THESIS OR DISSERTATION TITLE
TITLE LINE 2

by

Travis F. Collins

A Thesis
Submitted to the Faculty
of the
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements for the
Degree of Master of Science
in
Electrical and Computer Engineering
by

December 2012

APPROVED:

Professor Alexander Wyglinski, Major Advisor

Professor Y

Professor Z

Abstract

Put your abstract here.

Acknowledgements

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Since the advent of modern digital communications in the 20th century there has been an explosion in the demand for wireless spectrum. As a result spectrum is becoming an increasingly scare resource(Insert citation). This demand is a direct result of the availability and relatively inexpensive cost of such wireless device. Therefore in such envirorments as militaristic theatres the probability of interfering transmissions has steadily grown to a point where techniques need to be consider to combat such occurrences. More directly, in such situations when interfering signals are partially or completely understood measures can be taken to overcome such difficulties.

In military theatres it is extremely common to observe friendly operated high-power broadband jamming signals(citation). Such devices exists as part of group convoys in several branches of the military and in many other forms in contested territories or war-zones. Unfortunately such devices block both friendly and hostile communications, and current anti-jamming techniques haven't provided a viable solution to this problem. Therefore new avenues should be considered, utilizing more flexible radio technologies.

Understanding how to overcome such challenges is a complex task; with vastly different transmission envirorments and differing operating devices and operating standards. A new system that could combat such downfalls should rely on all friendly information, or be able to construct solutions of its own from a set of tools given to the radio. Such tools should be flexible and easily modified, changed, or improved. This ability to easily change or adapt is a key feature as the technical requirements can change from day to day, or between branches of the military itself. As such a solution should have the following attributes:

- **Flexible:**
- **Resilient:**
- **Hardened:** in changing enviroments

1.2 State of the Art

Current implementations in anti-jamming technology lies on the strateling point of hardware and software in the communications world. This is true because hardware provides the speed and performance needed for digital data transmission, while software provides higher level intelligence and flexibility in such layers as the media access control layer and the network layer of the OSI model (insert citation of OSI model.) For anti-jamming applications, high intelligence allows for mobility again the jammer. Therefore a large implementation in software must be considered when investigating anti-jamming technics.

Insert figure of OSI model

Current anti-jamming technics include channel hopping, spatial retreat, jammed area mapping, node escape, retreat restoration, frame masking, and many more[**?**]. All of these techniques use mechanisms of evasion or despection. These can be quite effective when attacked by generally narrowband, non-dynamic/learning jammers. In the case of wide and ultra-wide band jammers, they fail miserably. This wide-band enviorment is the primary situation of interest, and it generally considered a hopeless scenario. These anti-jam technics are design for specific situations and jammers.

Let us first examine these anti-jamming technics which are broken down into three primary categories: Proactive countermeasures, Reactive countermeasures, and Mobile agent-base countermeasures[**?**]. Reactive countermeasures relies on a varying array of detection mechanisms first to determine if that node is being jammed. These detection methods must be coupled with a countermeasure or the scheme is in operable. Examples of these detection methods include a transmitter-based approach and a receiver-based detection. In a transmitter-based approach, such as ad-hoc networks, a decision algorithm is used based on four metrics: PDR (Packet Delivery Ratio), RSSI (Received Signal Strength Indicator), Physical rate, and Noise levels[**?**]. In the receiver-based detection additional information must be injected into frames to help the receiver determine the number of frames lost.

Since frames can be easily lost in wireless transmissions, the receiver is handicapped when determining the number of retransmissions that have occurred. In the transmitter the PDR is deterministicly determined by the data-link layer, sequence numbers must be added to frames for the receiver to accurately calculate the PDR[**?**]. Several other detection methods exist including using a detected detector, cooperative detection among nodes in a wireless network, and more sophisticated methods of RF fingerprinting[**?**].

Once the jammer has been detected the reactive countermeasures come into play. Many evasion techniques exists to combat narrowband jammers such as: channel hoping, spatial retreat, retreat restoration, hybrid attacks, and many cognitive radio approaches[**?**]. Many of these technics utilize the network itself to adapt to the jammer, which is an appropriate assumpt because without a network communications are irrelivant. Channel hopping is quite simple and can be considered easiet to implement. If a channel is begining jammed simply ẗhoptẗo another channel. This is easily defeated in two cases, the first the jammer follows you or the jammer is simply wideband capable. The second, spatial retreat, is a mechanism to physically evade the areas being jammed. Based on the detection algorithm all nodes in a network try to estimate the jammed region and flee physically in the direction of safer place. Based on their estimation about the jammed region, nodes will utilize shortest path algorithms to determine location of retreat[**?**]. Retreat restoration is focused around how to rebuild a network once the jammer has left. Retreat restoration can be done by coordinated or uncoordinated communication, and the transmissions are based on a pre planned hop patterns among nodes[**?**].

There also exists systems that are design to resist jamming proactively. These hybrid systems[**?**] utilize preventatives measure to resist jamming such as frequency hopping spread spectrum $FHSS$. Spread-spectrum signals are highly resistant to narrowband jamming, unless the jammer has knowledge of the spreading key. In military applications the spreading key is generally created using a crytographic function(NEED citation). More hybrid solutions include synchronous and asynchronous spectral multiplexing where intermediary nodes are used to communicate at multiple channels. When a node changes its channel

because of jamming a neighbor will heal that connection by communicating with the node on its new channel and rest of the network on the old channel[**?**].

The largest problem with these techniques is they all have are designed to combat narrowband jammers, and even friendly jammers. If high powered wideband jammers enter the equation, all of these solutions fall apart. Note these techniques primarily exploit the dimensionality of their envirmonent by simply avoiding the jammer, and all techniques require intelligent flexible hardware solutions. To implement such solutions requires suffisticated hardware implementations, that can be quite rigid for rapidly changing communication envirorments and adversaries. To compensate solutions that push more of the radio operations from their original rigid hardware implementations into the more flexible software domain, provide a more cost effective and elligent solution. These software focused radios, also know as Software defined radios, have provided a solid platform for very adaptive anti-jamming technologies under the name cogntive radios. These radios have the ability to easily learn and adapt to their envirorment, which is the primary requirement of anti-jamming devices.

As mentioned above, it is quite common for the military to self-jam its own channels. Unfortunately this can hinder their own use unintenially. These disrupted users are known as "disadvantage users". They are commonly small mobile hand held devices and cannot simply overcome the jammer computationally or in raw power; therefore, more manageable and eligent solutions must be considered for such disadvantaged users. Beside self-jammming, adversarial jammers must also be considered. Fortunately certain characteristics can be statistically exploited if these jammer abide by certain properties. Since adversarial jammers tend to inject random data or energy to block communication, if these transmissions can be shown to repeat they can be exploited. In the case of self-jamming, the signal characteristic can be know *a priori*; therefore they also can exploted or removed, negating the effects of such devices. Such scheme must consider the energy or symbols of the jammmer that are orthoganol and/or non-orthoganol to the symbols of the communication itself.

The goal of this project is to exploit a self-jammed and statiscially determistic adversarially jammed channel, throught the utilization of cognitive radio, implemented on a software

defined radio platform. Software defined radios, defined as the intersection between hardware radios and computer software[**?**], provide a platform flexible enough to support highly intelligence operations such that anti-jamming requires. A proposed adaptive signal processing software solution for mitigating the effects of both intentional and unintentional jamming (including wideband jamming) via the combination of antenna subset selection, spectral subtraction, and blind source separation (BSS) techniques in order to extract specific transmissions from a mixture of intercepted wireless signals. The goal of our proposed solution, called BLInd Spectrum Separation (BLISS), is to enable reliable, high throughput, and robust end-to-end wireless communications.

