

2011

Compressed sensing approach to ultra-wideband receiver design

Jodhbir Singh
Michigan Technological University

Follow this and additional works at: <https://digitalcommons.mtu.edu/etds>


 Part of the [Electrical and Computer Engineering Commons](#)

Copyright 2011 Jodhbir Singh

Recommended Citation

Singh, Jodhbir, "Compressed sensing approach to ultra-wideband receiver design", Master's Thesis, Michigan Technological University, 2011.
<https://doi.org/10.37099/mtu.dc.etds/49>

Follow this and additional works at: <https://digitalcommons.mtu.edu/etds>

 Part of the [Electrical and Computer Engineering Commons](#)

A COMPRESSED SENSING APPROACH TO ULTRA-WIDEBAND RECEIVER
DESIGN

By

Jodhbir Singh

A THESIS

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

(Electrical Engineering)

MICHIGAN TECHNOLOGICAL UNIVERSITY

2011

© 2011 Jodhbir Singh

This thesis, "A COMPRESSED SENSING APPROACH TO ULTRA-WIDEBAND RECEIVER DESIGN," is hereby approved in partial fulfillment of the requirements for the Degree of MASTER OF SCIENCE IN ELECTRICAL ENGINEERING.

Department of Electrical and Computer Engineering

Signatures:

Thesis Advisor:

Dr. Zhi (Gerry) Tian

Department Chair:

Dr. Daniel R. Fuhrmann

Date:

To my parents and my education

Table of Contents

Table of Contents.....	4
List of Figures.....	6
List of Tables.....	7
Acknowledgements.....	8
Abstract.....	9
1. Introduction.....	10
1.1. Ultra-Wideband Receiver Literature Review.....	12
1.2. Motivation.....	15
1.3. Thesis Outline.....	15
2. Compressed Sensing Based UWB.....	17
2.1. UWB Signal Model.....	17
2.2. Introduction to Compressed Sensing.....	19
2.3. UWB Demodulation using Compressed Sensing.....	23
2.4. CS-UWB Receiver Design.....	24
2.4.1. CS-Based Channel Estimation.....	25
2.4.2. Signal Demodulation (Proposed).....	30
3. CS-UWB Implementation.....	33
3.1. Study of CS Hardware Implementations.....	33
3.2. Other Implementation Challenges.....	37
3.3. Problem Formulation.....	38
3.4. Proposed Method.....	40

4. Results.....	43
4.1. Simulation Setup.....	43
4.2. Performance Comparisons.....	44
4.3. Hardware Comparison	47
5. Conclusion and Future Work.....	49
References.....	50
Appendices.....	53

List of Figures

Figure 1.1: UWB and existing narrowband spectrum	11
Figure 2.1: UWB symbol structure.....	17
Figure 2.2: Block diagram of CS-UWB receiver design	26
Figure 2.3: CS based channel-template estimation.....	27
Figure 2.4: Matrix representation equation 14 of CS-UWB for a frame	28
Figure 2.5: CS-UWB analog demodulation block diagram.....	31
Figure 2.6: CS-UWB digital demodulation block diagram	32
Figure 3.1: Random filter implementation block diagram.....	34
Figure 3.2: Random sampling based compressed sensing.....	35
Figure 3.3: Random demodulation based compressed sensing	36
Figure 3.4: Receiver structure based on Distributed Amplifiers	36
Figure 3.5: Compressed Sensing matrix representation	38
Figure 3.6: Hardware implementation of CS-UWB	39
Figure 4.1: Performance graph for compression ratio less than 35%.....	45
Figure 4.2: Performance graph for compression ratio more than 35%.....	46
Figure 4.3: Performance graph for different number frames in each symbol.....	47

List of Tables

Table 4.1: Comparison of hardware complexity and performance in different compression ratios	48
---	----

Acknowledgements

I would like to thank Dr. Zhi(Gerry) Tian for being my advisor for my MS. I gained immensely from her to improve my research skills, problem solving, and presentation skills. Most importantly, her open door walk-in policy makes her very approachable.

Many thanks to Suryabh Sharma, who shared office with me, for his keen interest in my research topic which helped me to explain it to him and in turn learn. Many thanks go to Wang Yue for his time and patience to listen to my doubts.

I would also like to thank Dr. Chunxiao (Tricia) Chigan and Dr. Aurenice Oliveira for becoming members of my committee.

At last, I am very thankful to my friends and family for all the support and help they provided me during the course of my study.

Abstract

One of the scarcest resources in the wireless communication system is the limited frequency spectrum. Many wireless communication systems are hindered by the bandwidth limitation and are not able to provide high speed communication. However, Ultra-wideband (UWB) communication promises a high speed communication because of its very wide bandwidth of 7.5GHz (3.1GHz-10.6GHz). The unprecedented bandwidth promises many advantages for the 21st century wireless communication system.

However, UWB has many hardware challenges, such as a very high speed sampling rate requirement for analog to digital conversion, channel estimation, and implementation challenges. In this thesis, a new method is proposed using compressed sensing (CS), a mathematical concept of sub-Nyquist rate sampling, to reduce the hardware complexity of the system. The method takes advantage of the unique signal structure of the UWB symbol. Also, a new digital implementation method for CS based UWB is proposed. Lastly, a comparative study is done of the CS-UWB hardware implementation methods.

Simulation results show that the application of compressed sensing using the proposed method significantly reduces the number of hardware complexity compared to the conventional method of using compressed sensing based UWB receiver.

1.Introduction

In any wireless communication system, one of the most desirable features users want is high speed data transfer. However, most of the current wireless systems are limited mainly by their respective bandwidths. The Ultra-wideband (UWB) wireless communication system is a promising technology with a very wide bandwidth for short distance high data rate communication, and long distance low data rate communication. High data rate, low cost, and low power consumption are some of the potential advantages of the UWB compared to the conventional narrowband wireless systems[1-3].

In 2002, the U.S. Federal Communication Commission (FCC) unlicensed a very wide bandwidth of 7.5GHz, which spans from 3.1GHz to 10.6GHz for Ultra-wideband communication[3]. However, the FCC has put strict power limitation of -41.3dBm/Hz so as to keep it at the noise floor of the existing radio frequency (RF) systems because the UWB overlaps with existing narrowband systems. According to the FCC, any radio frequency (RF) transmission with fractional bandwidth more than 20% or system bandwidth of more than 500MHz is characterized as Ultra-Wideband. (The fractional bandwidth is defined as B/F_c , where $B = (F_h - F_l)/2$, denote -10dB bandwidth and center frequency $F_c = (F_h + F_l)/2$, with F_h being the upper frequency of -10dB emission point, and F_l the lower frequency of the -10dB emission point[1])

UWB has many advantages over existing narrowband wireless systems. For example, the large bandwidth provides very high channel capacity which provides high data rate; baseband transmission lowers the system complexity by eliminating the need of modulators, intermediate frequency, and other receiver processing circuits; the ultra-short duration pulses results in rich multipath diversity; and the ultra-short duration of UWB pulse provides high precision ranging and localization at centimeter level.

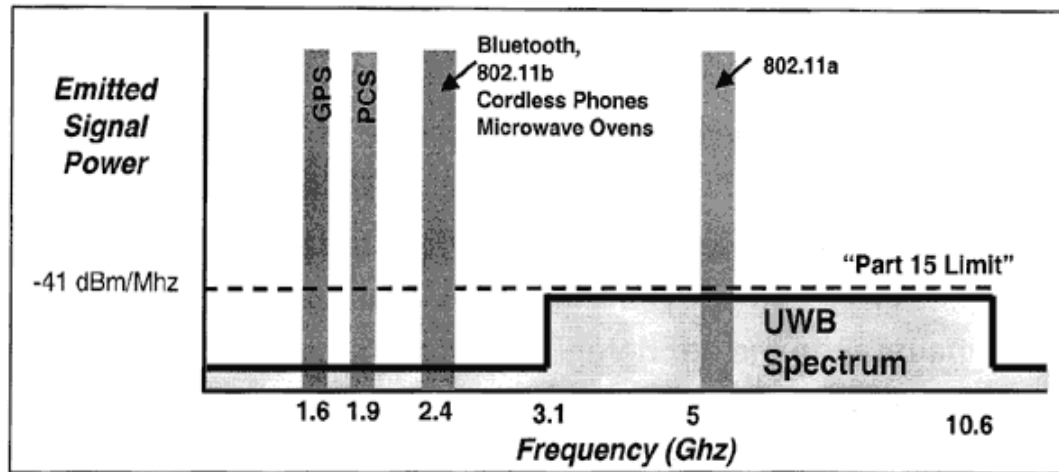


Figure 1.1: UWB and existing narrowband spectrum [4] (Appendix D)

In addition to these advantages, UWB also faces many technical challenges mainly in the receiver design. The formidably high bandwidth of 7.5GHz of UWB requires a very high sampling rate according to the Nyquist-Shannon theorem.

