Beamforming Software Defined Radios for Wireless Sensor Network Interference Evaluation

William Zhao

May 29, 2020

Beamforming Software Defined Radios for Wireless Sensor Network Interference Evaluation

by

William J. Zhao

A thesis submitted in partial satisfaction of the

requirements for the degree of

Master of Science

in

Electrical Engineering and Computer Science

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor David E. Culler, Chair
Professor Prabal Dutta

Spring 2020

Beamforming Software Defined Radios for Wireless Sensor Network Interference Evaluation

by William J. Zhao

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in partial satisfaction of the requirements for the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee:

Professor David E. Culler
Research Advisor

5/26/2020

(Date)

* * * * * * *

Prabal K. Dutta

Professor Prabal Dutta
Second Reader

5/29/2020

(Date)

Beamforming Software Defined Radios for Wireless Sensor Network Interference Evaluation

Abstract

Beamforming Software Defined Radios for Wireless Sensor Network Interference Evaluation

by

William J. Zhao

Master of Science in Electrical Engineering and Computer Science

University of California, Berkeley

Professor David E. Culler, Chair

Wireless communication on the 2.4GHz ISM band is becoming ever more prevalent in buildings. Additionally, various physical layer technologies share this RF spectrum (e.g IEEE 802.11 WiFi, Bluetooth, IEEE 802.15.4). As device density increases, devices sharing different physical layer technologies will be forced to transmit on the similar frequencies as each other, causing interference with each other's transmissions. Low power wireless sensor networks (WSN) utilizing IEEE 802.15.4 transmissions are particularly impacted by comparatively high powered WiFi emissions. As a result, previous research work has been done on network protocols to allow WSNs to mitigate the effect of cross-technology interference (CTI) produced from sources such as WiFi. However, the evaluation for these protocols differ protocol to protocol including the method of WiFi interference generation. Many factors that affect the CTI observed by the WSN such as testbed layout, testbed container shape, CTI location, and antenna parameters makes it difficult to precisely adjust the levels of CTI on a WSN.

In order to provide a dynamic CTI generation platform that can be tuned to impact different devices within a WSN with different levels of interference, we develop a software defined radio (SDR) solution. We propose Software Defined Interference Generation (SDIG) as a CTI generation method to induce WiFi CTI on motes instead of using real WiFi transceivers. Notably SDIG uses an SDR array which allows beamforming techniques to steer the interference signal to impact different devices with varying magnitudes of interference.

We evaluate SDIG in comparison to real and arbitrarily placed CTI transmitters. We demonstrate the system's ability to target individual nodes and deliver precise amounts of interference with minimal effects on other devices. However, we also reveal the scaling limitations of using SDIG to simultaneously impact many devices, each with varying amounts of interference. We recommend The use of beamforming in SDIG provides testbeds a way to carefully induce a level of CTI to a specific device within the testbed.

Contents

List of Figures

List of Tables

Acknowledgments

I would like to thank the researchers in the BETS (Buildings, Energy, and Transportation Systems) research group for their contributions to the WSN software/hardware stack that my research heavily utilized. I especially want to thank Hyung-Sin Kim for his excellent mentorship and guidance through my entire research experience at BETS over the past few years which includes the work done for this thesis. I would also like to thank Professor David Culler for introducing this interesting problem space to me as well as providing critical insight and advice for this project. I would also like to thank Professor Prabal Dutta for being a faculty reviewer for this paper. Lastly, I want to acknowledge my family and peers who have provided knowledge and support these past years.

Chapter 1

Introduction

Starting from over a decade ago, we have noticed an incredible increase in the number of wireless devices deployed [15]. Due to the limited bandwidth of the 2.4GHz unlicensed ISM band, devices sporting IEEE 802.11 (WiFi), Bluetooth, IEEE 802.15.4, and other forms of communication, all compete for connectivity in this band. Additionally, these devices communicate with varying levels of emission power and bandwidth. Due to the different physical properties of the transmissions, it's not guaranteed that the radios across different transmission protocols can understand each other. This is particularly relevant for low power wireless communication devices (which are used in the Internet of Things (IOT), home automation, and smart buildings) as such devices are constrained to be energy efficient. These devices cannot have high power transmitters and require a clear and reliable channel to transmit efficiently [9]. However, due to increase in the density of wireless devices, low power devices deployed in wireless sensor networks (WSN) are heavily impacted by other wireless transmitters due to frequency exhaustion. Transmissions from higher power devices such as WiFi overshadow the transmit strength of devices in WSNs (motes). Performance of WSNs are impacted so much from cross technology interference (CTI) that research work and network stacks for WSNs to mitigate CTI has been developed. Some mitigation strategies include having motes dodge the interference entirely while others involve the motes accurately predicting intervals in which it's probable for a transmission to succeed in a crowded channel [8, 11, 12, 21].

When comparing the performance of CTI mitigation protocols, packet reception ratio (PRR) is one of the main metrics of determining the success of the protocol. By limiting the amount of re-transmissions a mote has to make before a packet is proliferated through the network, the power efficiency of the mote increases. One of the main parameters tuned to evaluate PRR in an environment with interference generators is the distance the motes are from the generator. As the motes are placed further from the source of interference, the received "signal" becomes weaker due to propagation. A valid method of emulating the effect of an interference source being further way is to transmit using a lower gain in the first place [9].

Despite this degree of freedom for evaluation through transmit gain adjustment, it's still difficult to sculpt motes to receive particular levels of attenuation just by adjusting the transmit gain. Suppose we have a system with one interference generation source, one omni-directional antenna, and two motes that are placed equidistant to the omnidirectional antenna on opposite sides. Barring multipath effects, it would be impossible for the omni-directional antenna to induce different levels of observed interference on the motes.

This problem can be addressed through the use of a multi-antenna array. By constructively and destructively combining a target signal in a direction through spaced apart antennas, known as beamforming, a transmission array can impose different levels of interference on motes that are equidistant from the array but have different bearings. In order to accurately beamform a signal, tight synchronization between the signals transmitted across the antennas is necessary. This is non-trivial with off the shelf wireless transceivers. However, in the past decade, an increasingly popular technology, known as software defined radios, have been shown to achieve synchronization to allow for beamforming without too much trouble.

Software defined radios (SDRs) with transmit capabilities can generally take in-phase and quadrature samples from a machine, upshift them to a target carrier frequency, and transmit the results over the air. Alternatively, SDRs can be used to receive samples at a target carrier frequency, and convert the results to IQ (in-phase, quadrature) samples to be processed in software. This is incredibly powerful as software toolchains such as GNU Radio can be used to rapidly prototype routines to send and receive packets [4]. Additionally, these radios can be synchronized, making them a perfect candidate to perform beamforming.

In order to be able to better evaluate WSN performance with varying network stacks, we suggest an architecture known as Software Defined Interference Generation (SDIG) as a way to tune and emulate the WiFi interference experienced by motes in an arbitrary testbed environment.

Chapter 2

Related Works

SDRs have been utilized to emulate the radio stack for a multitude of wireless protocols including WiFi and IEEE 802.15.4 for both transmit and receive purposes. GNU Radio is also shown to be a staple tool used in the SDR community for many research and industrial applications [21, 10, 11, 14].

SDRs have been shown to be particularly useful in testbed deployments as a way to verify the correctness of other hardware receivers [14]. Additionally, they have even been used as the means of executing CTI mitigation strategies due to the ability to quickly prototype the entire protocol in software and arbitrarily deploy it to radios [9]. Most relevantly, SDRs have been used as a useful means of generating CTI in testbeds themselves. In these experiments, CTI is produced using radio stacks such as WiFi in order to evaluate CTI mitigation protocols where signal demodulation is important [10].

Beamforming has also been successfully used in SDR applications as well. However, in these cases, the beamformed signal was designed to be received and demodulated to mitigate interference rather than impart interference on a device [13].

We expand on the work done on using SDRs as a CTI source by utilizing beamforming as a way to induce varying levels of interference on motes in a testbed without being constrained to the testbed topology and environment. No currently known work explicitly uses beamforming as a way to tune the levels of CTI that motes receive in a deployment.