1.3 Thesis Contributions

This thesis will contribute the following to the wireless communications and signal processing research communities:

- A basis for blind source separation of define subset of signals, and tools on estimating and removing those signals.
- A practical implementation using over the air communications of a anti-jamming sytem utilizing software defined radios. This implementation will tackle wideband non-orthoganol and orthoganol jamming, and provide evidence of probability of operational.

1.4 Thesis Organization

This thesis will be organized into the following chapters. Chapter 2 provides the necessary background to understand basic communication system design, anti-jamming technics, and signal processing. Chapter 3 puts forward a theoritical simulations and a design of a physical anti-jamming system. Chapter 4 presents the results of the physical implement and analysis of its findings. Chapter 5 concludes the thesis, summarizing the accomplishments and outlines possible future work.

Chapter 2

Background

This chapter provides the background information needed to understand the chapters that follow. It examines the basic outline of a communication system and how non-idealities are compensated for, with addition of multiple input multiple output (MIMO) systems and a unique filtering technique called spectral subtraction. Secondly this chapter investigates common jammer scenarios and anti-jamming solutions. Finally it outlines the necessary hardware and software tools used during in the implementation chapter.

2.1 Jamming

In 1899 Guglielmo Marconi successful transmitted radio messages across the English Channel, and nine months later Alexander Bell was discussing how this could be jammed during wartime[**?**]. Bell stated that such a wireless system can be easily disrupted with simple electromagnetic distrubances. "Its as easy as cutting the wires".[**?**] In the early days of wireless communication, such systems were very fragile but today they have become exponentially more resilient. In the simpliest form radio jamming is defined as the transmission of radio signals that disrupt communications by decreasing the signal to noise ratio (SNR) between the transmitter and receiver(s)(need citation). This jamming can be either deliberate or unintentional. A common example of unintentional jamming is a microwave oven ironically. Microwave oven operate with a wavelength of 122 millimetres which translates to 2.45GHz from the equation shown below. This directly interferes with channels defined under the IEEE 802.11 standard, also known as Wi-Fi(insert citation). Deliberate jamming on the otherhand, is generally more saphisticated and takes many different forms.

$$\lambda = v/f \tag{2.1}$$

Intentional communications jamming is usually aimed at radio signals in a militaristic setting, where consequences are insigificant or out of the relm of the law. In the most rudimentary designs, a jammer will simply tune their own frequency to that of their enemy and with a similar moduation scheme and significant power disrupt the enemies transmissions. The most common types of this form of signal jamming are random noise, random pulse,

stepped tones, warbler, random keyed modulated CW, tone, rotary, pulse, spark, recorded sounds, gulls, and sweep-through(insert citation). These method obviously or subtly disrupt transmissions by inserting electronmagnetic energy into the transmission space of the receivers. Mathematically what is occuring is that the jammer is producing randomly chosen data that is non-orthogonal to the data which the friendly transmitter is producing. Since this jammer's data is pseudo random when his transmissions are added to the enemy's, the result appears to be random as well. Therefore the signal is unrecoverable. As mentioned above, the jammer must produce signals that are non-orthogonal to the enemy of his jamming will have no effect. An example below shows random noise at a significant noise level is added to a previously destinquishable signal.

INSERT FIGURE OF Modualated and noisy modulated DBPSK Signals

2.2 Anti-Jamming

Anti-jamming has been considerably outlined in the introductory chapter, therefore this section will examine more advanced narrowband and wideband techniques that involve filtering rather than avoidance. All of these approaches have various monitary costs, constraints, and power limitations. First of all narrowband mitigation techniques will be considered. These include adaptive filtering, time-frequency domain filtering, adaptive antennas and subspace processing. By combining several of the listed techniques wideband jammers can also be address, under certain conditions. The table below compares these techniques with various attributes.

INSERT TABLE

Adaptive filtering is a well defined solution in jammer mitigation, but is considerably the most limited. Most notably the jammer must be relatively narrowband and the period of the jammer must be relatively short. An example of an adaptive filtering technique is

a suppression filter. Suppression filters assume statistically the signal is gaussian, which results in the optimal filter being linear. This filter essentially solves the Wiener equation for an optimal filter, but generally a Least Means Squares (LMS) implementation is used instead of inverting the correlation matrix[**?**]. The matrix inversion of the correlation matrix is considered a zero forcing equalizer and is extremely unstable in the presence of small noise.

Time-frequency domain filtering attempts to represent the transform the received signal in such a way that it is possible to easily distinguish the jammer from the data signal. A Short-Time Fourier Transform (STFT) can be used to accomplish this goal. A STFT operates by sliding a window across a signal and taking the fast fourier transform (FFT) of that window. [**?**] uses the STFT to break a signal into its frequency components, from this information with a narrowband jammer only a small number of frequency domain bins contain nearly all of the interferer. Therefore these bins can be simply nulled and an inverse FFT is applied to the signal to regain its time domain version. This is very effective with the use of spread spectrum signal with a narrowband jammer.

Filter banks is a second methodology that can be used to reduce spectral leakage in the frequency domain, which is a large problem with the STFT approach. Also filter banks don't inject interference when the jammer isn't present, which is a common problem when the jammer turns on and on. Filter banks provide jammer suppression after their spectral decomposition stage, since at this point sub-band encoding can be accomplished this spectral modification can become excision for the jammer[**?**]. A similar decomposition is the wavelet transform. The wavelet transform is much more flexible than a STFT because STFT has a fixed resolution for a given FFT size unlike the wavelet transform. Subspace processing can also be applied in this way. The jammer subspace can be made to orthogonal to the wanted signal subspace, nullifying the jammer's effects[**?**].

Besides these signal processing methods, physical techniques can be use to do spatial filtering. These techiques make uses of several antennas, and as an assumption the number of interferers must be equal to or less than the number of antennas. The first approach is

called Null Steering. Null steering constantly computes the weights in order to minimize the received energy level. In effect, this technique attempts to steer the antenna away from the jammer. The second approach is called beamforming. Beam Forming tries to adjust the antenna in order to maximize the SNR. In effect, the antenna beam is steered in the direction of the desired signal. It is however, possible to end up in situations where the jammer is in the same direction as the signal source. This is a postcorrelation technique since the desired signal has to be correlated in order to obtain the SNR. Also, prior knowledge of the signal direction and the host location is required(insert citation).

All of these approaches historically applied to spread spectrum communication systems because narrowband jammers fundamentally are considerably easier to deal with in this setting. They are rather straightforward because the jammer effects only a fraction of the transmitter's transmission space. Therefore when wideband jammers exist many of these schemes fall apart. Other avenues or scenarios must be considered in such a situation to overcome this limitation. Before a solution can be considered, additional signal processing and communication theory must be understood. These topic will be examined in the following sections.

2.3 Communication Systems

Modern wireless digital communication systems are based on a rich tradition of analog experimentation and theory. These technologies surround us on a daily basis from cellphones, car radios, GPS, and many more. All these of these devices communicate over wireless links and are built upon the same building block of transmission and reception theory. Many perspective can be taken, but the most generic observation should be taken at the system level. Depending on the level of saphistication these blocks can expand greatly, but still solve the same issue caused by the wireless transmission of digital access across an envirorment. Such non-idealities such as frequency offsets, doppler effect, signal echos, phase shifts, and several more. These must be compensated for to successful receive uncor-

rupted information.

