages that are detected. The antenna alignment of the receivers is also significant. The positioning of the antennas affects the number of messages that are 'properly' received.

Under such conditions, certain WSN nodes dominate message reception over the transmission medium and also have better WFP classification performance. This effect is the most severe when SDR antennas are co-located. This might be related to problems in the experimental setup with deep fades caused by reflections or other small-scale variations in the RF environment.

Classification accuracy for specific SDR and node combinations using our WFP algorithm is stable over time. If a specific SDR (at a specific location) has issues with a

specific RF source, the identity of that 'problem' source and the classification accuracy performance for that RF source is also stable over time.

Our WFP algorithm is able to classify with good relative consistency over different RF transmission distances and performed well on multiple receiving devices. WFP templates generated using data collected over a short transmission distance can provide better classification accuracy, when used over long RF channels, than templates that are generated from data received over those long RF channels.

Our WFP is calculated over multiple chip positions in the IEEE 802.15.4 preamble. Each of our WFP templates contains average phase residual data for the phase residual values for all chip positions other than ones where the direction of the phase polarity is reversing. PCA techniques indicate that fewer than the 118 phase residuals used in our experiments are required for classification (see the next section).

WFPs can be used in an IEEE 802.15.4 WSN without requiring access to a centralized authority. We present a method using WFPs together with an existing protocol, to allow nodes to derive a shared secret group key. With this approach, key information does not have to be provisioned before deployment or using private locations, but can be derived using broadcast RF communication within a community of nodes.

We show that classification errors are independent with different receivers for different RF sources and that the WFP classification decision at different receivers is significantly different from multiple measurements made at the same receiver. This suggests that it is potentially worthwhile to use multiple WSN nodes to collaborate with each other either to produce more accurate WFPs and to make better classification decisions for a specific message.

As an example of this, we modify an existing protocol for secure establishment of a group key over the RF interface. We use WFPs to provide a method of linking the digital information being carried with the biometric information of the wireless signal used for carriage during the execution of the protocol. WFPs identify and authenticate the sender of the signal but do not authenticate the sender's information. Changes in the sender can be detected subsequently, since the WFP information is inextricably linked with the key information established during the execution of the protocol. This is important since key information is vulnerable in WSNs.

9.2 Future Work

We use a large number of chip coefficients to calculate the phase residuals for our WFP algorithm. However, we observe more variation in certain chip locations than others. Instead of using all chip locations, a subset of phase residuals in specific chip positions could be used. The specific positions that are used could be chosen based on a criterion for the signal to noise ratio, if specific chip positions have more phase margin or less variability for a given RF source.

Alternatively, the chip positions could be chosen adaptively to suit the RF channel conditions or to maximize discrimination between the set of known templates at any given time. PCA techniques, like the ones that we have used in our work, could be used to identify the most discriminating chip positions. The entries in the loading matrix to the distance calculation could be observed and analyzed further to determine if there was a consistent bias that magnified the contributions of certain chip positions.

At a network level, a trust-based system could be implemented on the WSN network to permit collaboration for nodes to generate better WFP templates or for nodes to improve

their WFP classification decisions. Within such a trust system, an aggregate WFP could be calculated as an average of individual training WFP templates. The template contributions from each participating receiver could be weighted appropriately by their range from the RF source, although establishing that range in a secure fashion is will be difficult.

A WFP-based intrusion detection or trust system (or a system based on other biometric types of authentication mechanism, rather than provisioned tokens) allows trust to be re-initialized after a system has been compromised by an attacker. Further research on attacks specific to WFPs would be useful as would a more detailed design of an interface with the WFP system and the intrusion detection system. Research into WFP aggregation methods which are resilient to attack would also be useful.

Appendices

Appendix A IEEE 802.15.4 Signal Format

The IEEE 802.15.4 specification [**47**] specifies the use of a data payload preceded by a set of header bytes in each PHY Physical Data Unit (PPDU) packet (Table 17). In our work, we have used the preamble sequence defined in the standard to synchronize to data. Because our work is at the physical layer, this is sufficient. In systems which do not have such a preamble defined, our methods and techniques for WFPs are still applicable but would require different methods for synchronization.

Table 17: IEEE 802.15.4 PHY PPDU Contents

Label	***SHR***		***PHR***		***PHD***
Number of Octets	4	1	1		variable
Description	Preamble	SFD	Frame length (7 bits)	Reserved (1 bit)	Physical Service Data Unit (PSDU)

In IEEE 802.15.4, the PPDU (data) octets are defined as follows (see Table 17):

- ***SHR***: This 5-octet field is broken down into two fixed-value sub-fields: 4 octets of *Preamble* and a single byte of *SFD* which have the following functions and form:
    - *Preamble*: This information allows a receiver to synchronize and lock onto the bit stream. The preamble consists of 8 identical '0000' symbols.
    - *SFD* (Start of Frame Delimiter): This sub-field signifies the end of the preamble and the start of the packet data. The SFD consists of a '1110' symbol followed by a '0101' symbol.
- ***PHysical Layer Header (PHR)***: contains information to specify the frame length (maximum value of 127 possible in 7 bits).

- ***PHysical layer Data (PHD)***: payload data which contains the Medium Access Control (MAC) sublayer frame. The length of the PHY payload has a maximum length of 127 octets, requiring 7 bits in the *PHR*.