Chapter 3

System Architecture

We propose SDIG (Software Defined Interference Generation) as the interference generation system to emulate WiFi interference for motes.

Deploying SDIG in any capacity requires the use of multiple calibration steps with feedback from the motes in the testbed. The main parameters that need to be obtained are the mote bearing with respect to the SDR array as well as the correct gain to transmit to the radio.

The steps involved to deploy SDIG are as follows:

1. Acquire calibration coefficients

   a) Acquire the transmit direction value for each mote.

   b) Acquire the transmission gain values required for the SDR array to match the observed interference from the true WiFi source at each mote.

2. Transmit a synthetic beamformed WiFi pattern, repeatedly switching the steering direction and transmit gain of the array in time to impact each mote.

3.1 Testbed Topology

Our testbed consists of the SDR array, the radio array controller machine, wireless motes, ethernet-networked machines with debuggers attached to the motes, and the interference generator. The networked machines are also part of the same wired network as the array controller machine. These do not necessarily have to be part of the same network as the sample interference source, but this is the case in our testbed. Figures 3.1 and 3.2 show an abstract and physical representation of our sample testbed.

During some tests, various objects were introduced to introduce multipath effects into the system. Motes were also moved around/introduced to the system. The arrays were placed approximately 0.25 meters apart from each other which is approximately the distance of 2 wavelengths at the transmit carrier frequency.

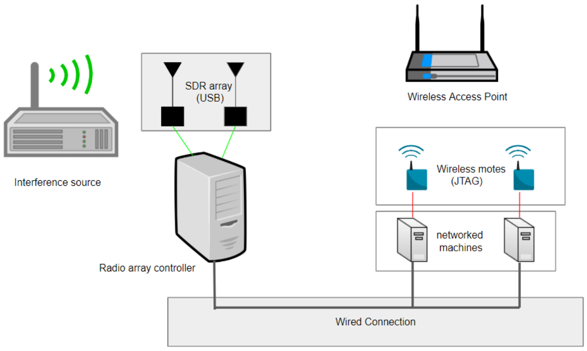


Figure 3.1: An abstract view of the network topology of the testbed. The main array controller has USB connections to each of the radios in the array. The ethernet-networked machines have JTAG connections to the motes to read debug information from the motes and to flash binaries to the motes.

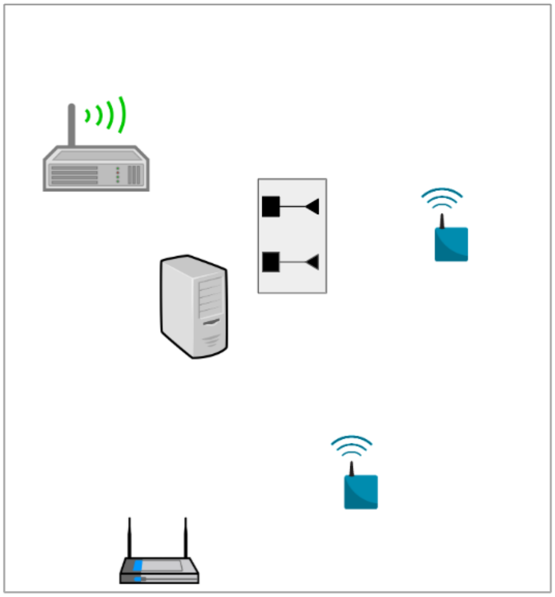


Figure 3.2: An example of a top-down view of the testbed with 2 motes. Additionally, the motes and radios are all approximately on the same elevation. Various pieces of furniture are also present in this.

3.2 Software/Hardware Stack

For our SDR array, we used 2 HackRF devices as our SDR choice [7]. These were affixed to a table. In the naive case, even if we transmit on both HackRFs, the transmitted samples can be offset by thousands of samples which would not be suitable for beamforming. To mitigate this drift, we had our first HackRF be a master device and the second HackRF be a slave device. The slave device utilizes the clock from the master. We also made sure that each HackRF shared a common ground. This synchronization procedure has been previously conducted and is documented in the HackRF wiki [16]. Each SDR was connected a desktop machine via USB which could run the radio flowgraphs to perform calibration and generate interference.

Our testbed mainly used two Hamilton motes running RIOT-OS with the GNRC network stack [1, 18]. TelosB motes running Contiki-NG with the RPL network stack were also used for an experiment [3]. When measuring the energy readings in the air, the Hamilton mote's radio has a register which outputs the "energy level" at a channel. This value evaluates the average energy level over 8 symbol periods. For the TelosB motes, the RSSI value is taken from the radio. A laptop and Raspberry Pi were responsible for reading the debug information from the motes as well as forwarding the data to the desktop machine [17].

For both the Hamilton mote and the TelosB mote, two binaries were developed: one that repeatedly yields the energy reading/RSSI on a channel and udp client/server binary in which the client mote periodically sends packets carrying a sequence number to the receiver.

A WiFi dongle was used to generate interference. UDP packets were sent over the air through an access point to a wired device on the network.

To control the SDR array, we used GNU Radio, a signal processing toolkit which allows the use and development of signal processing blocks to control signals. We heavily utilized an open-source 802.11 module which allows the generation of WiFi packets [5].

3.3 Mote Bearing Coefficient Acquisition

For deployments with greater than 1 transmitter, beamforming a signal to a mote requires the mote's bearing in order to steer the signal in the right direction. It should be noted that although traditional beamforming systems are complex and can adjust the transmit weights dynamically, our system's motes have coarse granularity and are static throughout the calibration and test process. Thus, our bearing detection algorithm for our 2 linear transmitter system involves sweeping the bearing space. For a given desired angle to beam towards, we can phase shift the signal for one of our transmitters by the following expression:

$$e^{jm(2\frac{\pi d}{\lambda}cos(\phi))} \tag{3.1}$$

In our implementation, m is the transmitter number, d is the distance in meters between transmitters, λ is the wavelength of the transmission frequency, and ϕ is defined by the angle to steer the signal.

In our system, having only 2 transmitters is not sufficient to completely disambiguate the steering direction in 360 degrees. A signal steered ϕ degrees is also steered $-\phi$ degrees as the cosine in equation 3.1 would evaluate to the same value as well as because the antennas are omni-directional. This is illustrated in Figure 3.3. The steering is done in software by multiplying a sample stream by the phase offset as shown in Figure 3.4.

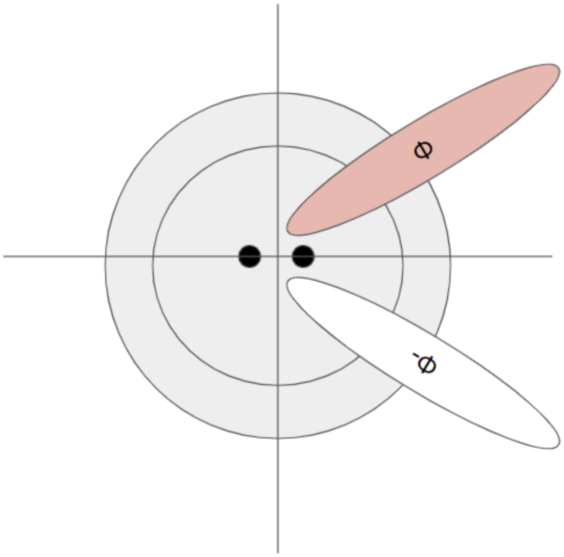


Figure 3.3: Top down view of transmitter array steering a signal by ϕ degrees. Steering a signal to an angle ϕ will also result in a steered signal to $-\phi$ when only 2 transmitters are used.

Thus, we only need to consider the angles between 0 and π radians. We first flash a binary to the mote under calibration which allows it to report its energy reading/RSSI values over the debug interface. We obtain the steering direction for each mote by arbitrarily steering the high rate synthetic WiFi transmission in different directions, recording the observed energy reading from the mote, and then taking the argmax of the result. An example of tested directions can be found in Figure 3.5. The gain used in both transmitters are the same and is set high enough to register above the noise floor on the motes. However, because of the limited resolution from mote (the measurements are in dbm), the dynamic range of the sweep is rather limited. Nevertheless, this strategy does yield the best bearing to the nearest dbm from evaluation. An example result of a frequency sweep is illustrated in Figure 4.6.