Before the receiver, which is the most complicated part of a communication pair, the transmitter must be examine. The transmitter's primary goal is to send data in a resilient form or structure to create a more managable signal for the receiver. This is accomplished in several steps, and the function or purpose of the overall system determines the sophistication of the design. Figure **??** outlines the major building blocks of the transmitter; consisting of the coder, pulse-shape filter, and frequency translator.

INSERT SYSTEM DIAGRAM OF A TRANSMITTER

The transmitter's soul purpose is the send data that is convient for the receiver to understand, and allow others to use the transmission medium as well. The coding phase of the transmitter can have many purposes and features, but simply it will encode data into a symbol with a form of redudancy or scheme that will help the receiver reconstructed the information more easily. Next the pulse-shape filter is used to help separate data from one another and help maximize the SNR at the receiver. This filtering can be done with an asortment of filter shapes, but the most popular is the raised square-root cosine filter. After the pulse-shaping the signal is translated into frequency information and upconverted to a high RF with a carrier signal. The translation is done with a modulation scheme such a binary phase-shift keying (BPSK) or pulse amplitude modulation (PAM). The is upconverted by mixing the signal with a sinusoid, seen by equation mixing. This done because low-frequency signals such as speech, music, or digital data can be much more efficiently transmitted at higher frequencies[**?**].

cosine(x)*cosine(y)=(cosine(x+y)+cosine(x-y))/2
(2.2)

At the system level, a modern digital receiver can be broken down into a small set of distinct categories or operations: carrier sychronization, timing synchronization, equalization, and frame synchronization. These sections work together in series to provide smooth transmission of data, and many techniques exist within theses categories to accomplish its goal. In most communication systems, after the radio frequency (RF) front-end, the first operation done on the received signal is frequency compensation and down conversion. This compensation needs to accomplished because non-idealities and differences exist between the transmitter's and receiver's oscillator. Therefore this is continually compensated for and corrected. Carrier recovery can be accomplished using several methods that include but are not limited to: squared difference loops, phase-locked loops, costas loops, and decision-directed phase tracking(INSERT CITATION).

After carrier recovery the signal is pulse-shaped with the same filter shape used at the transmitter. This will help maximize the SNR of the signal. Then the signal must be correct again for timing. The problem of timing recovery is to choose the instants at which to sample the incoming signal. This is generally done through a interpolation mechanism of the transmitted signal. Since at the transmitter the signal is upsampled to symbols, a single data point or bit is represented by several received data points. Therefore these points can be interpolated together for a more accurate estimate of the original data. Timing recovery also can be done with a several methods including: output power maximization, Mueller-Muller method, and decision-directed.

After this point the receiver designs can vary greatly, as the design in this thesis will present, because this is where most of the digital signal processing (DSP) will take place. This section, call Equalization, is responsible to correcting any effect the channel has on the signal. This includes multipath, noise, and other distortions that cause intersymbol interference (ISI). Equalizer implementations are designed to compensated for types of disturbances that occur under certain systems. The equalizer stage is most often coupled with

the frame synchronization stage so the equalizer itself can adapt to changing conditions. This is known as soft decision making. Equalizer techniques include but are not limited to: LMS, decision-directed, dispersion-minimizing, viterbi, blind, and turbo equalizers.

2.3.1 Equalizers

Equalizers can be considered the most complicated design of an entire communication system since they combat a series of distortions. The primary result of these distortions is called intersymbol interference (ISI). ISI simply means that symbols interact with one another in the channel space and cannot be considered independent from one another. Since this interference is generally considered a frequency selective disruption or disperation a filter needs to be employed to reverse such effects. This filter must be adaptable because the channel distortion cannot be know prior to transmission.

As listed in the previous section, many equalizers exists and operate under specific conditions. Here several linear equalizers will be discussed in detail including maximum-likelihood sequence detection, adaptively trained equalizer, and decision-directed linear equalization. The goal of all of these equalizers is to find an FIR filter that when convolved with the received signal produced the original transmitted data $\hat{\mathbf{x}} = ye$. The figure below outlines a typical FIR structure for which the equalizer will create the appropriate coefficents $c_0, c_1, ..., c_n$ for. These equalizers also examine the condition of an additive white gausian noise (AWGN) channel and uncorrelator or independent interferers.

INSERT FILTER STRUCTURE DIAGRAM

The Zero Forcing Equalizer (ZFE) uses peak distortion criteria to determine equalizer coefficents. If $H_c(f)$ is assumed to be the effects of the channel, the ideal equalizer would be $H_{Eq}(f) = 1/H_c)(f)$. This can also be consider the inverse of the channel. The filter coefficents are modeled as weighted pulses convolved with the channel show be the equation below.

$$p_{eq}(t) = \sum_{k=-M}^{M} w_k p_r(t - kT)$$

Unfortunately the ZFE has a large disadvantage, it cannot compensate for small amounts of noise. Technically it will amplify all noise of the received signal, and if any elements of the channel matrix are considerable small then the equalizer becomes unstable. Therefore this is generally considered a more theoretical or elementary equalizer formulation. To overcome this problem the zero ISI condition must be relaxed allowing for noise which if small can easily be overcome by such operations as quantization or decision making. The Linear Minimum Mean Squaed Error Filter (LMMSE) takes this relaxation into account.

The LMMSE assumes that the symbols are uncorrelated with one another and uncorrelated from the noise in the channel. This approach tries to minimize the mean square error, a common measure of estimator qualities. The estimator is defined as $\hat{x}_{MMSE}(y) = x|y$. If x and y are jointly Gaussian, then the LMMSE will be linear. This function or equalizer design minimizes the mean square error. To simplify further an extension to random vectors can be examined. An estimate can be made for the original vector x represented by $\hat{x}$, resulting in the linear equation $\hat{x} = Ay + b$. The LMMSE will minimize the mean square error $\|x - \hat{x}\|^2$.

Besides these linear equalizers outlined, an adaptive approach can also be considered. The LMS or Gradiant algorithm utilizes a traditional technique for minimizing the error in a signal. This method is historically known as the "Method of Steepest Decent" or "Newton's Method". By calculating the error of each received symbol, this can be fed back into the system for future symbols. This error with shape the equalizer's filter coefficents to match the inverse of the channel. The equations are outline below:

$$y[n] = w[n]^H F[n]$$

$$e[n] = A_n - y[n]$$

$$w[n+1] = w[n] + \mu[n] F[n]$$

In these equations μ acts as the algorithm's stepsize determining how quickly it will converge. It must also be considered that the larger the stepsize the higher the probability it may become unstable. As long as the channel's effects are slow changing this equalizer can easily maintain up to date estimates while corrupting little of the data as possible.

All of the methods proposed so far require known data to correct against. This data is called trainging data and generally comes in the form of a preamble in a frame. The preamble is added to the beginning of each frame so the equalizer can learn from the effects on that specific data. The preamble is the same for all frames and is always used so the equalizer will always be learning. But what happens when data is unknown in the frame, for example the data portion of the frame. This is where blind equalization comes into play.