Analog-to-Digital (ADC) converters at such high frequencies are very expensive and power consuming. The UWB channel results in many multipath components. Thus, the rake receiver is an optimal receiver to capture the maximum energy of the received signal; however, it requires large numbers of correlation fingers, which increases the hardware complexity of the receiver. Additionally, there have been non-coherent receivers mentioned in [5, 6] which significantly reduce the system complexity by avoiding the channel estimation, but they perform sub-optimally compared to the coherent receivers.

An alternative UWB receiver design has been proposed in [7-9] based on a mathematical concept which eliminates the problem of requirement of high speed ADCs for optimal receivers and performance degradation limitation of sub-optimal receivers. Using compressed sensing, a signal can be sampled at sub-Nyquist rate based on certain signal characteristics. In the case of UWB, it significantly reduces the sampling rate requirement of the UWB receiver.

However, there are some implementation challenges associated with compressed sensing based Ultra-wideband receivers. Most importantly, it requires a large number of

parallel structures of correlation and integration fingers to collect those compressed samples.

In this thesis, a new method- Joint Frame Reconstruction- has been proposed for reducing the hardware complexity of the conventional compressed sensing based ultra-wideband receiver (CS-UWB). The proposed method reduces the receiver complexity (number of parallel correlation-integration fingers) by exploiting the unique UWB signal structure. Moreover, a study of hardware implementation of compressed sensing has been done, and a new digital implementation method has been proposed for CS-UWB implementation.

1.1. Ultra-Wideband Receiver Literature Review

The Impulse Radio Ultra-wideband (IR-UWB) signal is transmitted at the baseband level which eliminates the need for a carrier modulator at the transmitter and a carrier demodulator at the receiver, thus simplifying the transceiver design. A Gaussian monocycle or its derivative pulse of nanosecond duration T_w is transmitted over the UWB channel.

The ultra-short duration pulse results in many resolvable multipath components at the receiver. Based on the trade-off between performance and hardware complexity, UWB receivers can be categorized into two types - Coherent receivers and Noncoherent receivers. In coherent receivers, the most common UWB receiver is the correlation (matched filter) based rake receiver which collects the multipath diversity of the received UWB signal[1, 3, 4].

A matched filter based rake receiver captures the energy of the multipath components in a received UWB signal. Rake receiver requires channel state information for demodulating the UWB signal. An estimated pulse waveform template is correlated with the received multipath signal at certain channel taps with corresponding channel coefficient. Thus, an essential criterion for rake receiver based UWB is estimating channel

information [1, 3, 5]. However, the sampling rate requirement for such channel estimation is very large. For a typical UWB indoor channel, the number of channel parameters can be as large as 400[1, 3, 10, 11].

The digital implementation of the coherent UWB receiver requires sampling of the received signal at or above the twice of highest frequency component of the signal. The sampling rate requirement for the UWB signal with a pulse width of $T_p = 0.7ns$ is in the formidable range of 17.9~35.7GHz[1]. The sampling rate requirement for accurate channels estimation requires more than 25GHz of ADC in [12]. ADCs of such high sampling rate require high power and are very expensive. Such sampling rate can only be possible using either wideband ADCs or a bank of polyphase ADCs with accurate timing control [8]. Also, the sub-nanosecond multipath received pulses add another challenging task for timing synchronization at the receiver. More importantly, a large number of resolvable multipath components through the UWB channel require a rake receiver with large number of correlation fingers to capture the rich multipath diversity.

Noncoherent receivers, on the other hand, provide an alternative to surpass high sampling rate processing of the coherent receivers[6, 11, 12]. They do not require expensive channel estimation method. An Energy detector (ED) and Autocorrelation receiver (AcR) are examples of the most common noncoherent UWB receivers. Energy detector consists of a squaring device which uses On-Off keying signaling scheme to measure the instantaneous energy of the signal. On the other hand, an Autocorrelation receiver consists of an additional delay element before the squaring operation. It uses Transmitted Reference (TR) signaling, in which a reference pulse is transmitted before a modulated pulse in the same symbol. At the receiver, the delayed reference pulse is used as a template for demodulating the modulated pulse.

Noncoherent receivers are simple in both operation and implementation compared to coherent receivers. However, the major limitation of noncoherent receivers is their degraded performance. They result in suboptimal performance compared to the coherent receivers[6]. Additionally, Auto-correlation receivers require long analog delay lines which are difficult to implement in hardware. Weighted energy detection is an improved

version of energy detector which divides the signal into smaller segments. It provides better performance, but it comes at the cost of additional hardware complexity[11].

Compressed Sensing (CS), a new technique to sample a certain class of signals below the Nyquist rate criterion and successfully reconstruct it, has gained application in various scientific fields in recent years. In the case of the UWB, it promises to solve the major problem of high sampling rate requirement. The applicability of compressed sensing technique to UWB has been shown in [7, 13].

Compressed Sensing, itself, is a nascent mathematical concept that goes against the conventional wisdom in data acquisition [14, 15]; therefore, there lies many implementation challenges for the compressed sensing based UWB (CS-UWB) receiver. In compressed Sensing, M signal measurements are taken of a sparse signal of length N where $M < N$. The signal is recovered from these fewer measurements M using a convex optimization algorithm [14-16].

In case of UWB, the number M can be very large which means M mixer-integrator parallel branches are required for to acquire measurements in hardware. However, a less complex compressed sensing based UWB receiver has been proposed in[8], which transfers the receiver complexity to the transmitter end. In[8] , transmitter and UWB channel are part of the compressed sensing process, such that the receiver simply down samples the received signal. There have been other compressed sensing based UWB (CS-UWB) implementation methods proposed in [17-19].

In this thesis we propose a new joint frame reconstruction method for reducing the hardware complexity of compressed sensing based UWB (CS-UWB) receiver by further reducing M , the number of mixer-integrator branches for acquiring measurements, than the conventional CS-UWB method. A digital receiver design is also proposed for the implementation of compressed sensing based UWB receiver. At last, a study is done of hardware implementation of compressed sensing based UWB receivers.

1.2. Motivation

The motivation of this thesis stems from the fact that the existing UWB receivers require very high sampling rate for analog to digital conversion (ADC). Such high-speed ADCs are power consuming, high in hardware complexity, and expensive compared to low rate ADCs. The alternative receivers to high-speed ADCs, though simple in hardware implementation, result in degraded performance. Thus, it is important to find a new approach for UWB receiver that not only provides optimal performance, also keeps the sampling rate below the Nyquist sampling rate.

Compressed Sensing is a new mathematical concept, which provides an alternative to existing UWB receivers which require high-sampling rate. In compressed sensing, a signal can be sampled below the Nyquist rate and successfully reconstructed based on certain criteria. Thus, UWB fulfils the compressed sensing criteria. However, compressed sensing based UWB receiver has its own challenges. One of the major challenges is the hardware implementation of compressed sensing for continuous-time domain signal. Thus, this thesis investigates the possible solution for reducing the hardware complexity of the compressed sensing based UWB receiver without compromising the performance of the system.

1.3. Thesis Outline

The thesis outline starts with the introduction to the UWB technology. It briefly describes the place of UWB in the wireless spectrum, its advantages, challenges, and potential applications. Rest of the thesis is divided mainly into five chapters which are discussed below:

Chapter 1: The first chapter is an introduction to the UWB wireless system. It provides an overview of UWB receiver's types with their working method, advantages, and challenges. The motivation for the thesis is also mentioned in the same chapter.

Chapter 2: This chapter provides detailed description about the UWB and compressed sensing. First, a UWB signal model is described. Second, a brief introduction to a new framework, compressed sensing, is provided. Further, it describes the applicability of compressed sensing to UWB system. In the last section, CS-UWB receiver design, a comprehensive compressed sensing based UWB system model is described. A new digital demodulation receiver design is also proposed.

Chapter 3: In this chapter, a study of CS-UWB hardware implementation method is provided. The study compares the working, advantages, and challenges of various compressed sensing hardware implementation methods. The problem is formulated for this thesis and a new method is proposed to solve it.

Chapter 4: In this chapter, all the simulation results are discussed. Matlab simulation setup is explained for the different comparison results. The proposed method is compared with an existing CS-UWB method. Matlab simulations show improvement of the proposed method over existing method. A hardware reduction comparison is made for various cases simulation.