We obtain the best bearing for each mote in the test using this method.

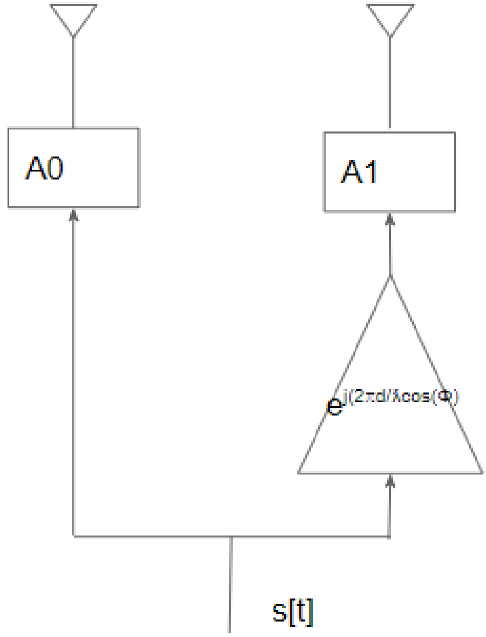


Figure 3.4: Phase driven beamforming array diagram. The beam is steered by phase shifting antenna 1. This phase shift is done in software by multiplying the sample stream by the phase shift constant before it is sent to the radio.

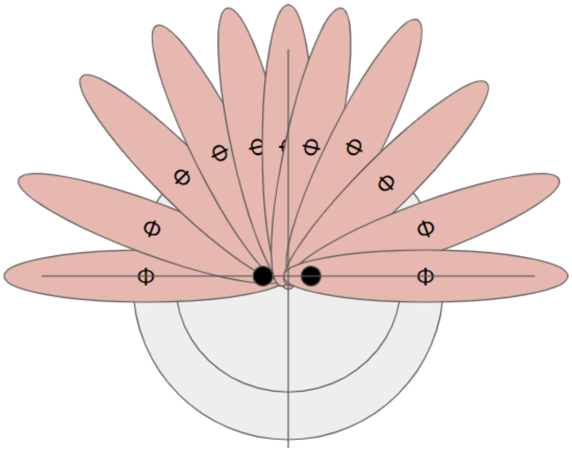


Figure 3.5: A top-down representation of the bearing sweep performed by the radio.

3.4 Gain Coefficient Acquisition

In order to evaluate the transmission (tx) gain that the SDR array should be using to beamform to the mote, it's important to see how the mote reacts to the original interference source. The mote is first flashed with the binary which allows energy readings to be output to the debug interface. The mote is then subjected to a high data rate transfer from the WiFi interference source. This is to ensure that a large percentage of the energy reading samples are taken when the transmitter is transmitting. Once the interference source is transmitting, we start sampling the mote's energy readings. After sufficient samples are collected, the median reading is taken to get the target energy reading to achieve with the SDR array.

If we are using more than 1 transmitter, we first load the bearing derived earlier into our synthetic WiFi transmission program . We then start transmitting a high data rate synthetic WiFi interference to that specific mote with an arbitrary gain that is higher than the noise floor for the calibrated mote. A service on the networked machines that are connected to the motes then reads the energy readings and serve them to any requester. Meanwhile, a logic block running on the SDR array controller requests samples from the networked machines attached to the mote. An averaging filter, designed to reduce the effect of noise on the control feedback system, consumes the samples. The difference between the target energy reading and the observed energy reading is then scaled by a gain constant (K_i) and added current TX gain to generate the new TX gain. This continues until the target energy reading is met on the mote or the SDR array saturates. Note that since this is a pure integral controller, the gain will likely never converge completely, but will slightly oscillate around the correct TX gain value. The feedback system is terminated whenever the energy readings have settled around the target. The TX gain value is then saved. Essentially, this system is just an auto gain control system which seeks to find the the right transmit gain to achieve the same energy reading that the real interference source achieved on the mote. A diagram of the feedback control system can be shown in Figure 3.6.

We obtain the best SDR TX gain for each mote in the test using this method.

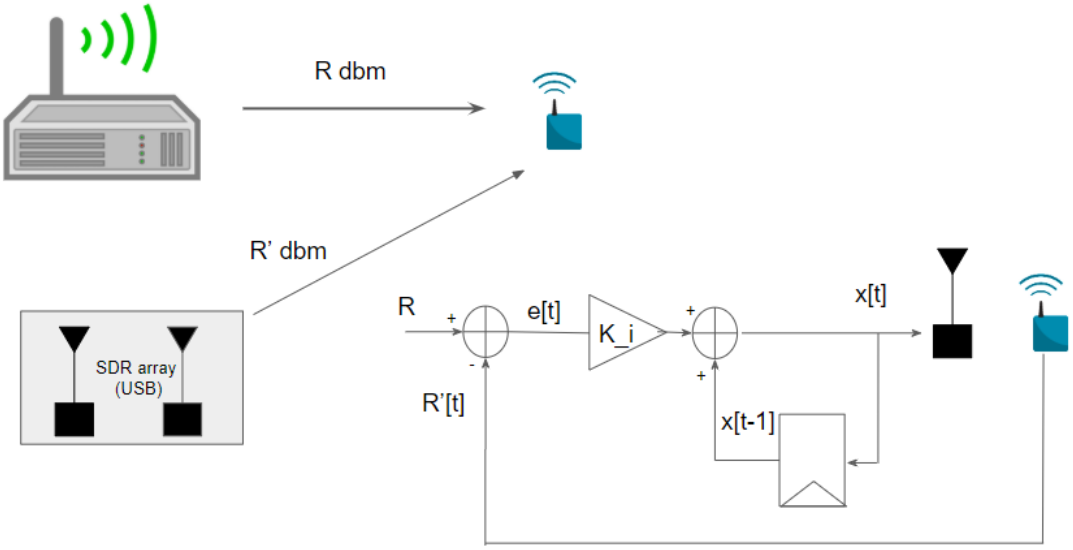
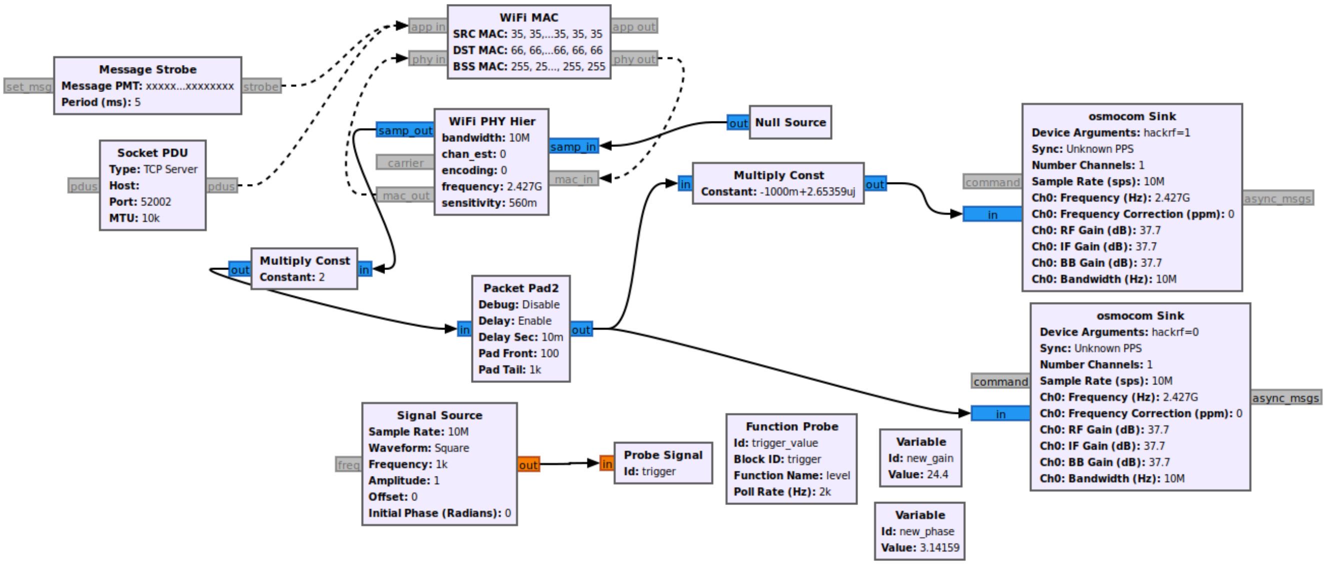


Figure 3.6: A block diagram representation of the gain coefficient acquisition subsystem. R is the recorded energy from the mote when exposed to the real interference source. R' is the measured gain coefficient from SDR array. x is the gain that the SDR array is transmitting at. R' changes over time as the difference between R and R' is fed back into the system to nudge x to a value such that R = R'.