Several blind equalizers exist but an extension of the LMS equalizer for blind situations will be examined here called the decision-direct equalizer. For a blind equalizer to operate an error generation mechanism must be evaluate, but since the data symbols are unknown, a decision device must be used inplace. This decision device is a quantization method and error is generated from this quantization. This error generation is elaborated from the equations below:

$$e = 1/2(sign(y[k] - y[k])^2$$

This is quite similar to the original LMS implementation except instead of a known symbol the data is quantized using the sign function. This type of quantization using the sign function is only applicable is binary modulation schemes such as BPSK. This equalizer method is usually combine with a training equalizer method in practice, since if a nearly closed eye is observed this equalizer cannot open it.

In this section we have examined several equalizer technics while outlining their advantages and disadvantages. Most techniques require some training mechanisms to operate under heavy channel distortion, and blind techiques such as decision-directed equalization will fail under these conditions. Unforunately such training data can take considerable re-

sources, lower overall data throughput. In practice as much as 20% of frame information is training. Therefore other technique must be considered to help overcome this obsticle.

2.3.2 Superimposed Training Equalizer

As mentioned in the previous section, many implementations exist for equalizer designs, but this thesis will examine the effectiveness of superimposed training symbols in frequency selective channels. In traditional equalizers, channel estimation is achieved through the use of training data or pilot symbols. These symbols are both known to the transmitter and receiver, providing the basis for an estimate. In these equalizers all training symbols are placed at the start of a frame,[**?**] shows that under high SNR training-based schemes are capable of capturing most of the channel capacity, while under low SNR they are highly suboptimal. Superimposed equalizers try to overcome this problem along with other to provide more optimal estimates. Superimposed equalizers physically add its training symbols to the data stream instead of concatinating symbols, saving precious bandwidyj bandwidth[**?**]. To accomidate such pilots, energy must be shared among the data and hidden pilots[**?**]. [**?**] shows that fora transmitter of fixed power, with an additive pilot sequence the decrease in data signal power is equal to

$$K_{loss} = \frac{E[\|s(k)\|^2}{E[\|s(k)\|^2] + E[\|u(k)\|^2]}$$

equivalent to $10logK_{loss}dB$ in signal to noise ratio (SNR). Other disadvantages include an increased signal envelope fluctuation that can be undesireable in nonlinear transmit power amplifiers[**?**].

At the receiver, channel estimation can be done using several techniques in both the frequency and time domain. [**?**] examines a time domain approach for synchronized averaging of the received signal. It is important to note that this synchronization isn't related to transmitter and receiver synchronization. [**?**] and [**?**] both assume that the signal $x(n)$ and noise $v(n)$ have zero mean and $E[m_x(n)] = d(n) = p(n) * h(n)$. Therefore since $p(n)$ is the known superimposed periodic pilot sequence, $h(n)$ can be determined. $h(n)$ is gen-

erally considered frequency selective, and such channels can be quite difficult to deal with esspecially with multipath. Multipath interference is a distortion cause when copies of the original signal arrive at the receiver delayed ontop of the originally received non-delayed signal. This delayed signal essential took another path to the receiver, and this interference is commonly called ghosting in such applications as television broadcasts(INSERT CITATION).

Superimposed equalizers are able to better compensate for large multipath channels because they can spread their training symbols throughout the signal itself. This spreading not only provides a spreading in time but also in other dimensions such a frequency. Therefore if the training symbols are chosen correctly and place correctly, then the can be spread across the frequency spectrum efficently and capture its selectivity. Before the pilots can be examined, the channel must be defined. The channel will be of block length N, and the channel is also time invariant across single blocks, but variable across blocks. The memory of this channel is of maximum length $L-1$, and the impulse response of the channel is defined as $\mathbf{h} = [h_0, ..., h_{L-1}]^T$. Since there are N blocks in the channel, the channel matrix H is modeled as an NxN circulant matrix, with the received signal as expressed as:

$$\mathbf{x} = H\mathbf{s} + \mathbf{v}$$

Here $\mathbf{v}$ is assumed to be zero mean white noise. The vector s is a combination of known training symbols and unknown data. The optimal placement for such training is where the channel undergoes nonergodic fading considered here [**?**]. [**?**] continues on to say that optimally, assuming symbols are placed in clusters of length $\alpha \geq 2L+1$, this scheme is quasi-periodic. The variable α represents the cluser size in this scenario. It is also important to note that this placement makes sure that the training is always orthogonal.

Another consideration that must be considered is how these training symbols interfer with the data itself, and is the training symbols dependent on the data or even the modulation scheme.[**?**] examines this aspect and proposes a solutions that provides a data independence condition. As explained, since the training data is periodic it can be placed

in equispaced frequency bins, while data is spread across all frequency bins. Therefore the pilot must be design to distort the data vector of the discrete fourier transform is zero. In the superimposed training data case, this is done by using the cyclic mean of the data. Therefore all that needs to be done is the removal of the cyclic mean $\mathbf{e} = \mathbf{Jw}$. J is the kronecker product of an identity matrix and the fractionally spaced locations of the pilot tones. Therefore at the pilot frequency only the training symbols are visible for the channel estimation. Formally here is the transmitted result including pilots and data:

$$s = (I - J)w + c$$

.

In summary, modern research on superimposed training focuses primarily on the training symbol generation for a certain type of communication sytems design from single transmission to MIMO. Unfortunately little to no physical implementations exists for such system. This is true because of the sychronization issue that exist when using superimposed training symbols. Since they are directly placed with transmittion data it can be difficult to determine their locations in a sequence blindly, which is done in real world systems. This problem must be considered when phyiscal implementations are proposed.

2.4 Software Defined Radio

For the past two decades there has been a peradyme shift is the definition of a radio device. The conversation has to do with the question of where hardware ends and where software begins. The term Software Defined Radio, coined by Dr. J. Mitola III, defined as a set of digital signal processing (DSP) primitives, a metalevel system for combining the primitives into communication system functions (transmitter, channel model, receiver, etc.), and a set of target processors on which the software radio is hosted for real-time communications[**?**]. Dr Mitola understood how software provided the flexibility that hardware never could, and as time made it more maliable SDR would become dominant.

SDR's can be flexible enough to avoid the "limited spectrum" assumptions of designers of previous kinds of radios, in one or more ways including: ultrawideband transceivers, cognitive radio, dynamic mesh networks, software-defined antenna arrays among others[**?**]. One of the first SDR implementations was a project called "SpeakEasy". The original purpose of SpeakEasy was to use programmable processing to emulate more than ten existing military radios, operating in frequency bands between 2 MHz and 2 GHz[**?**]. Therefore with this single radio, the operator could talk to ten radios operating under ten different standards. As simple enough idea, but unforunately the implementation left much to be desired. For example, physically the device encapsilated the entire back of a common pickup truck[**?**]. This might be great for a ground station that doesn't move, but for a mobile unit this was highly impractical. Secondly in 1992, field programmable gate arrays (FPGA) required significant time, compariatively to reflash or change their operational parameters. Again this also limited SpeakEasy's flexibility.

Today the prime implementations are within cellular basestations and other military applications such as the JTRS project. The JTRS or Joint Tactical Radio System, was a program of the US military to produce radios that provide flexible and interoperable communications. Examples of radio terminals that require support include hand-held, vehicular, airborne and dismounted radios, as well as base-stations[**?**]. Again this project still has limited results and many setbacks have occured. Commerically, from a wide spread penetration standpoint, SDR is still many years away. The two barriers to this are speed and size. To provide enough data throughput, modern SDR's need to quite large physically, which is a serious drawback in many applications. Aside from these limitations, SDR's provide excellent flexibolity esspecially in a laboratory and proof of concept envirornmnet. Rapid protoyping is an obvious place where such radios shine, allowing massive changes without hardware modification. To support this flexibility several software packages have been constructed around the SDR concept, allowing for aggresive prototyping. The two examples discussed here were selected because of operability with the selected hardware, which will be discused future in chapter 3.