Chapter 5: The thesis is concluded with a brief summary of thesis, contributions, and future work.

2. Compressed Sensing Based UWB

This chapter describes the ultra-wideband and compressed sensing in details including their fundamentals. It provides the explanation for the application of compressed sensing to the ultra-wideband. Moreover, in the later sections, two CS- UWB implementation models are described.

2.1. UWB Signal Model

The Impulse Radio-Ultra-wideband (IR-UWB) signal has a different symbol structure with respect to the narrowband symbol structure[3]. In the IR-UWB system, a symbol consists of number of frames (N_f), each of duration T_f . Each frame includes a pulse of very short duration T_w such that $T_w \ll T_f$. The duration of a symbol is then given by $T_s = N_f T_f$. Figure 2.1 shows the structure of a UWB symbol with multiple frames[4].

Consider a peer-to-peer IR-UWB transmitted signal $s(t)$ which can be represented as:

$$s(t) = \sum_{i=0}^{N_p-1} b_i \sum_{j=0}^{N_f-1} p(t - iT_s - jT_f) + \sum_{i=N_p}^{N_w-1} b_i \sum_{j=0}^{N_f-1} p(t - iT_s - jT_f)$$

$$0 \leq t < N_w T_s$$

.....(1)

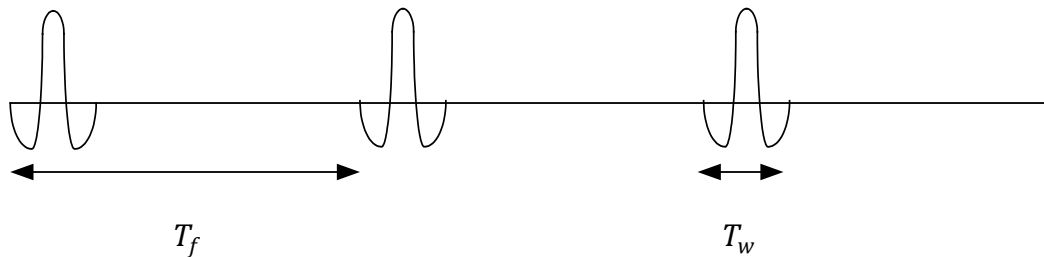


Figure 2.1: UWB symbol structure[4]

Where $p(t)$ is a Gaussian pulse derivative[1, 20]. $b(\cdot)$ is the symbol modulation. In this thesis we use binary PAM modulation i.e. $b_i \in \{\pm 1\}$. N_p is the number of pilot symbols, and N_s is the number of information bearing symbols. $N_s = N_w - N_p$ is the total number of symbols in a burst. $T_s = N_f T_f$ is the duration of one symbol.

The UWB channel has different characteristics from the narrowband system's channel which operates at signal bandwidth of less than 20MHz[21]. Multiple resolvable multipath components are received at the UWB receiver due to its large bandwidth. In the case of the indoor UWB channel, the multipath components arrive in clusters and each cluster has multiple rays (or delayed UWB received waveforms). The arrival of clusters and the arrival of rays in each cluster shows Poisson distribution[1]. The design of UWB receiver systems for optimum performance requires accurate channel modeling[4]. According to the channel modeling subcommittee of IEEE 802.15.3a, the standardized UWB channel model is a modified version of the Saleh-Valenzuela(S-V) model[20].

However, a simple model for the characterization of the UWB channel is a tap-delay-line fading model. The impulse response of the channel can be written as

$$h(t) = \sum_{l=0}^{L-1} \alpha_l \delta(t - \tau_l)$$

.....(2)

Where α_l represents the channel gain coefficient and τ_l is the channel delay for the l_{th} path of the total L multipath components. When the transmitted signal $s(t)$ passes through $h(t)$, it results in multipath components.

The received multipath signal at the receiver can be written as follows

$$r(t) = \sum_{i=0}^{N_p-1} b_i \sum_{j=0}^{N_f-1} p_h(t - iT_s - jT_f) + \sum_{i=N_p}^{N_s-1} b_i \sum_{j=0}^{N_f-1} p_h(t - iT_s - jT_f) + n(t)$$

.....(3)

Pilot Symbols + Information Symbols

Where $p_h(t) = \sum_{l=0}^{L-1} \alpha_l p(t - \tau_l)$ and it is referred as noiseless composite pulse-multipath channel[22]. It is of frame long duration T_f . $n(t)$ is the Additive White Gaussian Noise (AWGN). UWB channel $h(t)$ is considered invariant over the duration of one burst of symbols. For simplicity, it is assumed that IFI (Inter-Frame Interference) is absent, therefore, $T_f \geq T_p + \tau_{L-1}$ where τ_{L-1} is the maximum delay spread of the UWB multipath channel. Moreover, Inter-symbol Interference (ISI) is assumed to be absent in the system model for the sake of simplicity, also, with wide frame duration the ISI can be assumed to be very small in certain environments.

Receiver processing is illustrated in later chapters in which data demodulation is dependent on channel estimation using compressed sensing. Thus, the next section briefly explains the concept of compressed sensing, and leads to a compressed sensing based UWB receiver.

2.2. Introduction to Compressed Sensing

In all conventional signal processing applications, sampling theorem (or Nyquist-Shannon's theorem) plays a vital role in the analog to digital conversion of signals. According to Nyquist-Shannon's theorem, to acquire any analog signal, the sampling rate must be at least twice the maximum frequency component of the signal for its successful reconstruction. The key note to make here is that the Nyquist rate is so high in various electronics applications ranging from image processing to video capturing that compression of data becomes necessity before transmission or digital storage[16]. However, according to compressed sensing theory, it is possible to successfully reconstruct certain signals from a fewer number of samples based on certain criteria.

Compressed Sensing is a method in which fewer numbers of samples are acquired from a sparse signal along its length by taking non-adaptive linear measurements which preserves the structure of the signal, and then using a numerical optimization method, the original signal is reconstructed from the fewer number of measurements[14-16].

There are two main criteria in compressed sensing for the successful reconstruction of underdetermined signal[14].

- i. *Sparsity*, which is the property of the signal.
- ii. *Incoherence*, a property between representation basis and sensing basis.

Sparsity of a signal implies that the signal has few non-zero components in some basis. The information rate of the signal is lower than its bandwidth. According to CS theory, many natural signals are sparse in some domain or over some basis. For example, a digital image when acquired using Nyquist rate sampling has many samples, but even after throwing many small valued samples (negligible coefficient values) by compression techniques, the image reasonably preserves its structure[16]. Another similar example is cognitive radio which works on the principle that at any given time there are fewer number of frequency bands occupied in the wireless spectrum, making the signal sparse in the frequency domain. Mathematically, a signal \mathbf{x} , can be represented as a sparse signal when expanded on an orthonormal basis $\Psi = [\psi_1 \psi_2 \psi_3 \dots \dots \psi_N]$,

$$\mathbf{x} = \sum_{n=1}^N \alpha_n \psi_n \dots \dots \dots (4)$$

Where α is the coefficient vector of \mathbf{x} , $\alpha_n = \langle \mathbf{x}, \psi_n \rangle$. \mathbf{x} and α are the representations of the same signal in different basis.

Incoherence is the property which must be satisfied between the measurement matrix Φ and the signal's representation matrix Ψ . The Φ takes fewer linear measurements over the length of the sparse signal. Therefore, each projected sample has information over the complete length of the sparse signal. The purpose of incoherence the between measurement matrix and the representation basis of the signal is to spread the signal in the acquired domain. As an example, a spike or delta basis as a measurement matrix and a Fourier basis as a representation basis have maximum incoherence between each other[14].

To explain the concept of compressed sensing mathematically, consider real valued, finite-length, discrete signal, one-dimensional \mathbf{x} , where $\mathbf{x} \in \mathbb{R}^N$. The \mathbf{x} can be considered a $N \times 1$ vector which is sparse in some basis Ψ , where $\Psi = [\psi_1 \ \psi_2 \ \dots \ \psi_N]$, each ψ_i is a column vector of length N . Thus, the signal \mathbf{x} can be written as [16]

$$\mathbf{x} = \Psi \boldsymbol{\alpha} \tag{5}$$

Where $\boldsymbol{\alpha}$ is a $N \times 1$ vector which has few nonzero elements, and \mathbf{x} is simply the projection of vector $\boldsymbol{\alpha}$ over an orthonormal basis Ψ . Each element of \mathbf{x} , x_i is the inner product of the row of Ψ and column vector $\boldsymbol{\alpha}$. The sparsity of the signal \mathbf{x} is given by K , which is the number of nonzero elements in $\boldsymbol{\alpha}$.