3.5 Multiplexed Interference Transmission

Once the TX gain and bearing values for each mote have been calibrated, SDIG can be deployed. When a variable data rate synthetic transmission is being sent, the gains and bearing are switched periodically so that the signal is optimized for a different mote at any given time. This is achieved in software by generating a separate ramp function between the values of -1 and 1 at an arbitrary frequency of 1000Hz which serves as as a pseudo-timer to switch the gain and bearing weights in time. Adjusting the frequency parameter will not affect the fraction of the time a particular mote will be focused on. For a two mote testbed, the radio will transmit with the gain and bearing for mote 0 when the ramp function is between -1 and 0, and transmit with the gain and bearing for mote 1 otherwise. This can be extended to more motes by further dividing the intervals in the ramp function further. The GNU Radio flowgraph used to transmit the signal at these bearings and gains is shown in Figure 3.7.

This concludes the setup and deployment of SDIG.



Message Strobe
Message PMT: xxxxx...xxxxxxxx
Period (ms): 5

Socket PDU
Type: TCP Server
Host:
Port: 52002
MTU: 10k

WiFi MAC
SRC MAC: 35, 35,...35, 35, 35
DST MAC: 66, 66,...66, 66, 66
BSS MAC: 255, 25..., 255, 255

WiFi PHY Hier
bandwidth: 10M
chan_est: 0
encoding: 0
frequency: 2.427G
sensitivity: 560m

Multiply Const
Constant: 2

Packet Pad2
Debug: Disable
Delay: Enable
Delay Sec: 10m
Pad Front: 100
Pad Tail: 1k

Null Source

Multiply Const
Constant: -1000m+2.65359uj

osmocom Sink
Device Arguments: hackrf=1
Sync: Unknown PPS
Number Channels: 1
Sample Rate (sps): 10M
Ch0: Frequency (Hz): 2.427G
Ch0: Frequency Correction (ppm): 0
Ch0: RF Gain (dB): 37.7
Ch0: IF Gain (dB): 37.7
Ch0: BB Gain (dB): 37.7
Ch0: Bandwidth (Hz): 10M

osmocom Sink
Device Arguments: hackrf=0
Sync: Unknown PPS
Number Channels: 1
Sample Rate (sps): 10M
Ch0: Frequency (Hz): 2.427G
Ch0: Frequency Correction (ppm): 0
Ch0: RF Gain (dB): 37.7
Ch0: IF Gain (dB): 37.7
Ch0: BB Gain (dB): 37.7
Ch0: Bandwidth (Hz): 10M

Signal Source
Sample Rate: 10M
Waveform: Square
Frequency: 1k
Amplitude: 1
Offset: 0
Initial Phase (Radians): 0

Probe Signal
Id: trigger

Function Probe
Id: trigger_value
Block ID: trigger
Function Name: level
Poll Rate (Hz): 2k

Variable
Id: new_gain
Value: 24.4

Variable
Id: new_phase
Value: 3.14159

Figure 3.7: GNU Radio flowgraph deployed to switch interference transmissions between two motes.

Chapter 4

Evaluation Results

4.1 WiFi Interference Performance

Accurately controlling the physical bits sent over the WiFi NIC in time is non-trivial, so it is important to understand how transmissions manifest themselves in the physical layer. Using the testbed topology described in Figure 3.2 (where mote 1 is the mote closer to the SDR array as well as the WiFi interference source), we evaluated the effect of WiFi interference on a 2 Hamilton mote sensor network.

The WiFi access point was configured to transmit on WiFi channel 4 (2427 MHz center frequency). The motes were configured to evaluate the energy readings tuned to 802.15.4 channel 15 (2425 MHz center frequency). These channels were chosen as they had the least amount of traffic. Choosing an empty channel is extremely important as exogenous interference induces confounding effects on our calibration results. For our high data rate transmission, a program was created to repeatedly pipe in 1500 “x” characters over the air via netcat. For our low data transmission, the program would repeatedly pipe in 200 “x” characters with a 5 millisecond break between transmissions.

We first obtained a control result by evaluating the “empty” channel 15. The motes listened to the channel for 10 minutes and recorded the energy readings over this. Figure 4.1 shows a histogram of the results. The lowest value that can be read by the mote is -94dbm and this result dominates most of the energy reading samples for both motes. This is to be expected as the channel is “empty” in this instance.

We then transmitted using the true interference source and performed a similar energy reading measurement. Figure 4.2 shows a histogram of the results. As expected, the median reading of mote 1 is higher than mote 2 as mote 1 is closer physically to the transmitter. Interesting enough, around 5% of the samples were low energy readings even with the aggressive transmission. This brought up the question about whether the transmission itself was rendered as a bursty signal or whether the transmission was rendered as a sustained transmission over the air.

After generating a waterfall graph of the spectral density over time on 802.15.4 channel

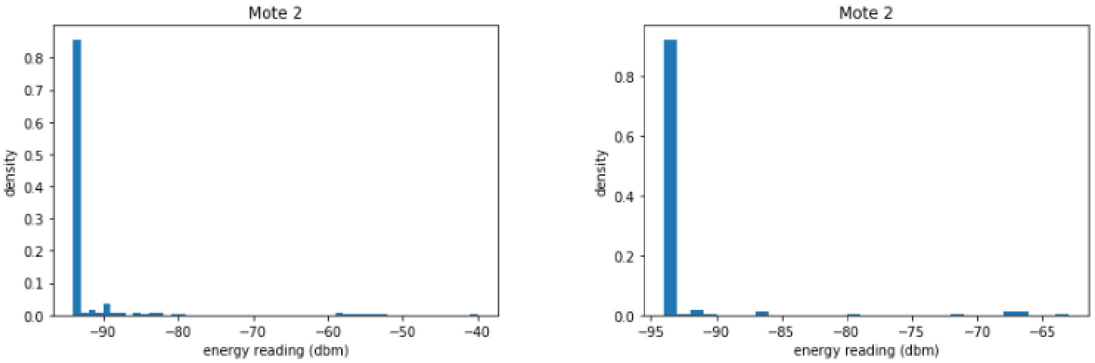


Figure 4.1: Energy reading histogram for mote 1 (left), and mote 2 (right) without explicit WiFi transmission over a period of 10 minutes.

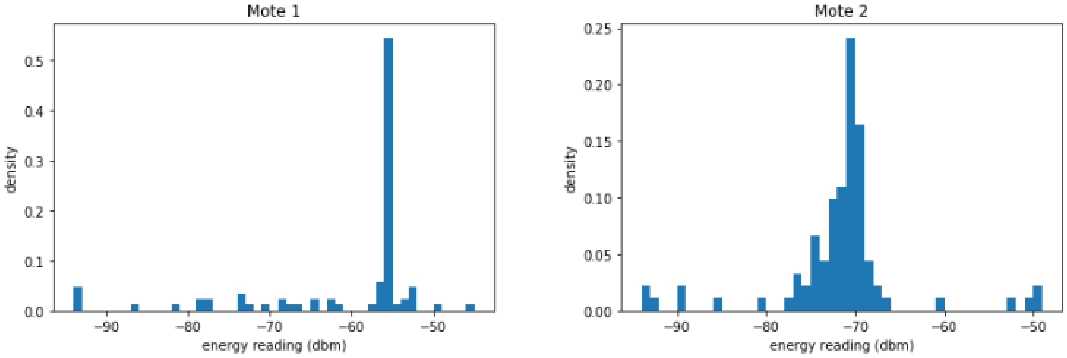


Figure 4.2: Energy reading histogram for mote 1 (left), and mote 2 (right) with an explicit high data rate WiFi transmission. The median readings are -51 and -71 dbm for motes 1 and 2 respectively. Energy readings were measured over 10 minutes.