2.4.1 GNU Radio

The first software package to be discussed by this thesis is GNU Radio. GNU Radio provides the reconfigurable signal processing blocks that are necessary for software defined radios. GNU Radio is an open source project allowing for SDR developers to develop unique signal processing blocks and SDR systems. GNU Radio was started in 2001, originally forked from the SpectrumWare project developed at the Massachusetts Institute of Technology(INSERT CITATION). Since 2001 the codebase has undergone massive changes, containing almost no code from the original SpectrumWare project. Physically the code consist of three languages Python, C++, and SWIG. Python provides the overarching control of the system or program, while C++ provides the actual signal processing blocks and mathematics. SWIG is a wrapper for C++ which allows Python to dynamically wrap around C++ and control or compile with it. A diagram below better illustrates this architecture. It is also important to mention that there as significant paradyme shifts in the community, pushing more and more code to Python rather than C++, due to its easier programming syntax and structure.

INSERT DIAGRAM ABOUT GNU RADIO's CODE STRUCTURE

2.4.2 MATLAB

MATLAB is an extremely well known engineering, mathematical, biological, and financial software suite. MATLAB provide massive data leverage and advanced communication system models and algorithm for significant data processing. Since 2007, they have also provided hardware compilance with specific SDR platforms through their Simulink platform, and more recently within MATLAB itself(GET LINK FROM MATLAB WEBSITE ON SDRu). This thesis primarily utilizes the signal processing and communication system aspects of matlab, since MATLAB cannot fully utilize all aspects of the chosen hardware. It is important to note under alternate constraints, MATLAB can provide adiquate performance directly interfacing with hardware, esspecially when accessing its targeting features

seen here(INSERT LINK ABOUT MATLAB TARGETING). Below shows an example of a common MATLAB SDR model through Simulink.

INSERT PICTURE OF SDR Model

2.4.3 Comparision

It is important to compare GNU Radio and MATLAB, from a user's perspective they perform quite differently. Firstly GNU Radio, is extremely fast, will the ability of sustaining the maximum throughput of the selected hardware. GNU Radio is also multi-threaded, and while mainitaining high throughput and complete background tasks on mutlicore machines quite easily. This performance has a cost, comparitively GNU Radio has an extremely learning curve and debugging can be challenging. But if you need the performance GNU Radio is your option, can does provide significantly more advanced hardware support is SDR implementations. If data analysis is more heavily desired MATLAB is the obvious choice. MATLAB provides easy and advanced data visualization functionality, and built in tools for analysis. Since MATLAB doesn't compile itself normally, it can be much easier to debug and solve problems. MATLAB's syntax provide similar data manipulation, especially in communication system primatives. Therefore it can be a rather simple choice, speed or ease of use.

2.4.4 Summary

This chapter outlined and examined the topics of jamming and anti-jamming techniques, and provided a foundation in communication system theory and advanced equalizer design. Secondly it setup an understanding of Software-Defined Radio, the power of such an architecture, and examples of implementations and exisiting software for future designs. Next this thesis will consider a new anti-jamming technique and design an implementation of such a system. After the implemenation is investigated, the result of specific experiments on such an implementation will be analysed.

Chapter 3

Implementation

3.1 Overview

Now that a significant background has been provided, the problem this thesis combats will be futher framed and defined. This chapter outlines the proposed implementation of a receiver design, for wideband jammer scenarios and low-mobility situations. An adaptive signal processing software solution for mitigating the effects of both intentional and unintentional jamming (including wideband jamming) through a combination of three techniques. These include: antenna subset selection, spectral subtraction, and blind source separation (BSS), which work in conjuction with one another to extract specific transmissions from a mixture of intercepted wireless signals. The goal of the proposed solution, called BLInd Spectrum Separation (BLISS), is to enable reliable, high throughput, and robust end-to-end wireless communications, especially high capacity multimedia (voice, data, imagery) transmissions. In particular, the focus of the proposed work is the so-called "disadvantaged user". These users are generally considered limited in transmission and processing power such as small-deck combatants, submarines, unmanned air vehicles (UAVs), dispersed ground units in urban and radio frequency (RF) challenged environments.

The BLISS solution integrates three well-known adaptive signal processing algorithms found in the open literature: antenna subset selection, spectral subtraction, and blind source

separation. Each of these algorithms is employed within the BLISS framework in order to enable the process of extracting individual transmissions intercepted from several mixtures of wireless signals. Although blind source separation can readily extract transmissions under ideal conditions, the BLISS system is aimed at harsh spectral envirnorments consisting of many users and in some cases jamming devices. Therefore BSS will not provide adiquate signal separation for robust throughput. Hence, the other two algorithms, spectral subtractions and antenna subset selection will aid in this effort.

In previous sections it has been understood that current anti-jamming techniques cannot compensate in deterministic wideband jamming scenarios. These scenarios must be throughly understood before a practical solution can be provided. For this thesis, the worst case scenario will be considered for the jamming device. For simplification a narrowband jammer will be considered as an adversary, and the transceiving devices cannot frequency hop thus remaining on the same frequency as the jammer. The jammer has an identical modulation scheme as the friendly tranceivers and the constellation is in phase. Finally the jammer is assumed at a similar distance and transmit power as the friendly tranceiving devices. Under these conditions the jammer is completely orthogonal and historically impossible to remove.

This chapter is broken down into several sections which include a system level overview, the hardware and software choosen, signal removal evaluation, the superimposed equalizer design, and the antenna subset selection work. Each of the systems that makeup BLISS have different purposes and goals allowing them to tackle different problems that occur. It is important to note that these systems are at differing stages of development due to the limited time and initial development put into these blocks.

3.2 System

To provide a more straight forward explaination of the BLISS system it is appropriate to provided a system level overview. The system's original purpose was to remove the effects of narrow and wideband jamming. It accomplishes this goal through a series of processing blocks and a selection block. These blocks include: the antenna subset selection (AntSS) block, spectral subtraction block, and finally the blind source separation block. The figure below shows the interconnections between these blocks and certain modification were made from the original design of the system due to practical constraints. These changes will be brought fourth as the blocks themselves are discussed in detail. Since an external research group is responsible to the AntSS block, it will not be throughly discussed by this thesis, but its fundimental purpose will be examined.

INSERT BLOCK DIAGRAM OF OVERALL SYSTEM

The first step in the BLISS system is to pass through the AntSS block. Physically this block is equipped with many antenna in groups of 4. As the block title portrays a subset of these antennas will be selected and they will be passed on to the next block. Precisely a 2^{M-to-2^N} downselection from an array of receive antennas to a set of BLISS receiver inputs. Each individual AntSS board provides 4-to-2 antenna downselection through a set of RF switches. The goal of AntSS is to provide spatial separation through an array of antennas maximizing the SNR of the wanted signal. It is important to note that the antenna spacing must be adiquet to provide enough separation or independence, depending on the operating frequencies or wavelength of the signals themselves. Once the appropriate antennas are selected two signals are to the spectral subtraction block.

The spectral subtraction block is next, which is used to removal known unwanted signal from the spectrum so the source separation block and work properly. The original design of the spectral subtraction block is to use an existing audio technique of removing noise or signals in the frequency domain through a subtraction and smoothing technique.