Now to compress a signal, a measurement matrix Φ of dimension $M \times N$ is designed in such a way that each row vector of Φ is incoherent with each column vector of Ψ , a property which must be satisfied for reconstruction of vector $\boldsymbol{\alpha}$. It is important to note that here $K \ll M < N$. Thus, the projected output vector \mathbf{y} (compressed) is given by

$$\mathbf{y} = \Phi \mathbf{x} \tag{6}$$

Where $\mathbf{y} \in \mathbb{R}^M$ and $\mathbf{x} \in \mathbb{R}^N$. The main problem here is designing a stable non-adaptive sensing matrix Φ which must allow the reconstruction of $\boldsymbol{\alpha}$ of length N from \mathbf{y} of length M where $M < N$. Thus, equation (6) can be written as:

$$\mathbf{y} = \Phi \mathbf{x} = \Phi \Psi \boldsymbol{\alpha} = \Theta \boldsymbol{\alpha} \tag{7}$$

Where $\Theta = \Phi \Psi$ is a $M \times N$ matrix.

The above problem is ill conditioned. However, if \mathbf{x} is K sparse and $K \ll M < N$, then for any vector sharing the same K non-zero entries as α and for some $\epsilon > 0$, this problem can be solved if it satisfies the following condition [14]

$$1 - \epsilon \leq \frac{\|\Theta\alpha\|_2}{\|\alpha\|_2} \leq 1 + \epsilon \quad \dots\dots\dots(8)$$

Here the matrix Θ must satisfy condition (8), called *Restricted Isometry Property* (R.I.P.) for the stable solution of α .

The important task here is designing a stable the measurement matrix Φ that satisfies both the incoherence and R.I.P. property. One such matrix can be designed with high probability using a random matrix, such as the Gaussian random matrix.

According to [14, 16], if the elements of Φ are chosen from an independent and identically distributed (i.i.d.) Gaussian probability density function with zero mean and $1/N$ variance, then Φ has R.I.P. with high probability if $M \geq cK \log\left(\frac{N}{K}\right)$, where $c \geq 1$ is the oversampling factor.

The last task is the reconstruction of the signal \mathbf{x} from the reduced set of projections \mathbf{y} using an optimization process. The optimization problem in (7) can be solved by l_1 - norm optimization.

$$\hat{\alpha} = \arg \min \|\alpha\|_1 \quad \text{such that} \quad \Theta\alpha = \mathbf{y} \quad \dots\dots\dots(9)$$

The sparse solution of α can be found using linear programming algorithms such as basis pursuit (BP) or other greedy algorithms. In this thesis a greedy algorithm called orthogonal matching pursuit (OMP) is used [14, 23].

2.3. UWB Demodulation using Compressed Sensing

As mentioned earlier, the coherent receivers of UWB requires a formidable sampling rate which has high hardware complexity. Compressed sensing (CS), on the other hand, takes samples at the sub-Nyquist rate and reconstructs a signal from fewer numbers of measurements. Therefore, compressed sensing is well suitable for UWB which has hardware limitations with high Nyquist rate sampling.

Compressed sensing (CS) is a mathematical concept which has recently gained application in various fields of engineering. In the case of UWB, the applicability of compressed sensing is only possible if the UWB system fulfils the required criteria of compressed sensing i.e. sparsity of the signal, incoherence between the basis matrix and the measurement matrix, and Restricted Isometric Property (R.I.P.).

The feasibility of CS to the UWB system can be answered by the following two questions:

- a) How is compressed sensing applicable to the UWB?
- b) Compressed sensing works with the discrete signal, so how does it work with the analog UWB signal?

The answer to the first question can be found if the UWB channel characteristics are observed. A UWB pulse through a multipath channel results in multiple delay pulses of very short duration. The path length differentials can be down to about 30cm[7-9, 12, 22]. The received UWB signal results in time sparsity which can be approximated by few pulses from a continuous basis. It is the time sparsity of the received UWB signal that makes it an ideal candidate for compressed sensing application.

To answer the second question, an analog signal model is needed for compressed sensing. According to[13, 24], any analog signal with a finite information rate can be represented by a finite number of parameters over some continuous basis. Mathematically, let $x(t)$ be an analog signal which is sparse over a finite number of continuous basis, then it can be represented as

$$x(t) = \sum_{n=1}^N \alpha_n \psi_n(t) \tag{10}$$

Where α has few a nonzero component (or dominant components) which means $x(t)$ is sparse over continuous basis vectors $\psi_n(t)$, where $n = 1, 2, \dots, N$

The incoherence property and restricted isometry property of the system can be achieved with high probability if elements of measurement matrix Φ be selected as identically distributed (iid) Gaussian random as mentioned in the previous section[14].

2.4. CS-UWB Receiver Design

It was said earlier that the UWB receiver has a big challenge of analog to digital conversion at very high bandwidth. Therefore, compressed sensing is a promising technique to solve this problem. Consider that a simple correlator (matched filter) based UWB detector is used, then, all that is needed is a frame template for frame based demodulation and sampling.

The CS based UWB receiver can be described in two parts. First, pilot symbols are used to estimate the channel-template. Compressed sensing is used to reconstruct the multipath-frame template[7, 21]. Second, the reconstructed multipath-frame template is used to demodulate the information-bearing symbols at frame rate using integrate and dump circuit[22].

Let $r_p(t)$ be the pilot signal carrying N_p pilot symbols with N_f frames in each corresponding symbol. Thus, $r_p(t)$ can be represented by the following equation:

$$r_p(t) = \sum_{i=0}^{N_p-1} b_i \sum_{j=0}^{N_f-1} p_r(t - iT_s - jT_f) + n(t)$$

$$0 \leq t < N_p T_s$$

.....(11)

If $r_p(t)$ is observed in the time window of $kT_f \leq t < (k+1)T_f$, where $k = 0, 1, 2, \dots, N_f N_p - 1$, then each T_f duration contains multipath frames contaminated with AWG noise. It can be represented as[19]

$$p_k(t) = b_i \left(\left\lfloor \frac{k}{N_f} \right\rfloor \right) p_r(t) + n(t)$$

.....(12)

$p_k(t)$ is frame long multipath channel.

Receiver processing can be divided into two parts from here - channels estimation using compressed sensing, and signal demodulation using correlation detector. The following block diagram shows that the received pilot symbols are used for channel-template estimation using compressed sensing, and the upper and lower part of the block diagram show two types of demodulation schemes used for signal demodulation.

The output of channel-template estimation using CS is a discrete vector at Nyquist rate. It is approximated to analog channel-template for analog demodulation of the information bearing signal. However, for digital demodulation, a new receiver design method is proposed which down-samples the discrete channel-template and information-bearing signal at sub-Nyquist rate.

2.4.1. CS-Based Channel Estimation

The compressed sensing based channel estimation is done in two parts. As shown in the figure 2.2, first, using a number of mixer-integrator parallel branches, compressed samples are take at frame rate sampling. The mixer operates at or above the Nyquist