15, we observed that this high data rate transmission did indeed have significant holes in the transmission in which the channel goes empty (shown in Figure 4.3). We were not able to completely eliminate these regions of inactivity with netcat.

We then evaluate the packet reception ratio from mote 1 to mote 2 and vice versa under no interference, high data rate interference, and low data rate interference. When evaluating mote 1's PRR (packet reception ratio), we had mote 2 send UDP packets with an embedded sequence number every 0.1 seconds to mote 1 while mote 1 was listening. It should be noted that we sent packets using the GNRC network stack. Similarly, to measure mote 2's PRR,

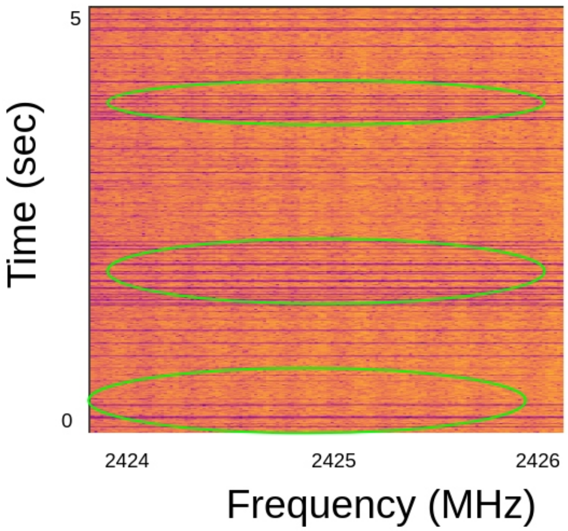


Figure 4.3: Waterfall plot of channel 15 with an active high data rate transmission. The height of the waterfall represents 5 seconds of activity. The highlighted horizontal purple streaks indicate periods of inactivity as the channel is empty for that time period. The horizontal streaks of orange indicate periods of high activity. Although the channel is mostly active, there are numerous periods where there is little activity.

we simply swapped which mote was sending and which was receiving. To estimate PRR in this scenario, we looked at the most recently observed sequence number and divided it by the total amount of packets received. The results are shown in Table 4.1. As expected, the PRR of the motes while under little CTI is near perfect. When the high data rate interference is sent, PRR plummets greatly. We also observed that Mote 2's PRR is significantly higher in both the low data rate experiment as well as the high data rate experiment. This is almost certainly due to the fact that mote 2 is significantly further from the interference source compared to mote 1.

Interference Type	Mote 1 PRR	Mote 2 PRR
None	99.9%	99.9%
Low Data Rate WiFi	67.6%	75.9%
High Data Rate WiFi	1.2%	5.0%

Table 4.1: PRR between mote 1 and mote 2 in our sample testbed over 30 minutes with no explicit WiFi interference, as well as with high data rate and low data rate interference.

4.2 Single Transmitter Evaluation

Using the same setup earlier, we evaluated the performance of SDIG with a single transmitter against 2 motes. This deployment cannot perform beamforming as only 1 antenna is present. This evaluation is done to serve as a comparison point for the multi-transmitter deployment of SDIG in section 4.3. Because it's impossible to determine a bearing with just a single omni-directional transmitter, we only needed to determine the TX gain for the transmitter. We found the transmit gains in order to achieve an energy reading of -51dbm from mote 1 and -71 dbm from mote 2. Our system in this instance calculated a baseband gain of 25db is necessary to achieve an average of -51dbm for mote 1 and a gain of baseband gain of 27.6db is necessary to achieve a gain of -71 for mote 2. We transmit a message of 1500 "x" values with a period of 1ms repeatedly for our high data rate transmission (about 12MB/s). Our modulation scheme is QAM-16 (same observed scheme from the interference source). We verified the integrity of the packet by receiving the signal with the second radio, passing the signal through a WiFi receive flowgraph, and examining the output in wireshark. Transmitting with these gains yield the observed energy detection readings in Figure 4.4. Note how the spread of energy readings is much smaller with this synthetic generation technique. Additionally, there are very few samples that were extremely low (less than -90dbm). This indicates that these synthetic packets are transmitted much more consistently compared to the packets sent with the real WiFi source. These results are validated in Figure 4.5 where the waterfall displays little to no discernible gaps between packets.

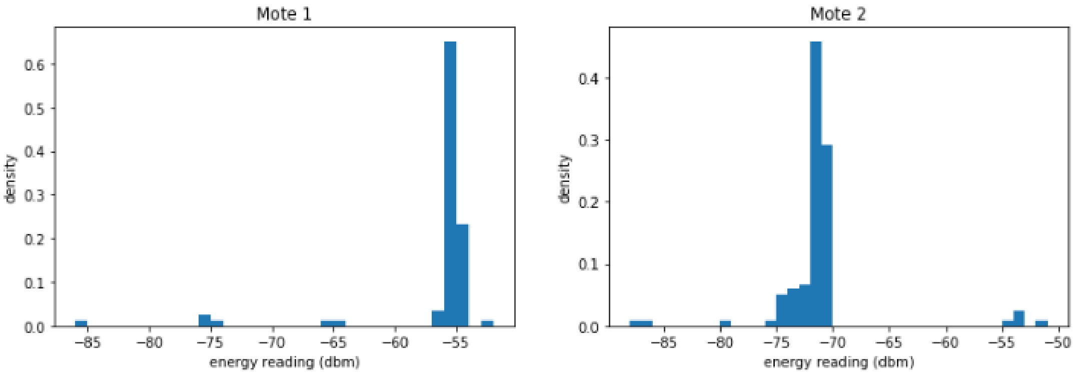


Figure 4.4: Energy reading histogram for mote 1 (left) with 25 db baseband gain, and mote 2 (right) with 27.6 db baseband gain. An explicit high data rate synthetic signal was emitted. The median readings are -51 and -71 dbm for motes 1 and 2 respectively.

We then evaluate the PRR performance of the motes for single transmitter SDIG. We evaluate the system without gain multiplexing by fixing the transmitter to be calibrated to

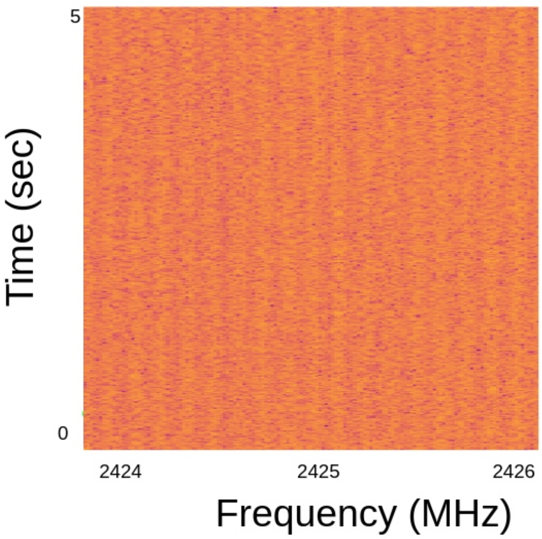


Figure 4.5: Waterfall plot of channel 15 with an active high data rate transmission from single transmitter SDIG. The channel remains consistently active through throughout the capture. The height of the waterfall represents 5 seconds of activity. Compared to Figure 4.3, there is a distinct lack of horizontal purple streaks, indicating that the channel is consistently active.

only mote at a time. We then evaluate single transmitter with gain switching. For each of these data points, we observe the PRR over 15 minutes. The results of this experiment is described in Table 4.2.

If we compare the correctly calibrated results for mote 1 and mote 2, we see a discrepancy in results when compared to the PRR observed with the real interference source. Particularly, when examining the high data rate transfers, we observe that the PRR for both motes 1 and 2 are under 1% when calibrated for each mote whereas the PRR in the real interference case is 1.2 and 5% respectively. This is most likely due to the fact that the actual WiFi interference source is more bursty (refer to Figure 4.3) than the packets being transmitted from the SDR. When evaluating the low data rate performance, even though both the real interference source and the synthetic interference source were sending approximately the same data at the same rate, the synthetic data yielded a lower PRR for both motes (60.5% verses 67.6%, and 70.5% vs 75.9%). A similar theory explaining this discrepancy is how the "on-time" of the transmitters differ between the generators.