This technique was discussed previously in the background section, therefore its historical literature will not be examined futher. To enable removal of unwanted signals, the Spectral Subtraction block maintained a database of known power spectral densities (PSD) of common modulation schemes. A recognition system would be implemented to automatic identification of the interfering signal and the block would simply subtract it out, through its already known estimate from its database. Next the newly subtracted signal would be passed to the Source Separation System, where the signal would be unmixed.

The source separation block separates signals when only their mixtures are observed. The operation is called blind, since the signal sources and mixing procedure are unknown to the receiver. Under some conditions this constraint cannot be completely upheld. This is true because the solutions needed to solve such an event become generally intractable. An initial approach in this project was to use a technique called AMUSE (Algorithm for Multiple Unknown Signals Extraction)[**?**]. AMUSE works by first collecting an estimate of the covariance matrix of the received signal, computing the singular value decomposition of that covariance matrix, then performing several transforms on the received signal once the number of mixed received signals is known. Then a covariance is calculated from these transformed received signals which are offset by some instance $\tau, and a eigenvalue decomposition is done upon these covariances. From this decomposition the singular values$

$$R_y = E[yy^t]$$

$$z = Cy$$

$$R_z = E[z(\tau)z(t-\tau)^t$$

$$\hat{x} = V^t Cy$$

It is important to note that for simplicity the mixing matrix for the original proposed solution involving AMUSE is generally constructed as a linear time invariant (LTI) system. There is some activity occuring with nonlinear mixing, but that was considered outside of the scope of this problem.

3.3 Hardware and Software Platforms

Before any implementation was considered a platform needed to be choosen for the end result. This selection provided the work flowpath for the implementation, eliminating many options. As discussed in previous chapters, the end result wants to leverage the power of Software-Defined radios (SDR). The hardware platform chosen was the USRP2 designed and built by Ettus Research[**?**]. These radios are readily available in the Wireless Innovation Laboratory and since the number of radios required for the design was still unknown, it was an obvious choice. There are several software packages that support the USRP2 hardware and several will be examined in this chapter.

The USRP2 or Universal Software Radio Peripheral are intended to be a comparatively inexpensive hardware platform for software radio, and is commonly used by research labs, universities, and hobbyists[**?**]. The USRP2 connects directly to a host computer through a Gigabit Ethernet link, which relays baseband sample that have been receiver or tobe translated. The motherboard provides the following subsystems: clock generation and synchronization, FPGA, ADCs, DACs, host processor interface, and power regulation. Several of these component are seen in the image below. These are the basic components that are required for baseband processing of signals. A modular front-end, called a daughterboard, is used for analog operations such as up and down conversion, filtering, and other signal conditioning. By replacing this RF daughtercard many different frequency ranges can be examined.

IMAGE OF USRP2 MAINBOARD

The information flow is important to understand within the phyiscal radio. This SDR block diagram shown below, outlines the common tasks done by the: daughtercard, FPGA, DAC/ADC, and host computer. Since the FPGA is programmable the operations can change if desire, but the three dominating software packages that utlize the USRP2 flow this structure. Beginning on the far left of the diagram and continuing to the right, at

the daughtercard are RF emitions are received and transmitted. The daughter also contain mixers that translate the signal to an intermediate frequency. Next come the dual 100 MS/s 14-bit ADCs, dual 400 MS/s 16-bit DACs, two digital down-converters with programmable decimation rates, and two digital up-converters with programmable interpolation rates[**?**]. These are located on the mainboard of the USRP2 itself. The FPGA is a Xilinx Spartan 3 XC3S2000, which with the current FPGA software is 59% free in general logic but only 3% free in memory. The FPGA also does not have any DSP resources. The limited memory left in the USRP2 FPGA severely limited any additional development. As a result on newer models, such as the N210, the FPGA has been upgrade.[]

SEE http://confluence.qu.edu.qa/display/NPRPRESEARCH/USRP2+Testbed for image

The data itself constain several pieces of metadata in a frame. RX metadata structure for describing sent IF data includes time specification, fragmentation flags, burst flags, and error codes. The receive routines convert IF data headers into metadata[**?**]. Such metadata can be used to indicate the position and FPGA timestamp associated with the sample that corresponds to the start of the underlying frame. By default, existing blocks will transparently propagate any attributes contained on their input streams to their output streams. Blocks that use the attributes can query their input streams to locate all (key, value, offset) tuples in the region of the stream that they are currently working on in their work method. Likewise, blocks can copy, add or delete attributes on their output streams[**?**]. This knowledge is extremely useful when doing multiple receive antenna arrays when alignment is necessary, or in any situation where fine timing information is required.

With the USRP hardware several software options are available including: GNU Radio, MATLAB, LabVIEW, and several custom packages. MATLAB and GNU Radio have already been discussed, therefore the selection between them shall be discussed. Since this system is a MIMO implementation signal alignment is a requirement. MATLAB doesn't support sample alignment in a multiple USRP system. The sample alignment is possible

through either external means through an external clock or through the option choosen here the MIMO cable. The MIMO cable, a picture of it can be seen below, is a standard 16-pole flatcable to connect tvrx, basic-rx or dbsrx boards. Of this 16pin flatcable only two pins are used (io15 and ground)[**?**]. An image also of the combined dual radio source block can be seen below from GNU Radio. With this requirement GNU Radio must be used for direct access with the USRP2. Aswell full implementation of the systems blocks were first attempted with GNU Radio. Fortunately, if necessary, data can be passed to MATLAB for signal processing from GNU Radio through the use of the file blocks and a script located in Appendix A.

PICTURE OF MIMO CABLE SIDE BY SIDE WITH DUAL GNU RADIO MIMO BLOCK

3.4 Spectral Subtraction

Now that the a formal system level approach has been presented and hardware setup chosen, a more detailed understanding of the blocks themselves can be examined. The goal of the spectral subtraction block is to removal signals to allow the blind source separation block to work properly. As discussed previously signals would first need to be identified and then removed based on information supplied in a precompiled database of known signals. The technique to remove such signals is called spectral subtraction, which primarily takes place in the frequency domain. This approach only relys on known PSD's of the interfering signal. Initially this technique seemed quite sound, but futher investigation proved otherwise.

Initial simulations were created to examine this spectral estimation technique at RF frequencies rather than the standard audio frequencies for which Spectral Subtraction is formally used. Only two signals were used in these simulations, both utilized the same modulation scheme and pulseshaping filters. The signals were chosen to be non-orthoganal, since

when they are orthogonal The frequency of the interfering signal was varied, and so were the oversubtraction parameter $\alpha and quantization floor \beta. Through experimentation \alpha worked best at a value greate$

INSERT GRAPH

As you can see this spectral subtraction technique operates extremely poor when the signal are overlapping at all. The reason system performs well at large frequency shifts is due to the bandpass filter which is used before the signal is quantized. The reason the result is poor is because the estimate is largely incorrect. Since the subtraction only utilzes the PSD's of the signals, half of the information is completely ignored. This results in a completely inaccurate estimate. The problem with traditional Spectral Subtraction is that its results are subjectively evaluated, which isn't accurate enough in a digital communication system. NEED MORE CITATION ABOUT SPECTRAL SUBTRACTION SUBJECTIVE TESTING

Since the initial simulations for traditional Spectral Subtraction proved inadiquated other optioned needed to be explored. First though, the problem needed to be an analyzed further for better understanding, then the appropriate solution could be formulated. Since the interfering signal and the wanted signal are non-orthoganal to one another they will share dimensional space, in this case the signals are inphase with one another. Therefore both planes real and imaginary must be considered. Non-orthoganal signal removal is a common task in communication system, which is done primarily by equalizers. Therefore an equalizer approach was considered next.