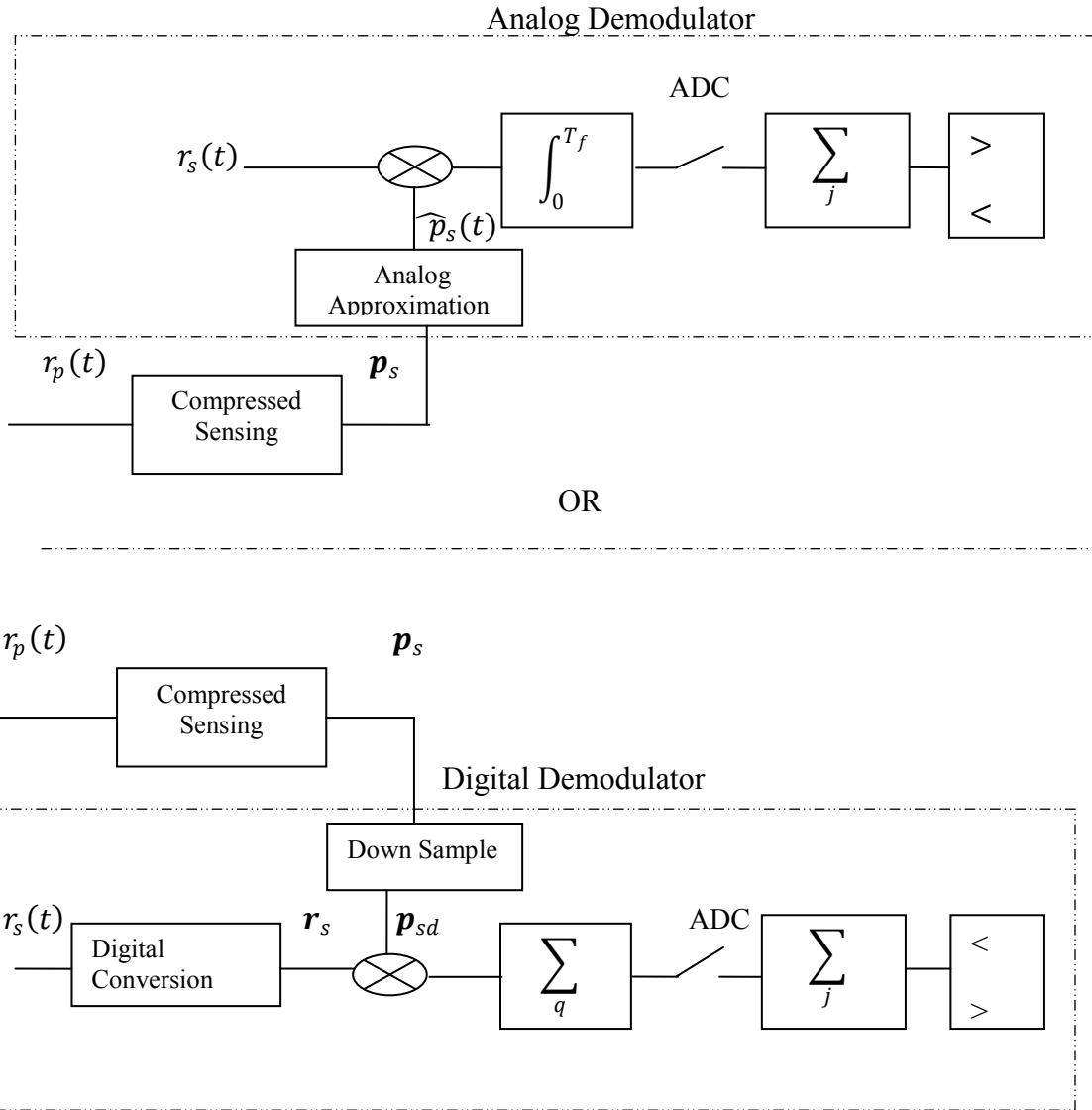


Figure 2.2: Block diagram of CS-UWB receiver design

frequency to provide randomness (The details of operation of mixer is explained in later chapter). Second, using convex optimization method, original multipath frame is reconstructed from fewer number of samples.

UWB channel shows time sparsity which means the representation (basis) matrix Ψ can be assumed to be an identity matrix[22]. The input to the compressed sensing block is

an analog multipath frame of each pilot symbol. The output of the reconstruction block is a discrete representation of analog multipath frame at Nyquist rate.

However, to explain the compressed sensing based channel estimation, consider \mathbf{p}_k the discrete time representation of the continuous multipath-frame in each pilot symbol which is obtained using mixer-integrator circuit in the compressed sampling block[22].

Thus, $\mathbf{p}_k = [p_k(0), p_k(T), \dots, p_k((N-1)T)]^\dagger$, where N is the number of samples, T is the Nyquist sampling period, and \dagger is the transpose operator[19]. The elements of the non-adaptive measurement matrix Φ , of dimension $M \times N$, can be chosen as identically distributed Gaussian random variables with zero mean and $1/N$ variance. If the sparsity order of the multipath-frame is given by K , then the number of required samples to fulfill the R.I.P. and incoherence property is given by

$$M \geq cK \log_{10} \left(\frac{N}{K} \right) \dots\dots(13)$$

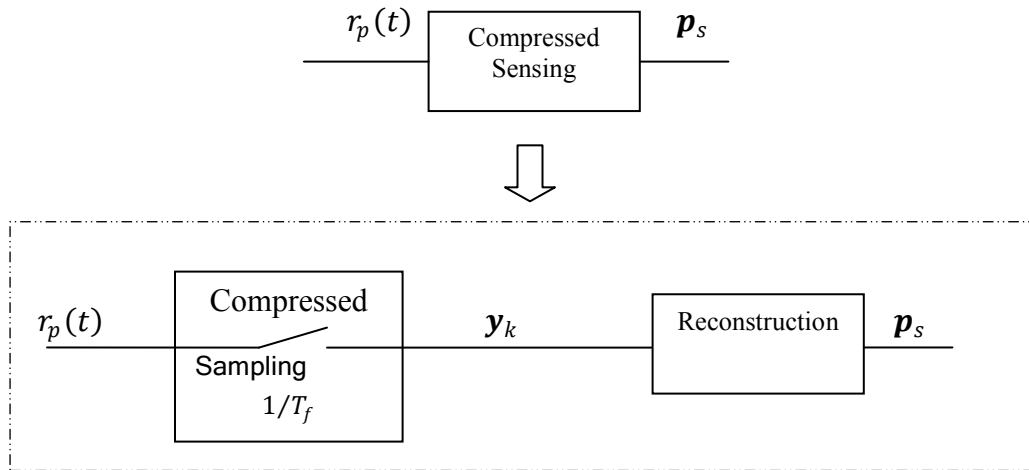


Figure 2.3: CS based channel-template estimation

Where c is an oversampling factor. Compressed sensing based sampling equation for k^{th} frame can be written as

$$\mathbf{y}_k = \Phi \mathbf{p}_k \tag{14}$$

Where \mathbf{p}_k is of $N \times 1$ dimensions

Φ is of $M \times N$ dimensions

\mathbf{y}_k is of $M \times 1$ dimensions

$$\begin{bmatrix} y_{k_1} \\ y_{k_2} \\ y_{k_3} \\ y_{k_4} \\ \vdots \\ \vdots \\ \vdots \\ y_{k_M} \end{bmatrix} = \begin{bmatrix} \varphi_{1,1} & \varphi_{1,2} & \varphi_{1,3} & \dots & \varphi_{1,N} \\ \varphi_{2,1} & \varphi_{2,2} & \varphi_{2,3} & \dots & \varphi_{2,N} \\ \varphi_{3,1} & \varphi_{3,2} & \varphi_{3,3} & \dots & \varphi_{3,N} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \varphi_{M,1} & \varphi_{M,2} & \varphi_{M,3} & \dots & \varphi_{M,N} \end{bmatrix} \begin{bmatrix} p_{k_1} \\ p_{k_2} \\ p_{k_3} \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ p_{k_{N-1}} \\ p_{k_N} \end{bmatrix}$$

Figure 2.4: Matrix representation equation 14 of CS-UWB for a frame

The M measurement samples are used to reconstruct noisy multipath-frame template $p_s(t)$ using the orthogonal matching pursuit (OMP) algorithm, an iterative greedy algorithm, which is computationally less expensive than linear optimization methods. It finds the component with maximum correlation in the measurement signal, eliminates it from the signal, and searches again for the maximum correlation component that remains in the residual signal[19].

We have $N_f N_p$ number of pilot frames, and M samples from each frame. According to [19], there are two methods to reconstruct noise multipath-frame template. First, the OMP algorithm can be used to reconstruct every frame $\widehat{\mathbf{p}}_k$ and average over $N_f N_p$ frames to obtain \mathbf{p}_s .

The objective function of this method can be defined as:

$$\widehat{\mathbf{p}}_k = \arg \min \|\mathbf{p}_k\|_1 \quad \text{such that} \quad \mathbf{y}_k = \Phi \mathbf{p}_k \quad \dots\dots\dots(15)$$

This method requires running the OMP algorithm $N_f N_p$ times and hence it is a computationally expensive method. The final multipath-frame template can be calculated by averaging the total number of reconstructed frames.

$$\mathbf{p}_s = \frac{1}{N_f N_p} \sum_{k=0}^{N_f N_p - 1} \widehat{\mathbf{p}}_k \quad \dots\dots\dots(16)$$

The second method (which will be referred to as conventional CS-UWB method) is one in which M samples from each frame can be averaged before reconstruction to obtain \mathbf{p}_s . In this method, the sampled vector, \mathbf{y} is given as

$$\mathbf{y} = \frac{1}{N_p N_f} \sum_{k=0}^{N_f N_p - 1} \mathbf{y}_k \quad \dots\dots\dots(17)$$

The objective function to recover \mathbf{p}_s using \mathbf{y} is given by

$$\mathbf{p}_s = \arg \min \|\mathbf{p}_k\|_1 \quad \text{such that} \quad \mathbf{y} = \Phi \mathbf{p}_k \quad \dots\dots\dots(18)$$

The second method is computationally less expensive because it requires only one reconstruction after ensemble averaging the frame samples over $N_f N_p$ frames. Moreover, it implicitly mitigates the effect of AWG noise.