IEEE 802.15.4 nodes transmitting at 2.4GHz transmit the OQPSK signal in the range of 2400 to 2483.5 MHz (16 distinct channels, spaced 5MHz apart). This portion of the RF spectrum is termed the Industrial/Scientific Medical (ISM) band. However, IEEE 802.11b/g/n routers, microwave ovens and various other equipment are also allowed to generate RF signals in the ISM band, making it a relatively noisy portion of the RF spectrum, world-wide, often containing unlicensed interfering RF sources.

The 2.4GHz OQPSK version of the IEEE 802.15.4 PHY encodes 4-bit data symbols into one of 16 different 32-chip pseudorandom Positive/Negative (PN) phase-shift sequences. The generated PN sequences used for each data symbol in the standard have a specific minimum hamming distance between each other and give the transmitted radio signal noise-like properties, spreading the signal energy over a wider transmission bandwidth. This makes the transmitted signal immune to a wider range of interference conditions present on the RF channel than if a narrower bandwidth were used. These interfering RF sources could also include other IEEE 802.15.4 radio signals.

The bytes below are transmitted over the air in order from left to right. The first transmitted bytes are the preamble bytes, which are used to synchronize the receiver. To align to data using the preamble, an IEEE 802.15.4 receiver synchronizes to the PN sequence that corresponds to the '0000' symbol, which is repeated eight times.

The '0000' symbol is mapped to the following 32-chip PN chip sequence:

11 01 10 01 11 00 00 11 01 01 00 10 00 10 11 10

If this pattern is split into two alternating sequences (shown in red and black) the resulting sub-sequences are:

'I' sub-sequence: 1 0 1 0 1 0 0 1 0 0 0 1 0 1 1 1

and

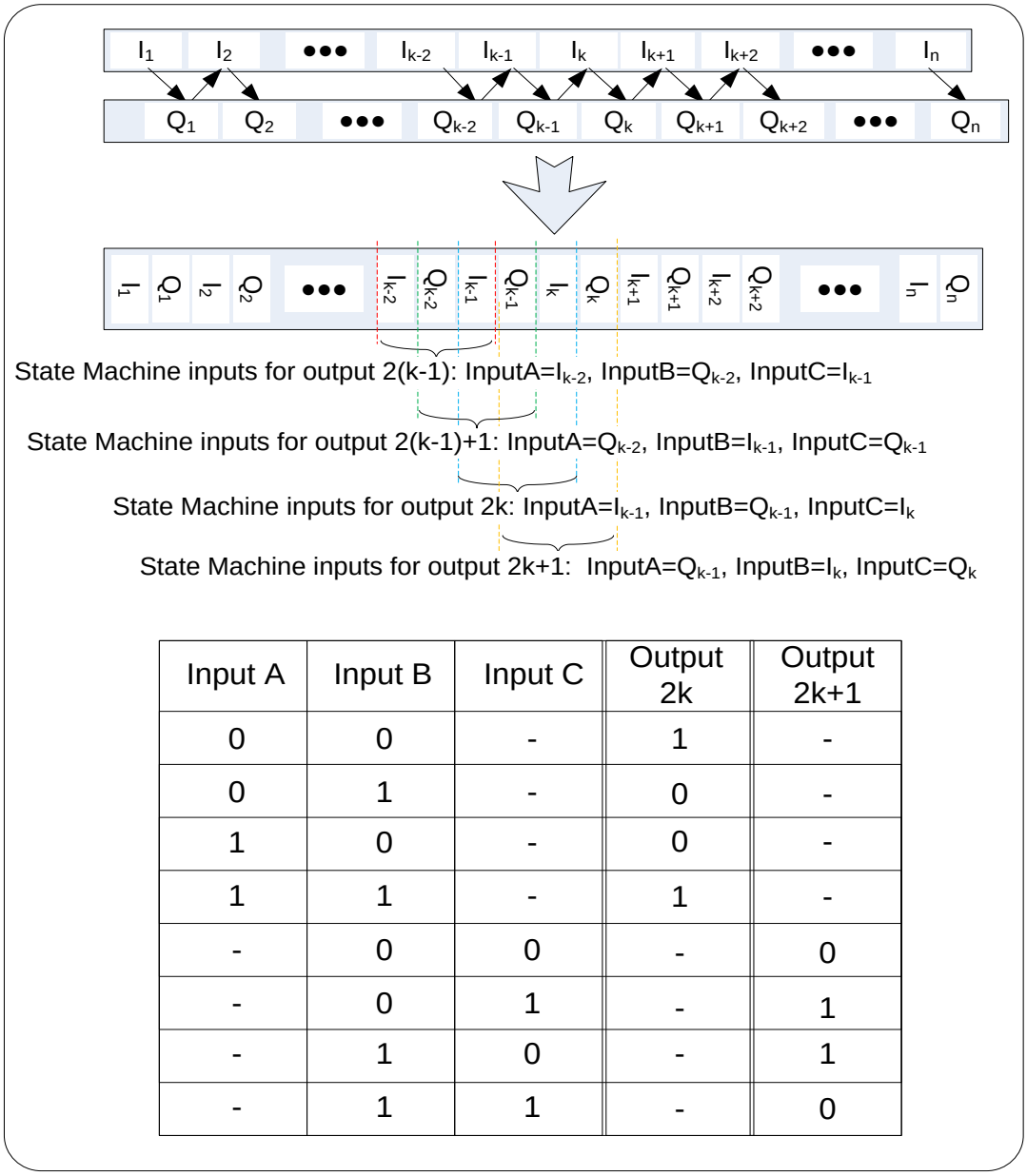
'Q' sub-sequence: 1 1 0 1 1 0 0 1 1 1 0 0 0 0 1 0