3.4.1 Equalizer Approach

The equalizer approach used in this Spectral Subtraction approach is a Least Means Square (LMS) equalizer, utilizing training data used in the front portion of each transmitted frame. This a common equalizer used in practice, allowing for future translation into a realized implementation. The LMS equalizer was also chosen for it robustness no

noise, which is a weakness of such equalizers as the zero-forcing equalizer and requires no matrix inversion such as the Least Square (LS) equalizer. For proof of concept the entire datastream is used as training data, which provides the best results of any given channel for an adaptive equalizer, since the maximum knowledge is gained about the channel for each frame received. The results below show the BER as the signals pass over one another in frequency, similar to the previous evaluation using traditional Spectral Subtraction.

INSERT FIGURE OF LMS EQUALIZER APPROACH

As you can see the figure above, the equalizer approach doesn't provide any improve beyond the traditional Spectral Subtraction approach. The problem with using traditional adaptive equalizers is that they can only be used with a comparative slowly fading channel. Since knowledge learned from the training data can be applied at the earliest to the next frame, if the interference changes enough it can render the equalizer useless. This rapidly changing spectrum or energy within the spectrum is unforunately a common characteristic of jammers. Even though this approach failed it provided an important observation and incite into the requirements and scenarios in which jammers can be overcome. For the sake of completeness an additional test was done with a small repeating sequence, smaller than the equalizer tap size, and as aspected the equalizer was able to overcome the interferer.

INSERT FIGURE OF SMALL SEQUENCE OVERCOME BY EQUALIZER

The important conclusion drawn from the previous experiment is that the when signals are orthogonal the receiver needs to be able to predict what data or energy is being transmismitted at a given time. Therefore the jammer problem must be constained future. As a result two jammer scenarios will be defined. The first scenarios is that the jammer's modulated data or energy is completely known to the receiver and the second is that the data sequence repeats with period being small. The larger the period the more resources the receiver will need to devote to its determination and evaluation. The sequence being completely known to the jammer is a reasonable assumption; primarily if the jammer is

friendly, as discussed previously in this thesis, then that knowledge can be readily available.

Now that the jammer scenarios have been defined further they can be evaluated. The first will be when the data sequence of the jammer is completely known to the receiver. The approach here will be to synchronize with the interfering signal, so the interferer will simply be subtracted off. To synchronize the signals a mathematical tool called correlation will be used. Correlation is a common tool used in synchronization in communication systems when looking for known symbols in a stream of data. The equation for correlation, shown below, simply passes signals over one another, the resulting sequence creates peaks where the data is most correlated.

$$(f \star g)[n] = summation f^{\star}[m] g[n+m]$$

NEED CITATION

An example of two sequences being cross-correlated with one another can be seen below, with the peak being where the signal line up of share the most mutual information with one another. Therefore from this data the location of the start of the interfers data can easily be located and removed. A simulation was created with this design in mind, with a unique result. Since the signals are frequency shifted over one another, when there frequencies match, it produces the best result, but as soon as they are offset, errors start to occur. This can easily be compensated for using a complex exponential multiplied by either the received or catalogued waveform. This will enduce a frequency shift cancelling out the shifting signal.

INSERT FIGURE OF known signal remove

This simulation was also repeated but this time was subtracted in the frequency domain. The result produced near identical results. Since using this approach requires two fourier transforms, into and out of the frequency domain, computationally it is much more involved and requires many more resources than the time domain approach. If the data was

already the frequency domain from some other process or signal processing technique then the frequency subtraction would be a viable option.

Now that a viable subtraction technique has been determined, the final implementation for the Spectral Subtraction block can be realized. As discussed in the Hardware and Software Platform section of this thesis, GNU Radio was the first to be examined because of its realtime attributes. This was quite an involved process requiring many weeks of trial and error. The first implementation was entirely written in C++, which is the recommended language for signal processing blocks in GNU Radio.

Since C++ within GNU differs from many modern programming styles a code implementation or route was taken to ensure accuracy and speed up development. Therefore all coding was done with C++ itself, using no GNU Radio built-in libraries[**?**]. To allow for matrix operations the Aramdillo C++ Library[**?**] was imported and provided needed vector operations such as correlation and faster mathematical functions instead of having to rewrite common search operations. This library would also be needed for the Signal Separation block, therefore coding with Armadillo would provide the knowledge for future implementations needed in that block. From the standard C++ implementation results were compared with Matlab, and the code was ported into GNU Radio.

Standard C++ -¿ GNU Radio Simulations -¿ GNU Radio Hardware Final

GNU Radio C++ are basically written by first creating test cases and writing your code until they are solved. This is a common practice among the programming community and provides a definitive endpoint to the code itself. The code was written and compiled successfully but unforunately the python wrapper called SWIG[**?**], which GNU Radio uses to interact with the C++ block through python, was unable to export the library. This is an undocumented problem within GNU Radio and was only identified through discussions directly with the GNU Radio core development team. Therefore another approach had to be considered.

The next option was to use python itself for signal processing. This is approach was primarily developed by Josh Blum, one of the core developer of GNU Radio. It isn't recommended due to speed issue, but it is quite easier to implement and debug for those familiar with python. As a result the previously C++ code was ported to python standards libraries and then to GNU Radio. The NumPy libraries were used within python. NumPy is the fundimental package for scientific computing in Python[**?**]. It like the armadillo library provides matrix operations such as correlation. Again under the Python standard libraries with NumPy the results were varified with MATLAB. Then the code was port to GNU Radio.

Again more problems occured, stonewalling all progress. The signal processing block was written as a subprocess using the queuing system built into GNU Radio. Queueing provides barriers between the connected blocks; therefore they can run freely, limiting bottlenecks. The system built would operate correctly for several hundred samples but would eventually segmentation fault. Several attempts to fix this error with even architectural changes to the code. Finally the lead developer of GNU Radio was consulted, Tom Rondeau [**?**], but he was also unable to determine a solution. The assumed problem was a type casting occuring within the queue itself, that would eventually accumulate and cause a segmentation fault.

With these setbacks, it became necessary to look beyond GNU Radio and just utilize MATLAB for signal processing. Therefore the decision to load captured signals from GNU Radio and process them in MATLAB. It is nowhere near realtime, but it will process the data appropriately. The simple GNU Radio needs frontend is important because it allows tight synchronization between multiple receive antenna, which is a require of the original design of the system. The GNU Radio model can be seen in the figure below.

GNU RADIO RECEIVE MODEL from GRC

3.4.2 Non-deterministic Scenarios

For completeness it is important to discuss the scenarios when the interferer's modulated data is unknown but repetative with a small period. The approach to estimating short sequences is a rather obvious one, an autoregressive algorithm is used to predict samples. The simulation here, which was just used for proof of concept, uses a linear predictive filter. The filter determines coefficients of a forward linear predictor by minimizing the prediction error in the least squares sense[**?**]. It finds the coefficients of a pth-order linear predictor (FIR filter) that predicts the current value of the real-valued time series x based on past samples.

$$\hat{x}[n] = -a(2)x(n-1) - a(3)x(n-2) - ... - a(p+1)x(n-p)$$

For the linear predictive filter to operate efficently the number of filter taps must be equal to or greater than the period of the repeated sequence. If the number of taps is smaller it cannot capture the randomness the in the interferer's data.