2.4.2. Signal Demodulation (Proposed)

There are two signal demodulation methods. First, Analog demodulation, in which an approximate analog of the reconstructed multipath frame is used to demodulate the analog information-bearing signal using correlation based demodulator. A second method, we propose, is a new digital implementation method in which information-bearing signal and the reconstructed multipath frame are down-sampled below the Nyquist rate and demodulated using digital correlation method. Both of these methods are explained below:

Analog Demodulation

The compressed sensing based reconstructed multipath frame can be used to demodulate the information-bearing symbols at frame rate sampling using the correlation based approach in analog domain. However, we have discrete form of multipath-frame temple. Using Whittaker–Shannon interpolation formula \mathbf{p}_s can be approximated to analog $\hat{\mathbf{p}}_s(t)$. [25]

$$\hat{\mathbf{p}}_s(t) = \sum_{n=1}^N \mathbf{p}_s[n] \text{sinc}\left(\frac{t-nT}{T}\right) \dots\dots\dots(19)$$

Now $\hat{\mathbf{p}}_s(t)$ is used to demodulate the information-bearing part of received signal $r_s(t)$ which is $r(t)|_{i=Np}^{Ns}$. Each symbol consists of N_f frames, thus the decision statistic is given by adding N_f correlator output samples for the i^{th} symbol[19]. It can be written as:

$$z(i) = \sum_{j=0}^{N_f-1} \int_{jT_f+iT_s}^{(j+1)T_f+iT_s} r_s(t) \hat{\mathbf{p}}_s(t - jT_f - iT_s) \dots\dots\dots(20)$$

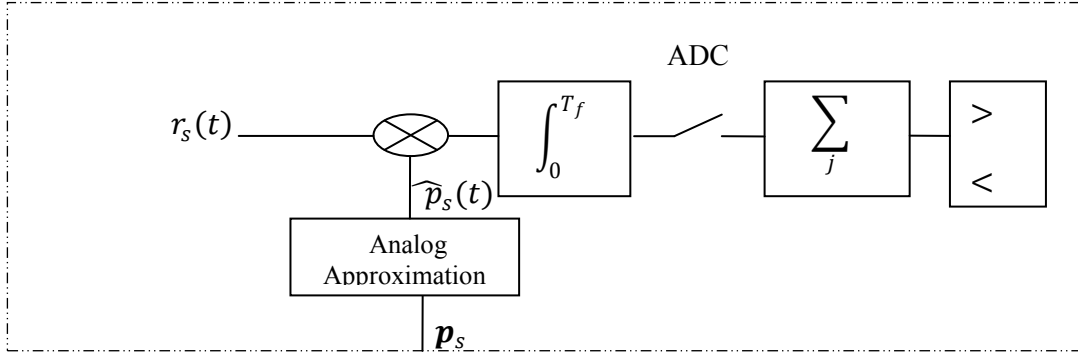


Figure 2.5: CS-UWB analog demodulation block diagram

The detected symbol is given by

$$\hat{b}_i = \text{sgn}(z(i)) \quad \dots\dots\dots(21)$$

Proposed Digital Receiver Design:

We propose a new demodulation method for CS-UWB receiver in the digital domain. In the digital demodulation of the information bearing signal, instead of approximating \mathbf{p}_s discrete vector into an analog version, the remaining information-bearing signal $r_s(t)$ and discrete vector \mathbf{p}_s are both down-sampled to $\mathbf{r}_s[m']$ and \mathbf{p}_{sd} respectively, where the down-sampling rate is less than the Nyquist rate of the signal. The following equation shows the down-sampling operation:

$$\mathbf{r}_s[m'] = r_s(t)\delta(t - m'T_m) \quad \dots\dots\dots(22)$$

Where $m' = 0,1,2 \dots \dots N_s N_f Q - 1$. Q is the number of samples in each frame such that $T_f = QT_m$. T_m is the sampling period which greater than the Nyquist sampling period i.e. $T_m > T$. Similarly, for digital demodulation, same number of samples Q , are downsampled from \mathbf{p}_s to \mathbf{p}_{sd} .

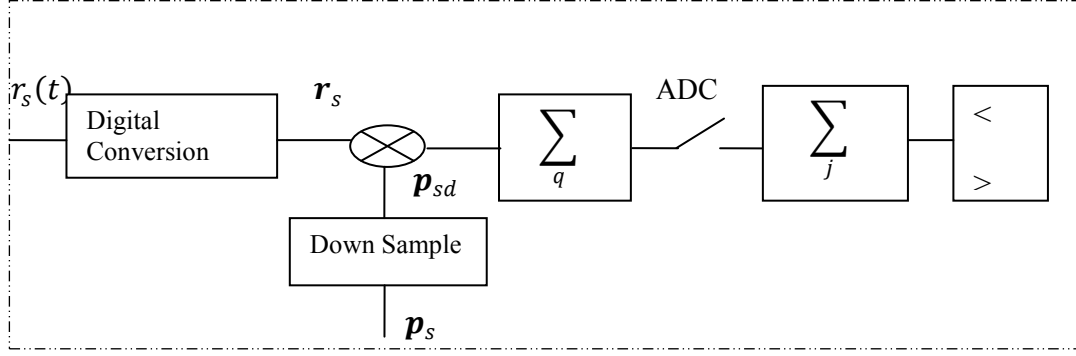


Figure 2.6: CS-UWB digital demodulation block diagram

$$\mathbf{p}_{sd}[q] = \mathbf{p}_s[n] \delta \left[n - \frac{N}{Q}q \right] \dots\dots\dots(23)$$

Where $q = 0,1,2 \dots Q$. The down-sampled vector \mathbf{p}_{sd} is then used to demodulate the digital received vector \mathbf{r} . Similar to the analog demodulation method, a decision variable is obtained by adding the correlated vector \mathbf{r}_s with \mathbf{p}_{sd} over N_f frames over the each symbol duration.

$$z[i] = \sum_{j=0}^{N_f-1} \sum_{q=0}^{Q-1} r_s p_{sd}[q - jQ - iQN_f] \dots\dots(24)$$

The decision over each demodulated symbol is made as follows:

$$b[i] = \text{sgn}(z[i]) \dots\dots\dots(25)$$

The next section, explains how CS-UWB is implemented in hardware, and its challenges.

3. CS-UWB Implementation

This chapter describes study of possible compressed sensing and related techniques' hardware implementation methods. The problem formulation of the thesis is also explained in the same chapter with proposed method of solution.

3.1. Study of CS Hardware Implementations

The concept of compressed sensing relinquishes the dogma of the Nyquist theorem which has ruled the entire electronics industry for very long time. It is a relatively new concept, with possibilities of application in wide variety of areas. However, one of the most challenging tasks in compressed sensing is its hardware implementation. This section describes the study of possible hardware implementation methods for compressed sensing or related techniques.

3.1.1. Random Filters:

It is based on the concept of compressed sensing, however unlike compressed sensing, it is simple and efficient[26]. A fixed random tap FIR filter is convolved with incoming signal and then down sampled at a rate lower than the Nyquist rate to get compressed output vector. The sampled signal is then reconstructed using greedy algorithm such as OMP.

It is simpler than compressed sensing because it doesn't require whole signal at once for sampling, it is time-invariant, with easy implementation in software and hardware, and it can be generalized to streaming and continuous-time signals.

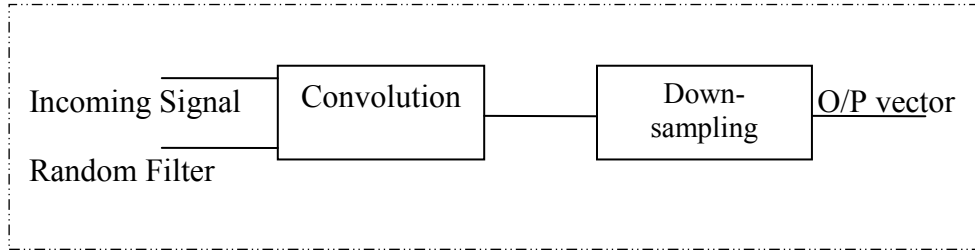


Figure 3.1: Random filter implementation block diagram

However, after its early presentation at a conference [26] not much research has been reported on random filters. Moreover, authors in the same paper encourage further development of random filters with more focus on the tradeoffs between measurements and computational costs.

3.1.2. Analog-to-Information Converter (AIC):

It is a method to convert streaming analog signals into discrete form without ADC at Nyquist rate. Compressed sensing can be implemented using one of the methods of AIC- random sampling, and random demodulation.