In the IEEE 802.15.4 OQPSK system, these two sub-sequences are transmitted in quadrature using orthogonal sinusoidal carriers and data is encoded as streams of positive and negative π/2 phase shifts. A 'zero' PN-sequence value is represented by a negative π/2 phase shift and a 'one' PN-sequence value is represented by a positive π/2 phase shift. At the receiver, these two orthogonal streams of I and Q data are recovered using a quadrature receiver and a state machine.

The conversion back to 'digital' 'zero' or 'one' PN sub-sequence values at the receiver uses a simple decision process, based on the polarity of the received phase shift. Any positive phase change is decoded as a 'one' and any negative phase change is decoded as a 'zero'. To reconstruct the final received data pattern, these I and Q PN digital sub-sequences are decoded using a decoding state machine.

The decoding state machine processes the two sub-sequences to produce a single output stream. The state machine outputs are completely determined by portions of the two most recent I and Q digital sample values. Using the nomenclature that I_k and Q_k are the k^{th} received I and Q digital values, respectively, the time-ordering of the I and Q sample inputs and the corresponding state machine output is illustrated in Figure 42.

During the sample period at time k, the state machine produces two output bits, labeled 2k and 2k+1, where k is the label of the input I or Q sample. Note that the $(k-1)^{th}$ I or Q



Input A	Input B	Input C	Output 2k	Output 2k+1
0	0	-	1	-
0	1	-	0	-
1	0	-	0	-
1	1	-	1	-
-	0	0	-	0
-	0	1	-	1
-	1	0	-	1
-	1	1	-	0

Figure 42: IEEE 802.15.4 Receiver State Machine Inputs and Outputs

sample value is required to produce the k^{th} output value. Therefore, if the bit that occurs just before the first preamble bit is not defined, this first preamble bit will not necessarily be decoded properly.

Using the state machine illustrated in Figure 42 and the sub-sequences given earlier, the state machine output for the preamble[7] '0000' symbols is:

```
11100000011101111010111001101100
```

For a perfect waveform, sampled at double the transmission rate, this corresponds to repeating every bit twice, which inflates the pattern to:

```
111111000000000000111111001111111100110011111100001111001111000 0
```

Our receiver code searches for this pattern (and the repetitions of it) to find the best alignment of data with the I/Q data samples that were received. Note that, with the exception of the very first preamble bit, this pattern should be seen 8 consecutive times. However, practical RF systems have gain control loops operating and sources of phase noise inside the receiver and over the RF channel, which can all create errors and differences from the expected pattern at the receiver.

[7] Since the previous phase value is not defined, the first '0000' preamble symbol state machine output is strictly: `X11000000111011110101110011011 00`, where X is an unknown value.

Appendix B Frequency and Phase Offset

0The IEEE 802.15.4 standard allows a slight offset of ±40 parts per million (ppm) in the frequencies of different transmitters or between the transmitter and the receiver. Such a small frequency offset only induces a small 'creep' in the relative phases of the two signals during the course of a message. Figure 43 shows a long-term measurement of the RF (transmitter and receiver) reference clock frequency for six different WSN nodes, measured in parallel, over a period of almost five days (413,579 seconds).