When looking at the off-calibrated results, we see a significant change in PRR. Because the transmission gain for mote 1 is lower than that of mote 2, we see that PRR for mote 2 increases significantly when transmitted with mote 1's gain (79.8% vs 70.5% for mote 2). Similarly, we see that mote 1's performance drops significantly when it's subject to mote 2's

Interference Type	Mote 1 PRR	Mote 2 PRR
None	100%	100%
Low Data Rate WiFi	67.6%	75.9%
High Data Rate WiFi	1.2%	5.0%
Low Data Rate 1TX SDIG (Mote 1)	**60.5%**	*79.8%*
Low Data Rate 1TX SDIG (Mote 2)	*55.9%*	**70.5%**
Low Data Rate 1TX SDIG (Switching)	58.9%	76.3%
High Data Rate 1TX SDIG (Mote 1)	**0.5%**	*1.2%*
High Data Rate 1TX SDIG (Mote 2)	*0.35%*	**0.83%**
High Data Rate 1TX SDIG (Switching)	0.46%	0.93%

Table 4.2: Evaluation of single transmitter SDIG for high and low data rate transmissions. Because a single transmitter is used, no beamforming can be done. (Mote x) means the transmitter is using the gain for Mote x. Switching indicates the data is taken when the transmitter is multiplexing between the gain values associated with mote 1 and 2. Bolded values are results from the mote that is properly calibrated to the transmitter. Italicized values are result from the a mote that is receiving packets from a mis-calibrated transmitter.

transmit gain (55.9% vs 60.5%) as mote 2's transmission gain is higher than that of mote 1. When gain switching is enabled, we observe an averaging of the PRR performance from the calibrated and off-calibrated result. In the case of mote 1, the new PRR is worse than the PRR with the calibrated transmitter. For mote 2, the new PRR is better than the PRR with the calibrated transmitter.

Some key insights to takeaway from this experiment is that SDIG with 1 transmitter, even when calibrated to the correct mote, our payload can't perfectly replicate the PRR for a sample interference load from a real transmitter. This is due to the fact that the active time of the transmitter differs from that of the real transmitter. Because the physical layer signal is created at software level, this can be finely controlled and adjusted for future work. Additionally, we observe that gain switching achieves an undesirable averaging effect for the motes. Motes that are closer to the transmit array will see significantly worse PRRs than its calibration PRR and motes further from the transmit array will notice better PRRs. This makes a single transmitter deployment of SDIG undesirable for emulating interference at an arbitrary location. Even when independently trying to interfere with a target node, due to the lack of beamforming, the non-targeted mote is impacted significantly as well. This is demonstrated clearly when Mote 1's PRR becomes worse when the device is targeting mote 2 which means more interference is being detected by mote 1.

4.3 Two Transmitter Evaluation

Using the same setup used before, with the exception of the second transmitter turned on, we evaluated the performance of SDIG with a two transmitters against 2 motes. To sanity check the energy detection readings, we first transmitted the same high data rate synthetic payload and same gain value used in the single transmitter evaluation for mote 1 to both transmitters and measured the observed energy detection on mote 1 (which is in front of the array, so $\pi/2$ bearing). Unsurprisingly, we observed a 3dbm increase in the energy reading, which roughly corresponds to a 2 times increase in energy presence which makes sense as we simply doubled our transmit capabilities.

We then run the bearing coefficient acquisition scheme to acquire the bearings for the motes. The results of the sweep are displayed in Figure 4.6.

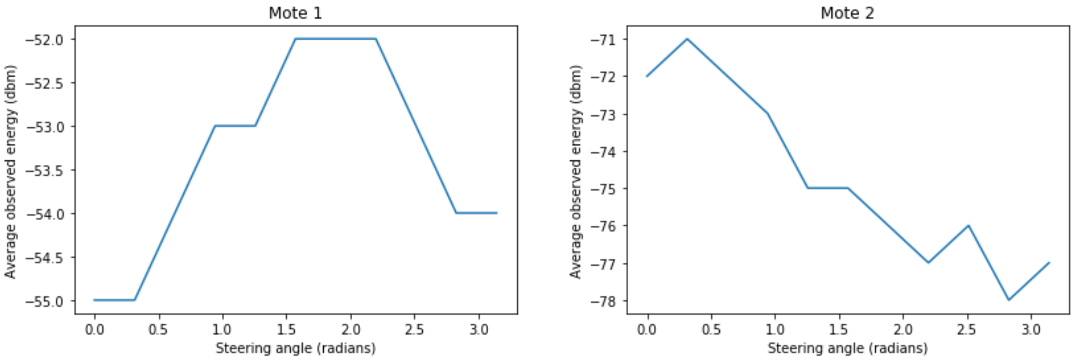


Figure 4.6: Observed energy detection plots from a sweep from 0 to π radians in 11 divisions for mote 1 (left) and mote 2 (right). Energy detection values were measured for one minute per division. In this case, the bearing was determined to be at $\frac{\pi}{2}$ for mote 1 (in front of the transmitter), and $\frac{\pi}{10}$ for mote 2 (which is roughly to the right of the transmitter array).

We then ran the gain coefficient acquisition scheme for each mote to calibrate the transmitters to achieve the correct energy readings that were observed in the true WiFi interference experiment. Note that we preload the appropriate bearing when calibrating the mote. Once we have achieved all the vectors, we evaluated a couple PRR measurements. We observed the PRR from motes 1 and 2 when we: beamform to only mote 1, beamform to only mote 2, and multiplexed between beamforming to mote 1 and mote2. The results are described in Table 4.3.

From these results we see that when the transmitters are calibrated to the motes, the results from the calibrated motes in the single transmitter and two transmitter case are pretty consistent. However, as expected, the off calibration results are significantly different. Though, in the case mote 1, its results weren't as impacted compared to the single transmitter case. Even though a higher gain is used to beamform to mote 2, because the bearing is

Interference Type	Mote 1 PRR 1TX	Mote 1 PRR 2TX	Mote 2 PRR 1TX	Mote 2 PRR 2TX
LDR (Mote 1)	**60.5%**	**61.1%**	*79.8%*	*84.3%*
LDR (Mote 2)	*55.9%*	*59.2%*	**70.5%**	**68.5%**
LDR (Switching)	58.9%	60.4%	76.3%	78.5%
HDR (Mote 1)	**0.5%**	**0.52%**	*1.2%*	*5.3%*
HDR (Mote 2)	*0.35%*	*0.47%*	**0.83%**	**0.4%**
HDR (Switching)	0.46%	0.49%	0.93%	1.2%

Table 4.3: Evaluation of one and two transmitter SDIG. LDR stands for "low data rate", and HDR stands for "high data rate". A column with 1TX indicates a result measured with 1 SDR transmitter. A column with 2TX indicates a result measured with 2 SDR transmitters. Bolded values are results from the mote that is properly calibrated to the transmitter. Italicized values are result from the a mote that is receiving packets from a mis-calibrated transmitter.

significantly different, the net negative effect on the PRR for mote 1 is diminished compared to that of the single transmitter case. In the case of mote 2, the off calibration result shows a drastic improvement in PRR as not only is the TX gain less, but also the bearing is out of alignment, which reduces the negative impact of the transmitter. This demonstrates that we are able to minimally impact other motes when we are targeting a particular mote.

In general, in comparison to the single transmitter switching, the 2 transmitter switching system yields higher PRR in each case. It is apparent that the performance disparity between the switching and non-switching schemes comes from the fact that the system can currently spend a fraction of its time beamforming to one mote a time. This is observed from the mote in Figure 4.7.

We further tested the limitations of multiplexing the signal by calibrating the system to 4 motes. We introduced two TelosB motes into the testbed. Additional objects were also added to the testbed which introduced multipaths during this experiment. Figure 4.8 shows the new topology of the testbed. RSSI readings were pulled from the radio on the TelosB motes and were used to calibrate the gains. Additionally, communication between the motes were strictly Hamilton mote to Hamilton mote and TelosB mote to TelosB mote. The TelosB motes also transmitted on 802.15.4 channel 15.