3.1.2.1. Random Sampling:

In this method, the incoming signal is non-uniformly sampled at an average sampling rate, lower than the Nyquist rate. It has two key components- random sampler which is implemented in hardware, and information recovery algorithm to reconstruct the sampled signal. Random sampling is implemented using Random Sampling Analog to Digital converter (RSADC) [27]. In RSADC non-uniform samples are taken of the incoming signal using pseudo-random generator clock which operates at or greater than Nyquist rate, to capture very short duration samples of the signal. However, these samples are taken with low probability. Figure 3.2 shows the block diagram of random sampling implementation.

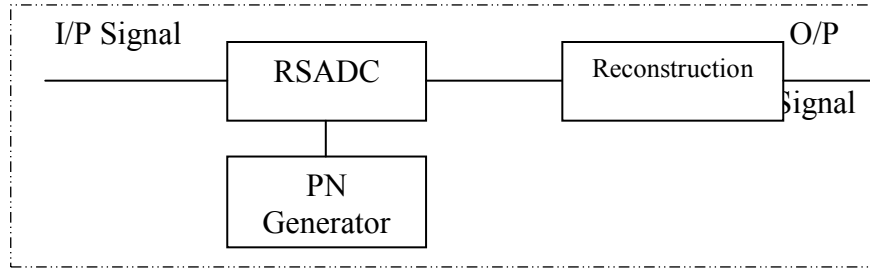


Figure 3.2: Random sampling based compressed sensing

The original signal is reconstructed using a recovery algorithm which is much faster than the compressed sensing algorithms. The naïve use of RSADC to wideband analog signal doesn't take advantage the designed algorithm. The wideband analog signal is sampled at an average rate, lower than the Nyquist rate, and acquired low rate samples are used to recover the original signal. A segmented AIC has been proposed in [28], which segments the signal before applying compressed sensing.

Implementation model for RSADC has been suggested in [29]. However, operation of some of the switching devices causes undesired behavior. Moreover, synchronization of the sampling part and recovery part must be perfect for successful reconstruction.

3.1.2.2. Random Demodulation:

It makes use of the existing state-of-art of electronics components. It uses mainly mixer, integrator, and low-rate ADC for implementation of AIC[24]. Figure 3.3 shows that the incoming wideband signal is multiplied by a predetermined pseudo-random (PN) sequence followed by low-pass filtering and low-rate ADC. PN sequence (± 1) with chip rate at or greater than Nyquist rate spreads the signal on entire spectrum, so that it doesn't get destroyed by the next stage. It changes the polarity of the streaming signal.

The key point to note in this method is that demodulation of the incoming signal using PN sequence (± 1) is much easier to implement in hardware than sampling it at the Nyquist rate[13]. Thus, it can be implemented using easily available components.

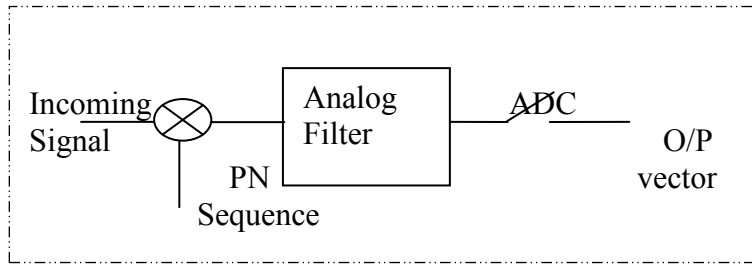


Figure 3.3: Random demodulation based compressed sensing

The reconstruction is done using convex optimization algorithms. However, slower sampling rate and complex recovery algorithm are some of the limitations of the random demodulation AIC method.

3.1.3. Distributed Amplifiers:

An alternative to the mixer-integrator implementation is suggested in[17]. Distributed Amplifiers (DA) can be used for analog compressed sensing followed by low rate ADCs. An implementation diagram of DA is shown in figure 3.4.

A DA consists of multiple repeated taps, each containing a section of micro-strip input and output transmission lines, and the gain cells. There are basically three advantages of DA: characteristic impedance of high frequencies changes very slowly, a time delay of very short duration (50ps) can be achieved, and gain coefficient scan be re-configured to change the random coefficients of the measurement matrix.

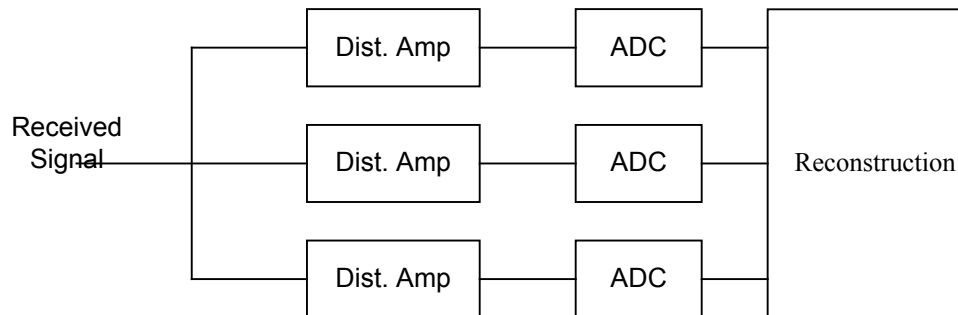


Figure 3.4: Receiver structure based on Distributed Amplifiers [17]

However, the concept of DA has been proposed for a noiseless environment for the sake of simplicity[17]. It is suggested that the performance of the system will degrade substantially in Gaussian noise environment and the authors encourage study on the signal reconstruction for noisy channels. Moreover, the effect of nonlinearity has also been neglected for the simple design of the system. The non-linearity if considered may produce undesired components in the coefficients of the gain cells.

3.2. Other Implementation Challenges

Another challenge for CS-UWB is the implementation of the measurement matrix. The most common distribution for measurement matrix that satisfies the incoherence property and R.I.P property of the compressed sensing is the Gaussian distribution. However, it is also the most difficult to implement in the hardware[19].

Therefore[13, 24], uses Maximum-Length Linear Feedback Shift Register (MLFSR) which produces $+1/-1$ values at the Nyquist rate of the signal. However [19], uses an analog friendly hadamard transform, which is comparatively easier to implement in hardware. In [30], independent identically distributed(i.i.d.) Bernoulli random distribution is used for its measurement operator.

The above mentioned hardware challenges are related mainly to the component level of CS-UWB implementation. However, for successful reconstruction of the sampled signal it requires M number of mixer-integrator parallel branches. In case of UWB, this M may be very large. The structural problem is the main part of this thesis and is discussed in next section in detail.

3.3. Problem Formulation

This thesis identifies its problem in the structural implementation of the measurement matrix Φ , which requires a large number M of mixer and integrator branches to obtain a reduced length vector \mathbf{y} . This problem is illustrated below.

If \mathbf{x} is a discrete, finite length, noise-free UWB frame long signal of length N , and sparsity order K , then the following matrix representation shows that every measurement sample is the inner product between each row vector $\varphi_{m,N}$ of measurement matrix Φ and column long vector \mathbf{x} where $m = 1, 2, \dots, M$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ y_M \end{bmatrix} = \begin{bmatrix} \varphi_{1,1} & \varphi_{1,2} & \varphi_{1,3} & \dots & \varphi_{1,N} \\ \varphi_{2,1} & \varphi_{2,2} & \varphi_{2,3} & \dots & \varphi_{2,N} \\ \varphi_{3,1} & \varphi_{3,2} & \varphi_{3,3} & \dots & \varphi_{3,N} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \varphi_{M,1} & \varphi_{M,2} & \varphi_{M,3} & \dots & \varphi_{M,N} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ x_{N-1} \\ x_N \end{bmatrix}$$

Figure 3.5: Compressed Sensing matrix representation

The above matrix can be represented in hardware using M mixers and integrator circuits in the figure 3.6.