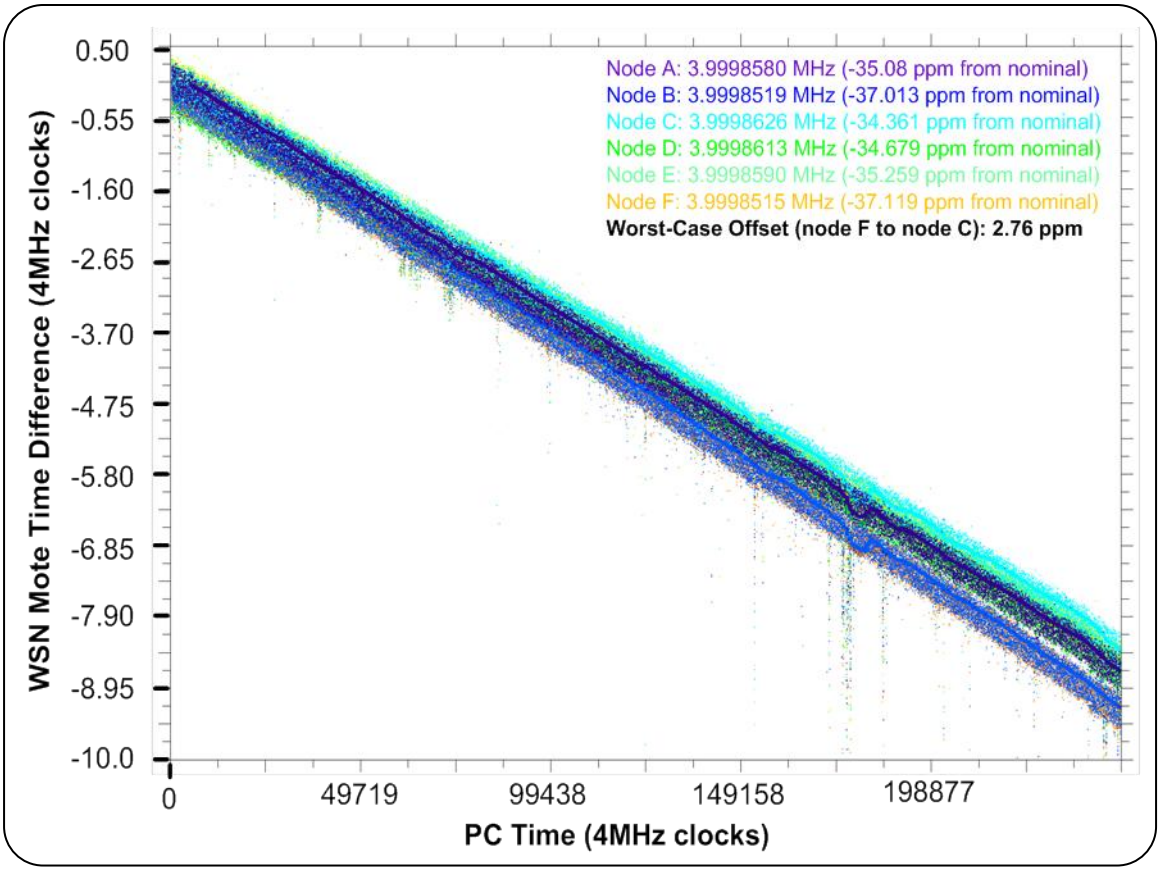


Figure 43: RF Clock Frequency Accuracy and Stability over Time (6 WSN Nodes)

A hardware timer/edge counter in the microprocessor on each WSN node was used to count the reference clock edges used for the RF circuitry. These counts were sent

periodically to a logging PC via the RS-232 interface on the node. The PC was connected to the Internet, which provided periodic synchronization adjustments from a Stratum-1 aligned clock source via the NTP (Network Timing Protocol) service.

The different RF reference clock rates can be seen to diverge over time, but the absolute frequency is only 10 ppm away from the ideal nominal value. The worst-case frequency difference between our six different WSN nodes is slightly over 3 ppm. We believe the periodic dips in the measured frequency are related to PC synchronization adjustments or errors over the RS-232 interface, since all six WSN nodes make the adjustments at the same time, but they are all running independently of one another.

While the clock frequencies are very close, the initial phase offset of the transmitter's clock is randomly distributed across the period of the receiver's clock. Multiple symbols must be received before accurate re-alignment of the receiver's clock is possible and the tuning algorithm is set up to occupy most of the preamble (i.e. 8 'zero' symbols). Our algorithm has been tested to work well even with just a single 'zero' symbol.

Symbol synchronization is required for both the WFP training process and the WFP classification processes. We need a method for aligning the training templates with each other and also to allow the newly-arrived signals in the classification set to be aligned with those templates. We now estimate an upper bound for the changes in the phase offset of the transmitter and receiver clocks during reception of the portion of the received IEEE 802.15.4 signal with known data signal content (i.e. the preamble zero symbols).

Since the clocks are not required to be synchronous in an IEEE 802.15.4 WSN, the initial receiver clock phase is random with respect to the transmitter clock phase unless the RF

source clock and Receiver clock are exactly equal (unlikely) Therefore, for an OQPSK demodulated digital I/Q signal where both I and Q signals are being sampled at a rate of 2Msamples per second, the initial clock phases could be off by as much as $\pm\pi/4$. However, the alignment will move slightly during the 8 zero symbols of the preamble.

The Tx and Rx carrier frequencies are not exactly the same. However, they are supposed to be within +/- 40 ppm of a perfect frequency according to the IEEE 802.15.4 specification. In a message of x bits (e.g. IEEE 802.15.4 preamble duration of 32 bits, the total number of clock phases gained/lost by the worst-case frequency difference would be:

$$\text{Number of chips in preamble} = 32\ (bits) \div 4 \left(\frac{bits}{symbol}\right) \times 32 \left(\frac{chip}{symbol}\right) = 256\,(chip)$$

$$\text{Chip rate} = 2\times10^{6}\left(chip/\text{sec}\right)$$

If sending and receiving oscillators are both at the extremeties of the ± 40 ppm specification,

$\Rightarrow$ Maximum possible Tx clock offset with respect to the Rx clock = 80 ppm

$$\Rightarrow \text{Clock delay difference per chip} = 80/1\times10^{6}\,(ppm) \div 2\times10^{6}\left(chip/\text{sec}\right) = 40\times10^{-12}\left(\text{sec}/chip\right)$$

$\therefore$ Maximum possible clock shift during the preamble sequence, T_{max}, is given by :

$$T_{max} = 256\,(chip) \times 40\times10^{-12}\left(\text{sec}/chip\right) = 10.24\times10^{-9}\,(sec) = 0.02048(chip)$$

Less than 2% of a chip period gain/slip occurs in the very worst case frequency offset conditions during the entire preamble. This is considered to be negligible movement of the receiver clock with respect to the transmit clock and is ignored in our algorithm. Note that our algorithm also works in less than 16 bits (one half) of the total preamble duration and that our actual WSN oscillator frequencies were only 3 ppm apart from each

other (or 1/26 of the worst case frequency offset of 80 ppm). This yields a gain/slip of approximately 0.04% of a chip period in the worst case.