We performed a low-data rate deployment of SDIG against these 4 motes. We multiplexed the gain/bearing parameters 4-way to each of the devices during transmission. The results are summarized in Table 4.4. As we can see, the PRR discrepancy between calibrated result and switching result is amplified as we add more motes. This shows that one of the main limitations of this schema is the fact that as more motes are added, the less time the system has to satisfy the interference demand of the mote.

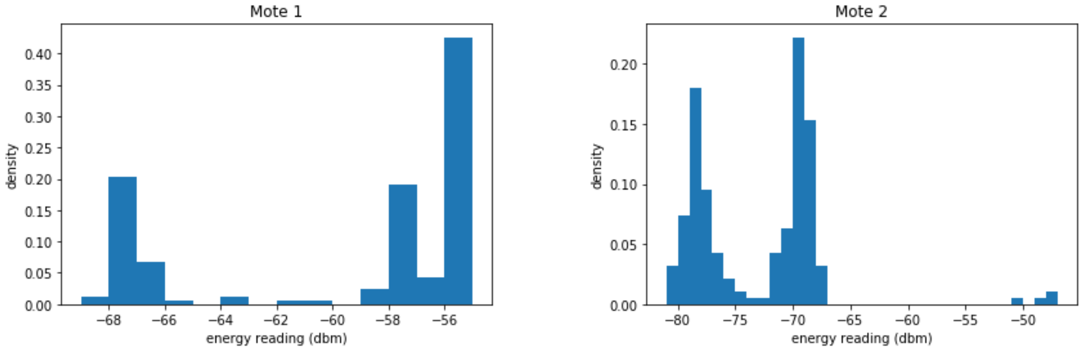


Figure 4.7: Observed energy readings for mote 1 and 2 when subject under 2 transmitter SDIG. Note how there are two peaks in each case which demonstrates the gain/bearing switching schema.

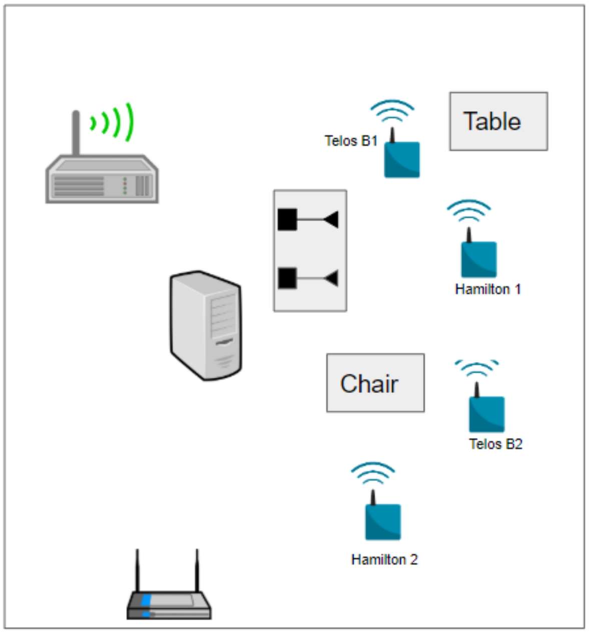


Figure 4.8: New testbed topology. New furniture is added as well as 2 new motes. These motes are connected to USB hubs which are in turn connected to a laptop.

Interference Type	Hamilton 1 PRR	Hamilton 2 PRR	TelosB 1 PRR	TelosB 2 PRR
LDR Calibrated PRR	64.3%	78%	63.4%	66.5%
LDR Switching PRR	72.3%	91.3%	77.2%	82.7%

Table 4.4: 4 mote 2 transmitter SDIG system comparison to focused calibrated PRR compared to PRR with SDIG switching.

4.4 Cross Channel Interference Performance

We also tested how the energy readings for a different 802.15.4 channel would compare when subject to a 2 transmitter SDIG deployment calibrated to channel 15 verses the actual wireless interference transmission. We calibrated our deployment to channel 15 for a single mote and updated energy reading binary on the mote to display the energy readings for channels 14 to 17. We then exposed the mote to interference from the actual WiFi interference source as well as 2 transmitter SDIG deployment. The results are shown in Table 4.5. As it turns out, the calibrated gains only hold well for channels close to the center frequency of the WiFi channel. Although channel 14 and 17 are normally impacted by traffic on wireless channel 4, they aren't as impacted as hard in the SDIG deployment. This phenomenon may be relevant if SDIG were to be deployed to a multi-channel testbed, however, this was not evaluated.

Channel	WiFi Interference	2 TX SDIG Calibrated
14	-50	-67
15	-51	-51
16	-52	-52
17	-54	-72

Table 4.5: Comparison between the energy readings when subject to high data rate wireless traffic as well as high data rate SDIG interference from mote 1.

Chapter 5

Development Challenges and Limitations Discussion

In this section, we discuss technical implementation challenges that were experienced in evaluating or developing SDIG.

5.1 Bearing acquisition

Before settling on the beamforming scheme illustrated in chapter 3, additional development attempts were made to attempt to create a more elegant bearing calculation algorithm.

In the initial draft of SDIG, instead of performing a bearing sweep, a direction of arrival algorithm was proposed as a way of obtaining the bearing. A binary for the motes was developed to cause it to periodically transmit a signal. In this configuration, the SDR array was set to receive the 802.15.4 packet from each transmitter and based off the differences in the received signal on each antenna from each radio, a bearing can be produced. This would ideally eliminate the need for an exhaustive search.

The initial proposed method of determining the bearing involves listening for the start of the 802.15.4 packet on both transmitters, evaluating the time of arrival for each transmitter, and, based off the difference in arrival (DOA), perform some trigonometry to figure out bearing. The time difference would be determined through cross correlation between the radios. However, after some preliminary analysis, we deemed that the resolution of this technique is not sufficient. Assuming both motes were receiving at 10 megasamples per second and perfect synchronization between radios, if radio 1 received the packet 1 sample before radio 2, that would mean that the received signal propagated 100 nanoseconds longer before being received. Given the speed of light in a vacuum, this method would only yield a resolution of 30 meters per sample which is way too coarse. Thus, a time difference approach for direction of arrival is not feasible.

Another approach for deriving the bearing involved using PCA on the measured readings to determine the best bearing, also known as the Capon method [2]. This approach was originally implemented as an open source module in GNU Radio for the RTL-SDR [19]. We attempted to proceed with this method for measuring DOA. In the sample flowgraph, the SDRs will be synchronized via cross-correlation, the phase difference between the received signals would be filtered, and an arc-cosine function is used to output the correct angle. We ported this module to GNU Radio 3.8 and adapted the example DOA flowgraph to use the HackRF sources instead of the RTL-SDR sources. However, we were not able to achieve significant results using this method as the measurements were too noisy. Figure 5.1 illustrates the noisy performance of this method when transmitting 802.15.4 packets. We tried changing the packet length to see if it could be processed properly through the module but with limited success. The original use of this module was to identify the source of a user transmitting using a walkie talkie, so perhaps this package was not particularly compatible the modulation scheme used in 802.15.4.

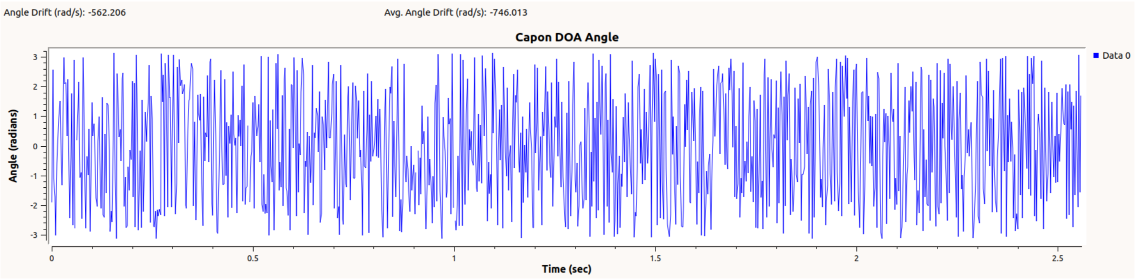


Figure 5.1: Estimation of mote bearing using gr-doa package. This method was ultimately scrapped since the data was too noisy to use.