3.4.3 Over the Air Implementation Considerations

When moving towards a real implementation of the Spectral Subtraction block, the non-idealities introduced by the envirorment needed to be considered. These include frequency and phase shifts, as well as timing offsets. Certain considerations needed to be made as well, since instanteous changes occur when signals overlap. Therefore a more advanced control scheme needed to be constructed around the common signal compensation or correction. The basic idea used here is a receiver within a receiver, one for each signal received. This will be discussed in detail.

The system assumes that the jammer is always present within the envirorment therefore it was concluded that the jamming signal should synchronize with first, be removed and then all that remains should be the wanted signal. There is were the receiver within a receiver design comes in, since first the interferer will be synchronized to, utilizing phase

and frequency recovery and then timing recovery. Unforunately such an implementation isnt as straight forward as expected. Since when both signal are present in the spectrum it is impossible for these algorithm to operate correctly, therefore modifications need to made, which is where a controlling mechanism comes into play.

When multiple signals are in the envirorment the compensation algorithm learn incorrectly; as a result, a decision was made to pause these algorithm when both signals were present and continue when the signal interferer was only present. This operation relys on two assumptions, the first is that both signals are present for short periods of time which can be controlled. The second is that the calculated offsets of cause by the envirorment don't rapidly vary during the periods of time for which the two signals are visible. This assumption is quite reasonable especially with non-mobile transceivers. CONSIDER ADDING A SOURCE FOR EXAMPLE.

To accomplish this algorithm holding mechanism, energy detection was chosen to be uses for its simplicity. Below you can see an image of the jammer signal by itself and the combined signals. A large increase in energy or step can be seen, which can easily numerically detected. Energy is calculated using the following equation: $E_s = |x(t)|^2 dt$. This was implemented with a moving average filter with a small window to reduce sperious changes due to signals gaps or outliers. In practice the average peak energy level of the interferer is first calculated, then when it increase to a level roughly 1.5x that level the compensation algorithm are trigured. A simple technqiue technique, which is commonly uses but in the inverse fashion. The final results of the Spectral Subtraction block are examined in the next chapter.

3.5 Superimposed Equalizer

Moving on, the next block to discuss is the signal separation block. The development of this block is very staggered and due to time requirements shortcuts needed to be made.

The original desired result was to use a blind source separation technique outlined here [**?**], which is able to separate multiple signals from one another under specific constraints. For this process to work efficently an appropriate channel model needs to be created. Since the goal of this system is to be very robust an estimator for very frequency selective channel is desired. The progession of the signal separation block will be examined in this sections and the limitations will be discussed.

The first objective is to examine the channel mixing model which assumes a single-input multiple-antenna broadcast channel. A J-channel FIR system excited by K transmit antennas is considered. A quasi time-invariant multipath channel is assumed which remains constant during the transmission of a set of consecutive symbols, which are called slots. These slots are assumed independent from one another. The channel estimation is performed over each slot. Each symbol inside a slot is assumed to be the result of a known redundant precoder acting on an input transmit symbol vector drawn from an M-PSK constellation. Therefore the receiver receives signals not only from the intended transmitter but also from Q other interferers. The interferers are assumed to employ the same redundant precoder as the desired signal[**?**].

With this model, [**?**] shows that at no assumptions are necessary regarding the number of the transmit antennas of each of the interferers and the channel orders as long as they are smaller than the block size. The block size in this case is equal to the combination of the individual channel lengths and the number of transmit antenna used by the desired transmitter. But this evaluation rely on three assumptions:

1. The data sequence $\mathrm{x}_d isani.i.d. sequencesuchthat x_d\ CN(0, \sigma_d^2$

2. The distribution over the MIMO channel vector is $p(y;\theta)\ CN(\mu_y, R_w)$, and the interference vector is distrbuted normal with covariance R_w

3. The transmitted symbols, the channel vector, and the interference vector are jointly independent

With these assumptions the mixing process can be undone but the channel estimation

needs to be calculated first. Since this is a MIMO channel, frequency selective fading will need to be captured to providing appropriate channel knowledge. To accomplish list, using a superimposed equalizer was chosen, whose operation was heavily discussed in the background section of this thesis. In summary, [**?**] uses a superimposed symbol transmission scheme to estimate frequency-selective channels. Several points of the DFT of the data are set to known values. This operation can be easily implemented in the time domain when these DFT points are equispaced. The channel is estimated using the DFT of the received signal at these selected DFT points. The detection itself is done using an iterative method across these points. Unlike traditional equalizers, the proposed method does not require bandwidth for training. It instead trades spectral power for those symbols themselves, spreading its energy over the entire bandwidth capturing the entire spectrum space. [**?**] also proves that by placing the training symbols in quasi-periodic position they will not interfer with the data itself.

With our channel estimation method chosen, a simulation was created to prove the effectiveness of such a scheme. The result of this implementation were directly compared with the results of the paper to prove correctness. The results of this simulation are seen below. It examines a random frequency selective channel, with very high suppression with a across a number of SNR values. As expected there is a linear relationship between SNR. For completeness a comparision was done with a traditional LMS equalizer to show the effectiveness of the superimpose equalizer.

INSERT Figure of SNR and channel estimate for Superimpose equalizer

INSERT Figure with comparision of superimposed equalizer vs. traditional linear equalizer

From the comparision of the figure above you can see in a frequency selective fading channel the traditional LMS equalizer operates poorly. These simulations provide the necessary foundation to push towards the final implementation of the signal separation block.

Due to time constraints the certain decisions needed to be made about this block and future work will be need to complete the overall goal and desired performance of the block. With that in mind the primary goal for the signal separation block in this thesis is to provide an accurate channel estimate. The unmixing model requires this knowledge to work appropriately. Therefore instead of the proposed AMUSE[**?**] technique, another MIMO cross-channel technique called Maxmimal Ratio Combining (MRC). This decision was made with both project managers.

Maximal Ratio Combining is a method of diversity combining in which the signal add first weighted and then added together. These weights or gains are made proportional to the RMS signal level and are inversely proportional to the mean square noise level in that channel, and the same proportionality constant is used for all channels[**?**]. Therefore the channel with the best SNR provides the greatest impact on the resulting sequence. This process needs to explain more throughly. Assuming the received signal is an array of samples received the individual antennas $\mathbf{x}(t) = \mathbf{h}(t)u(t) + \mathbf{n}(t)$ and the individual channels $\mathbf{h} = [h_0, h_1, ..., h_{N-1}]^T$, and the additive noise $\mathbf{n} = [n_0, n_1, ..., n_{N-1}^T$. The equalized symbol $\hat{x} = x + (h^H n)/(h^H h)$ [rsadve/Notes/DiversityReceive.pdf**?**].

A simple evaluation of MRC was done to prove its capabilities, which is based on the simulations here [**?**].

INSERT MRC PLOT OF INCREASING ANTENNAs

As you can see as you increase the number of antennas in a frequency selective channel the better the result. Therefore MRC was introduce into the framework of the signal separation block, and the new model for this block can be seen below. This block first utlizes the superimposed equalizer on the channels individually then uses MRC to combine there results maximizing the SNR of the desired signal. MRC was combined with the channel estimate approach using superimposed equalizers. From the knowledge learned in the previous with the implementation involving GNU Radio, the signal separation block was created entirely in MATLAB. The results of this operation will be discussed in the final chapter of the thesis.

3.6 Antenna Subset Selection

3.7 Summary