Appendix C Comparison of USRP1 and WSN Node Architectures

For our purposes, the most significant difference between the RF interface of the two platforms is the degree of user access provided to the RF signals that are being transmitted and received by the radio. The physical-layer RF functionality of the WSN node is implemented in specialized hardware and firmware that is not modifiable by the user.

In contrast, the USRP1 SDR is designed specifically to allow a user to write custom software and also FPGA firmware to do as much of the RF signal processing as is practical. Further, a growing database of this software is also publicly available through the online Gnu Radio user community.

In addition, the FPGA hardware that is used at the front-end of the USRP1 can also be modified by the user to provide direct access to the data samples without the bottleneck of either a PC interface or PC CPU pre-processing. Indeed, we use a custom version of the FPGA firmware designed by CMU researchers [**57**] to provide high-resolution timestamp information on groups of I and Q demodulated data samples that are sent over the PC USB interface. We have not made any other modifications to the FPGA. As of the time of writing, timestamps have subsequently been incorporated into the latest Gnu Radio 'UHD' software but, unfortunately, they are not supported yet for the older USRP1 platform.

In spite of the differences between the two experimental platforms, both of them use a zero-IF RF receiver hardware architecture. The name arises from the absence of an intermediate frequency mixing stage that is common in many high-performance radio

architectures. This section discusses the basic architecture of a zero-IF radio receiver design that replaces the more ubiquitous heterodyne architecture (still) used traditionally in wireless receivers.

In a zero-IF architecture, the RF signal is converted as quickly as possible into a digital format. Digital Signal Processing (DSP) techniques implement functionality that used to require specialized and somewhat sensitive hardware components. Therefore, besides the reduction in hardware cost and complexity that is due to the loss of an intermediate mixing stage, the main advantages of the zero-IF architecture are: increased flexibility and consistency as well as ease of adjustment. These are all essential attributes for our work.

The received analog radio signal is detected initially as a wireless transmission by an antenna of the appropriate length for the specific RF carrier being used (e.g. 2.4GHz). The carrier information is subtracted from the RF input signal to produce a demodulated data signal with much lower frequency components (measured in MHz). Before this is possible, the weak signal from the antenna must be amplified. This first-stage of amplification has a critical effect on the noise performance of the system and, for this reason, user adjustment is limited.

To remove the RF carrier information, the output from this Low-Noise Amplifier (LNA) is multiplied with a locally-generated sinusoid of the same (nominal) frequency as the RF carrier used by the transmitter to send the signal to the receiver. In practice, the two carrier frequencies of the sender and receiver will not be exactly the same. The received signal and the local sinusoid are multiplied together using a high-frequency analog mixer component. After filtering out the undesired signal components that arise as a natural

result of this multiplication process, the lower-frequency demodulated data signal is recovered.

Both of our two experimental platforms use an initial low-noise amplifier (LNA) that has some limited low-granularity gain value settings, followed by a pair of mixers in an I/Q demodulator configuration. In this configuration, one (In-phase; I) mixer multiplies the received signal with the locally-generated oscillator signal. The other (Quadrature; Q) mixer multiplies the input signal with a quadrature version of the same local oscillator signal.

Both of the I and Q signal components are then amplified by a second-stage amplifier and converted into a digital format using an Analog to Digital (A/D) converter, sampling at a rate of 2Msamples per second. The A/D digital signal output is processed digitally to filter the undesired frequency components that result from the mixing process and compensate for distortions arising from the transmission of the signal over the RF channel.

The second amplifier stage has a much less critical effect on noise than the first-stage[8] LNA and is used to adjust the signal with much more precision than possible with the first stage. The optimum setting for the gain of both stages is controlled continuously in an automatic gain control (AGC) loop. The gain is set once at the beginning of the reception of each new message and remains constant during the reception of that message.

---

[8] The amount of noise introduced depends directly on the gain of the first stage of amplification, but noise introduced by subsequent stages is scaled with the product of all of the preceding gain stages (follows from Friis' formula for calculating the Noise Figure in a set of cascaded gain stages).

There is an inherent trade-off required when choosing the optimum gain for a given set of input signal conditions. The ideal gain setting is one that is small enough so that the input signal does not saturate the analog outputs of the components in the RF chain, up to and including the A/D input. If an output saturates, then distortion can occur since the output no longer tracks input changes properly. Conversely, while not saturating component outputs, a signal that has been too weakly amplified will be overwhelmed by small amounts of noise in the RF components prior to the A/D conversion.

The details of the AGC algorithm in the WSN node hardware are proprietary, so what we have deduced has been from what documentation is available in the public domain. It is believed that a firmware look-up table is used to implement the AGC control loop. The control outputs of this AGC algorithm were observed in real time on our experimental WSN node platform.