We also considered implementing the MUSIC algorithm to satisfy the DOA acquisition [6]. However, after obtaining preliminary results with the bearing sweep experiment, an implementation of MUSIC was ultimately not implemented. That being said, performing DOA analysis in an environment with multipaths through extensions of the MUSIC algorithm is not only possible, but has been shown to be accurate to the decimeter level [20]. Future work that refines the beamforming procedure should utilize these more elegant methods of DOA.

5.2 Computational Demands of SDIG

In the first high data rate deployment of SDIG, the system would stop functioning properly after a few seconds. This resulted in the radios not transmitting anything at all. This issue was not observed when generating packets with a high periodicity. Ultimately, it was observed that our machine could not keep up with the real time demand of constructing, tagging, and modulating the packets when the period between the packets is incredibly low. In order to resolve this issue, we generated the complex sample signatures for a few high data rate packets and saved these results to a file. When performing high data rate SDIG, we instead transmit packets from this file repeatedly rather than going through the motions of generating each bit of the packet. Although we sent the same payload every time in our low data rate and high data rate tests, further work involving random messages may have trouble generating random packets at a high enough rate if repeated samples cannot be sent.

5.3 Issues With Interference Multiplexing

As demonstrated in the 2 and 4 mote demonstration of 2 transmitter SDIG, when the transmitters start multiplexing the transmissions to different motes, each mote only directly receives interference for a fraction of the transmission duration. This leads to an overall increase in PRR as the motes will register the channel impacted less often. This fundamentally limits the scalability of motes that SDIG can service in a testbed. To illustrate this, consider the ideal example where we have n motes and a perfect beamforming array barring multipath effects such that only 1 mote receives interference at all at any time instance. Assuming that each mote is fairly served interference, that would mean that the mote would only ever receive interference $\frac{1}{n}$ of the time. Assuming packets are received properly when there's no interference and no packets can be achieved when there is interference, the expected PRR per mote in this ideal system would then be $\frac{n-1}{n}$.

Chapter 6

Conclusion

In this paper, we present the design and implementation of software defined interference generation system against wireless sensor networks without prior knowledge of channel information. By developing a simple energy detection binary for each mote architecture, SDIG can be used to calibrate its steering angle as well as transmit gain to emulate the performance of a real wireless transmitter that is located elsewhere on motes located throughout a testbed. We also demonstrate consistency issues when it comes to using a real wireless transmitter as an interference source. This issue is not observed using SDIG. We believe the mote performance discrepancy is mostly due to the difference in transmission consistency between the true WiFi interference source and SDIG. We show that through the use of multiple antennas and beamforming, we are able to somewhat prevent unnecessary interference levels from impacting other motes as well. Finally, we reveal the scaling limitations of SDIG by demonstrating its inability to match the performance of a focused transmission when it multiplexes its signal to various motes. Our use of the radio is fundamentally different from that of emitting a signal designed to persist through a channel; we generate noise designed to match the effects of the channel from a different source. Ultimately, one of the realizations of this experiment was that providing high fidelity control of interference over a WSN is rather difficult without compromising on either the scalability or the flexibility of the of the system compared to generating an explicit signal.

One way to potentially scale the system is by deploying multiple transmitter arrays to serve a larger number of motes in a testbed. This method is quite expensive as the cost of a single software defined radio with transmit capabilities costs on the order of hundreds of dollars. Furthermore, more computing resources would be needed to drive these radios as well.

Nevertheless, a single deployment SDIG can serve as an invaluable evaluation system on its own in a testbed as its physical interference signature can easily be defined and configured in software to impact a particular spot in space. For instance, a single deployment of a multi-transmitter SDIG can test the effect of an intermittently active mote within a WSN by compromising its receive capabilities. Different interference types (e.g 2.4GHz microwave radiation, 802.15.4 packets, etc), different bearings, and various transmit gain coefficients can all be defined in software. Additionally, the utilization of flow-graphs created in GNU Radio allows interference generation patterns/signatures to be built and shared across different deployments and testbeds. This potentially enables remote testbeds to support a wide spectrum of CTI signatures.

Bibliography

[1] Michael P Andersen, Hyung-Sin Kim, and David E Culler. "Hamilton: a cost-effective, low power networked sensor for indoor environment monitoring". In: *Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments*. 2017, pp. 1–2.

[2] Jack Capon. "High-resolution frequency-wavenumber spectrum analysis". In: *Proceedings of the IEEE* 57.8 (1969), pp. 1408–1418.

[3] *Contiki-NG, the Next Generation Contiki*. URL: `https://github.com/contiki-ng/contiki-ng`.

[4] *GNU Radio Website*. URL: `http://www.gnuradio.org`.

[5] *GR-IEEE80211. IEEE 802.11 a/g/p transceiver.* URL: `https://github.com/bastibl/gr-ieee802-11`.

[6] P. Gupta and S. P. Kar. "MUSIC and improved MUSIC algorithm to estimate direction of arrival". In: *2015 International Conference on Communications and Signal Processing (ICCSP)*. 2015, pp. 0757–0761.

[7] *HackRF*. URL: `https://github.com/mossmann/hackrf`.

[8] Anwar Hithnawi, Hossein Shafagh, and Simon Duquennoy. "TIIM: technology-independent interference mitigation for low-power wireless networks". In: *Proceedings of the 14th International Conference on Information Processing in Sensor Networks*. 2015, pp. 1–12.

[9] Anwar Hithnawi, Hossein Shafagh, and Simon Duquennoy. "Understanding the impact of cross technology interference on IEEE 802.15. 4". In: *Proceedings of the 9th ACM international workshop on Wireless network testbeds, experimental evaluation and characterization*. 2014, pp. 49–56.

[10] Anwar Hithnawi et al. "Controlled Interference Generation for Wireless Coexistence Research". In: *Proceedings of the 2015 Workshop on Software Radio Implementation Forum*. 2015, pp. 19–24.

[11] Anwar Hithnawi et al. "Crosszig: combating cross-technology interference in low-power wireless networks". In: *2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. Ieee. 2016, pp. 1–12.

[12] Song Min Kim and Tian He. “Freebee: Cross-technology communication via free side-channel”. In: *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 2015, pp. 317–330.

[13] A. Krdu et al. “Beamforming for interference mitigation and its implementation on an SDR baseband processor”. In: *2011 IEEE Workshop on Signal Processing Systems (SiPS)*. 2011, pp. 192–197.

[14] S. G. Ku, H. S. Lim, and A. W. C. Tan. “A software defined radio testbed for simulation and real-world testing of RF subsampling receiver”. In: *International Conference on Frontiers of Communications, Networks and Applications (ICFCNA 2014 - Malaysia)*. 2014, pp. 1–6.

[15] FCC Lab. “Report on Trends in Wireless Devices.” In: (2011). URL: `www.fcc.gov/oet/info/documents/reports/wirelessdevices.doc`.

[16] *Multiple device hardware level synchronization*. URL: `https://github.com/mossmann/hackrf/wiki/Multiple-device-hardware-level-synchronization#upgrade`.

[17] *Raspberry Pi*. URL: `https://www.raspberrypi.org/`.

[18] *RIOT - The friendly OS for IoT*. URL: `https://github.com/RIOT-OS/RIOT`.

[19] Todd Moon Sam Whiting Dana Sorensen. “Direction of Arrival Analysis on a Mobile Platform”. GR Con 17. 2017. URL: `https://www.gnuradio.org/grcon/grcon17/presentations/real-time_direction_finding/Todd-Moon-Gnuradio-DOA.pdf`.

[20] Elahe Soltanaghaei, Avinash Kalyanaraman, and Kamin Whitehouse. “Multipath triangulation: Decimeter-level wifi localization and orientation with a single unaided receiver”. In: *Proceedings of the 16th annual international conference on mobile systems, applications, and services*. 2018, pp. 376–388.

[21] X. Zhang and K. G. Shin. “Gap Sense: Lightweight coordination of heterogeneous wireless devices”. In: *2013 Proceedings IEEE INFOCOM*. 2013, pp. 3094–3101.