The USRP1 AGC algorithm is completely determined by user software. The AGC algorithm on the USRP1 is less critical for this phase of our research since the USRP1 provides full software access to the I and Q digital baseband data in real time. However, for both platforms, the digital demodulated I and Q signal components are the input to the AGC algorithm.

Bibliography

[1] (Last Accessed: 2010, Jan.) National Aging in Place Council. [Online]. http://www.ageinplace.org/

[2] E. D. Mynatt, A. .-S. Melenhorst, A. D. Fisk, and W. A. Rogers, "Aware Technologies for Aging in Place: Understanding User Needs and Attitudes," *IEEE Pervasive Computing*, vol. 3, no. 2, pp. 36-41, Jul. 2004.

[3] D. A. Knox and T. Kunz, "AGC-based RF Fingerprints in Wireless Sensor Networks for authentication," in *IEEE World of Wireless Mobile and Multimedia Networks (WoWMoM)*, Montréal, Canada, 14-17 June 2010 , pp. 1-6.

[4] D. A. Knox and T. Kunz, "RF Fingerprints for Secure Authentication in Single-Hop WSN," in *IEEE International Workshop on Security and Privacy in Wireless and Mobile Computing Networking and Communications*, Avignon, France, 2008, pp. 567-573.

[5] D. A. Knox and T. Kunz, "Practical RF Fingerprints for Wireless Sensor Network Authentication," in *8th International Wireless Communications and Mobile Computing Conference*, Limassol, CYPRUS, 2012, pp. 531-536.

[6] D. A. Knox and T. Kunz, "Secure Authentication in Wireless Sensor Networks Using RF Fingerprints," in *IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC '08)*, Shanghai, China, 17-20 December, 2008, pp. 230-237.

[7] (2012) National Aging in Place Council. [Online]. http://www.ageinplace.org/

[8] W. Mao, *Modern Cryptography Theory and Practice*, Fourth Edition ed. New Jersey, U.S.A.: Prentice Hall PTR, 2004.

[9] C. Boyd and A. Mathuria, *Protocols for Authentication and Key Establishment*. Berlin, Germany: Springer-Verlag, 2003.

[10] J. Hall, M. Barbeau, and E. Kranakis, "Radio Frequency Fingerprinting for Intrusion Detection in Wireless Networks," *IEEE Transactions on Dependable and Secure Computing*, vol. -, no. -, pp. 1-35, 2005.

[11] R. Nanda, S. Tiwari, and P. V. Krishna, "Secure and Efficient Key Management Scheme for Wireless Sensor Networks," in *IEEE Conference on Electronics Computer Technology (ICECT)*, Kanyakumari, India, 2011, pp. 58-61.

[12] M. Li, S. Yu, W. Lou, and K. Ren, "Group Device Pairing based Secure Sensor Association and Key Management for Body Area Networks," in *IEEE Conference on Information Communications (INFOCOM)*, San Diego, U.S.A., 2010, pp. 2651-2659.

[13] H. Chen and Y. Guo, "A Key Agreement Scheme Based on Bilinear Pairing for Wireless Sensor Network," in *IEEE International Conference on Dependable, Autonomic and Secure Computing (DASC)*, 2009, pp. 384-388.

[14] K. Ren, H. Su, and Q. Wang, "Secret Key Generation Exploiting Channel Characteristics in Wireless Communications," *IEEE Wireless Communications*, pp. 6-12, Apr. 2011.

[15] T. Choi, H. B. Acharya, and M. G. Gouda, "The Best Keying Protocol for Sensor Networks," in *IEEE International Symposium on a World of Wireless, Mobile and*

Multimedia Networks (WOWMOM), Lucca, Italy, 2011, pp. 1-6.

[16] F. Stajano and R. J. Anderson, "The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks," *Security Protocols*, vol. LNCS 1796, pp. 172-182, Apr. 1999.

[17] C. Kuo, M. Luk, R. Negi, and A. Perrig, "Message-in-a-Bottle: User-Friendly and Secure Key Deployment for Sensor Nodes," in *ACM International Conference on Embedded Networked Sensor Systems (SENSYS)*, Sydney, Australia, 2007, pp. 233-246.

[18] C. Chan and M. A. Jensen, "Secret Key Establishment Using Temporally and Spatially Correlated Wireless Channel Coefficients," *IEEE Transactions on Mobile Computing*, vol. 10, no. 2, pp. 205-215, Feb. 2011.

[19] K. Haataja and P. Toivanen, "Two Practical Man-In-The-Middle Attacks on Bluetooth Secure Simple Pairing and Countermeasures," *IEEE Transactions on Wireless Communications*, vol. 9, no. 1, pp. 384-392, 2010.

[20] N. Patwari and S. K. Kasera, "Robust Location Distinction Using Temporal Link Signatures," in *MobiCom'07*, Montréal, Canada, 9-14 September, 2007, pp. 111-122.

[21] B. Sieka, "Active Fingerprinting of 802.11 Devices by Timing Analysis," in *IEEE Consumer Communications and Networking Conference*, Las Vegas, 2006, pp. 15-19.

[22] Q. Qiu, T. Li, and J. Biswas, "Improving Sensor Network Security with Information Quality," in *ESAS 2005 (LNCS 3813)*, Visegrad, 2005, pp. 68-79.

[23] R. Mayrhofer and H. Gellersen, ""Shake Well Before Use": Authentication Based on Accelerometer," in *Pervasive 2007 (LNCS 4480)*, Toronto, 2007, pp. 144-161.

[24] A. Hossain, Y. Jin, W. Soh, and H. Van, "SSD: A Robust RF Location Fingerprint Addressing Mobile Devices' Heterogeneity," *IEEE Transactions on Mobile Computing*, vol. PP, no. 99, p. 1, Nov. 2011.

[25] B. Danev, D. Zanetti, and S. Capkun, "On Physical-layer Identificiation of Wireless Devices," *ACM Computing Surveys*, pp. 1-31, 2011.

[26] J. L. Wayman, "Biometrics in Identity Management Systems," *Security and Privacy Magazine*, vol. 6, no. 2, pp. 30-37, Apr. 2008.

[27] C. Vielhauer, "Fundamentals In User Authentication-Techniques for Binding Identities to Information," in *Biometric User Authentication for I.T. Security*. Berlin, Germany: Springer-Verlag, ISBN:978-0-387-26194-2 (print), 2005, ch. 4, pp. 77-115.

[28] Y. Sutcu, H. T. Sencar, and N. Memon, "A Secure Biometric Authentication Scheme Based on Robust Hashing," in *Proceedings of the 7th workshop on Multimedia and security (MM-SEC'05)*, New York, 2005, pp. 111-116.

[29] T. C. Clancy, N. Kiyavash, and D. J. Lin, "Secure Smartcard-Based Fingerprint Authentication," in *ACM Workshop on Biometric Methods and Applications, (WBMA'03),*, Berkeley, 2003, pp. 45-52.

[30] A. Juels and M. Sudan, "A Fuzzy Vault Scheme," in *IEEE International Symposium on Information Theory,*, Lausanne, 2002, pp. 408-421.

[31] Q. Li, Z. Li, and X. Niu, "Analysis and Problems on Fuzzy Vault Scheme," in *IEEE*

2006 International Conference on Intelligent Information Hiding and Multimedia, Washington, 2006, pp. 244-250.

[32] Ö. H. Tekbas, O. Üreten, and N. Serinken, "Improvement of Transmitter Identification system for low SNR Transients," *Electronics Letters*, vol. 40, no. 3, pp. 1-2, Feb. 2004.

[33] G. E. Suh and S. Devadas, "Physical Unclonable Functions for Device Authentication and Secret Key Generation," in *Design Automation Conference*, San Diego, 2007, pp. 9-14.

[34] J. Hall, M. Barbeau, and E. Kranakis, "Detection of Transient in Radio Frequency Fingerprinting using Signal Phase," in *Proceedings of the 3rd IASTED International Conference on Wireless and Optical Communications (WOC)*, Banff, 2003, pp. 13-18.

[35] P. C. A. Roberts, "Understanding Phase Noise in RF and Microwave Calibration Applications," in *NCSL International Workshop and Symposium*, Orlando, U.S.A., 2008, pp. 1-8.

[36] K. B. Rasmussen and S. Capkun, "Implications of Radio Fingerprinting on the Security of Sensor Networks," in *Security and Privacy in Communications Networks*, Nice, 2007, pp. 1-10.

[37] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in *Mobicom '08*, San Francisco, U.S.A., 2008, pp. 116-127.

[38] U. Rehman, S. Sowerby, and C. Coghill, "RF fingerprint extraction from the energy envelope of an instantaneous transient signal," in *Australian Communications*

Theory Workshop 2012, Wellington, New Zealand, 2012, pp. 90-95.

[39] W. C. Suski, M. A. Temple, M. J. Mendenhall, and R. F. Mills, "Using Spectral Fingerprints to Improve Wireless Network Security," in *IEEE Globecom*, New Orleans, 2008, pp. 2185-2189.

[40] W. E. Cobb, E. D. Lapse, R. O. Baldwin, M. A. Temple, and Y. C. Kim, "Intrinsic Physical Layer Authentication of Integrated Circuits," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 1, pp. 14-24, Feb. 2012.

[41] f. harris and G. Smith, "On the Design, Implementation and Performance of a Microprocessor-Controlled AGC System for a Digital Receiver," in *Milcomm '88*, San Diego, U.S.A., 1988, pp. 1-6.

[42] E. Miletic, M. Krstic, M. Piz, and M. Methfessel, "Digital Automatic Gain Control Integrated on WLAN Platform," *Proceedings of World Academy of Science, Engineering and Technology (PWASET)*, vol. 31, pp. 572-576, Jul. 2008.

[43] Ettus Research. (Last Accessed: 2007, Dec.) Ettus Research USRP1 Software-Defined Radio. [Online]. http://www.ettus.com/downloads/ettus_ds_usrp_v7.pdf

[44] H. Arslan, "Special Issue (Editorial): Cognitive radio, software-defined radio and adaptive wireless systems," *Wireless Communications and Mobile Computing*, vol. 7, pp. 1033-1035, May 2007.

[45] O. Ureten and N. Serinken, "Wireless Security through RF Fingerprinting," *Canadian Journal of Electrical and Computing Engineering*, vol. 32, no. 1, pp. 27-33, 2007.

[46] M. D. Williams, M. A. Temple, and D. R. Reising, "Augmenting Bit-Level Network

Security Using Physical Layer RF-DNA Fingerprinting," in *IEEE Global Telecommunications Conference (IEEE Globecom)*, Miami, 2010, pp. 1-6.

[47] IEEE Computer Society, "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)," IEEE Computer Society Standard IEEE 802.15.4-2006, Sep. 2006.

[48] K. H. Mueller and M. S. Muller, "Timing Recovery in Digital Synchronous Data Receivers," *IEEE Transactions on Communications*, vol. COM-24, no. 5, pp. 516-531, May 1976.

[49] F. M. Gardner, "A BPSK/QPSK Timing Error Detector for Sampling receivers," *IEEE Transactions on Communications*, vol. COM-34, no. 5, pp. 423-429, May 1986.

[50] D. Zanetti, B. Danev, and S. Capkun, "On Physical-layer Identification of Wireless Devices," *ACM Surveys*, 2011.

[51] Texas Instruments Semiconductor. (Last Accessed: 2007, Mar.) Texas Instruments CC2420 RF IC. [Online]. http://www.ti.com/product/cc2420

[52] S. Haykin and M. Moher, *Modern Wirelss Communications*. Upper Saddle River, New Jersey, U.S.A.: Pearson Prentice Hall, 2005.

[53] Community, GNU radio User. (Last Accessed: 2012, Jan.) GNU radio. [Online]. http://www.gnuradio.org

[54] Ettus Research. (Last Accessed: 2012, Feb.) Ettus Products. [Online]. https://www.ettus.com/product

[55] Ettus Research. (Last Accessed: 2011, Dec.) RFX2400 Daughter Card. [Online].

https://www.ettus.com/product/details/RFX2400

[56] T. Schmid and G. Nychis. (Last Accessed: 2011, Jan.) The Comprehensive GNU Radio Archive Network UCLA Zigbee Phy. [Online]. https://www.cgran.org/wiki/UCLAZigBee

[57] G. Nychis. (Last Accessed: 2009, Nov.) Comprehensive GNU Radio Archive Network. [Online]. https://www.cgran.org/wiki/CMUmacs

[58] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless Device Identification with Radiometric Signatures," in *MobiCom'08*, San Francisco, California, USA., September 14–19, 2008, pp. 116-127.

[59] Robert Gentleman, Ross Ihaka and Free Software Foundation contributors. (Last Accessed: 2012, Nov.) The R Project for Statistical Computing. [Online]. http://www.r-project.org/

[60] Michels, Helmut. (Last Accessed: 2011, Jun.) Max Planck Institute for Solar System Research - Dislin home page. [Online]. http://www.mps.mpg.de/dislin/

[61] A. Sharma and K. K. Paliwal, "Fast Principal Component Analysis using Fixed-Point Algorithm," *Pattern Recognition Letters*, vol. 28, pp. 1151-1155, 2007.

[62] H. M. Jones, A. Saha, and T. D. Abhayapala, "The effect of finite antenna separation on the performance of spatial diversity receivers," in *Seventh International Symposium on Signal Processing and Its Applications*, vol. 2, Paris, France, 2003, pp. 515-518.

[63] M. Sewell, "Ensemble Learning," University College of London Department of Computer Science, Research Report - Chronological Literature Review RN/11/02,

2011.

[64] J. D. Leeper. (Last Accessed: 2012, Oct.) Choosing the Correct Statistical Test. [Online]. http://bama.ua.edu/~jleeper/627/choosestat.html

[65] J. N. K. Rao and A. J. Scott, "A Simple Method for the Analysis of Clustered Binary Data," *Biometrics*, vol. 48, pp. 577-585, 1992.

[66] N. A. Obuchowski, "On the Comparison of Correlated Proportions for Clustered Data," *Statistics in Medicine*, vol. 17, no. 13, pp. 1495-1507, Dec. 1998.

[67] W. F. McCarthy. (Last Accessed: 2012, Aug.) Adjustment to the McNemar's Test for the Analysis of Clustered Matched-Pair Data. Collection of Biostatistics Research Archive - COBRA Preprint Series. [Online]. http://biostats.bepress.com/cobra/art29/

[68] M. Burmester and Y. Desmedt, "A Secure and Efficient Conference Key Distribution System," *Advances in Cryptology- Eurocrypt '94*, vol. Springer Verlag LNCS, no. 950, pp. 275-286, 1995.

[69] S. Brands, *Rethinking Public Key Infrastructures and Digital Certificates*, 2nd ed. Cambridge, Massachusetts, U.S.A: MIT Press, 2001.

[70] L. Buttyan, P. Schaffer, and I. Vajda, "RANBAR: RANSAC-Based Resilient Aggregation in Sensor Networks," in *SASN '06*, Alexandria, U.S.A., 2006, pp. 83-90.

[71] S. Capkun, L. Buttyan, and J. P. Hubaux, "Self-Organized Public-Key Management for Mobile Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 52-64, Jan. 2003.

[72] M. Just and S. Vaudenay, "Authenticated Multi-Party Agreement," in *ASIACRYPT - International Conference on the Theory and Application of Cryptography*, Kyongju, Korea, 1996, pp. 36-49.