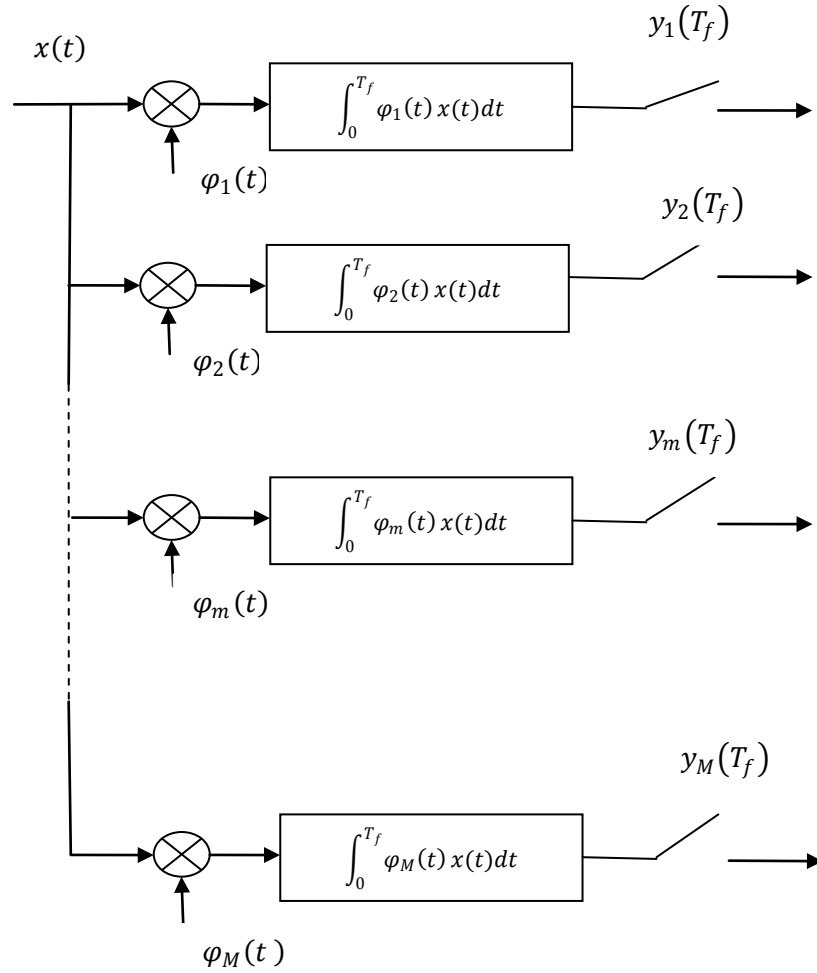


Figure 3.6: Hardware implementation of CS-UWB

Consider an example of a discrete length UWB multipath-frame of $N = 1200$ with sparsity order of $K = 35\%$. then, the required number of measurement samples M can be given by the equation $M \geq cK \log \left(\frac{N}{K} \right)$, with $c = 1$, such that it satisfies R.I.P. property and meets the incoherence criterion between the measurement matrix and the basis matrix. In this case, M is approximately found to be 440, which is very large as far as the hardware complexity of the system is concerned.

A parallel segmented compressed sensing method is proposed in [30] in the context of cognitive radio. It divides the signal into smaller segments before compression. Then, it rearranges the samples in a long vector before reconstruction. Therefore, each segment now requires lesser number of measurement samples. Moreover, the performance of the system improves with increased number of samples. A similar work has been proposed in [30]. However, segmenting the signal comes at the cost of increased sensing time. It requires windowing function for segmenting the signal, which requires fine timing precision. Additionally, each sub-sample in above mentioned papers collects incomplete information of the signal. Our proposed method is similar in approach but different and very specific to UWB signal structure.

3.4. Proposed Method

The conventional method discussed in section 2.4 uses CS to collect M samples from each frame of the pilot symbol, and reconstructs the multipath frame template after ensemble averaging over N_f frames. The reconstructed multipath-frame template is then used to demodulate the remaining information bearing signal by using integrate and dump operation. The number M in the conventional method is found to be very large for hardware implementation.

Therefore, we propose a different reconstruction method which significantly reduces the number of mixer and integrator branches M , and also improves the performance of the system for the same M .

We propose a joint frame reconstruction method, which exploits the unique symbol structure of the UWB signal. As we have discussed, a UWB symbol consists of a number of frames which carry the same information every T_f time period. The proposed method collects M measurement samples from each frame, and stacks them in a long sequence before reconstruction. The simulation results show the efficiency of the proposed method.

Considering the same CS-UWB system as mentioned in section 2.4,

$$p_k(t) = b_i \left(\frac{k}{N_f} \right) p_h(t) + n(t) \dots\dots\dots(26)$$

Where $p_k(t)$ is the k^{th} received pilot frame and $k = 1,2,3 \dots \dots N_p N_f$

In the proposed method, the elements of the measurement matrix are also chosen from identically distributed Gaussian random variables with zero mean and $1/N$ variance. Each frame is sampled by a different Φ . Therefore, for k^{th} frame, Φ_k can be attributed as a measurement matrix with the dimension $M \times N$. Consider $N_p = 1$, for the simplicity to understand the method. However, it can be extended to $N_p > 1$, where the same method can be repeated for frame reconstruction for each pilot symbol. Thus, $k = 1,2,3 \dots \dots N_f$

The following equation shows the CS operation over k^{th} frame.

$$y_k = \Phi_k p_k \dots\dots(27)$$

Where $y_k = [y_{k_1} y_{k_2} y_{k_3} \dots \dots y_{k_M}]^T$, and contains M samples from each p_k frame.

The proposed reconstruction method stacks the collected samples from each frame as:

$$y = [y_1^T y_2^T y_3^T \dots \dots y_k^T \dots \dots y_{N_f}^T]^T \dots\dots(28)$$

The measurement matrix used for each frame is also stacked in similar manner to get the actual reconstruction matrix:

$$V = [\Phi_1 \Phi_2 \Phi_3 \dots \Phi_k \dots \Phi_{N_f}]^T \dots\dots(29)$$

The objective function for reconstruction is again based on $l_1 - norm$ optimization process as follows:

$$\mathbf{p}_s = \arg \min \|\mathbf{p}_k\|_1 \quad \text{such that} \quad \mathbf{y} = \mathbf{V}\mathbf{p}_k \quad \dots(30)$$

In this thesis, we used orthogonal matching pursuit (OMP) algorithm for the reconstruction. The sampled vector \mathbf{y} and the reconstruction matrix \mathbf{V} are the only two main input requirements for the OMP algorithm. Similar to the section 2.4, the reconstructed and analog approximated frame $p_s(t)$ is used to demodulate the information-bearing signal.

Thus,

$$z(i) = \sum_{j=0}^{N_f-1} \int_{jT_f+iT_s}^{(j+1)T_f+iT_s} r(t) p_s(t - jT_f - iT_s) \quad \dots(31)$$

The following section shows the performance and complexity comparisons between the conventional method and proposed method.

4. Results

The results of the proposed method are presented in this chapter. It begins with the description of simulation setup. The performance for the mentioned method is presented with graphs.

4.1. Simulation Setup

This section describes various parameters setup for Matlab simulations. The proposed method is compared with the CS-UWB time sparsity method shown in [22], which will be referred to as the conventional CS-UWB method. In the simulation, the Gaussian monocycle pulse is used with time width $T_w = 0.6ns$. Each UWB symbol has $N_f = 5$ number of frames with $T_f = 200ns$ as the frame duration. Thus, the time duration of the symbol is $T_s = 1\mu sec$. Each burst of the signal consists of $N_i = 1000$ information-bearing symbols and $N_p = 1$ number of pilot symbols or otherwise stated. For simplicity, the system is modeled for a single user, with only one pulse in each frame.

The I-UWB channel is modeled as the indoor residential LOS channel (CM1) mentioned in[20]. Various parameters for the used channel model are set according to the recommended values in[20]. Inter-Frame Interference (IFI) is assumed to be absent so that $T_f \geq T_w + \tau_{max}$ where τ_{max} is the maximum delay spread of the channel, and T_w is the pulse duration. In the simulation setup, frequency dependency path loss is taken out of the used channel model[20] for the sake of simplicity. This means that only real-valued impulse response is considered.

In the simulation, 2-PAM modulation is used in which independent binary symbols are generated with equal probability. The system sampling frequency is set to 20GHz , which is higher than the Nyquist frequency of the UWB pulse. It gives a time resolution of 50ps.

The number of channel realizations is kept at 4000 (for each channel model) or otherwise stated. The channel is assumed to be invariant over a burst of symbols. Bit error rate (BER) is used as the performance criterion with respect to signal-to-noise ratio. However, performance comparisons are done on averaging the (BER) over the total number of channel realizations.

The number of samples in each multipath frame is $N = 1200$ with sparsity $K = 35\%$, the minimum number of samples required from each frame is $M_{min} = 440$. Orthogonal Matching Pursuit(OMP) algorithm is used for sparse signal recovery. The number of iteration is set to 300 for residual energy of $\varepsilon = 10^{-4}$ [31]. The high value of ε may result in fast processing of the algorithm, however, the results may not be accurate. On the other hand, if the value is kept very low, then the algorithm takes a very long time to process results. The term ‘Compression Ratio’ is used quite often which means the ratio of actual number M and length of a frame N . The following section compares the performance of conventional compressed sensing based ultra-wideband (CCS-UWB) receiver and proposed compressed sensing based ultra-wideband (PCS-UWB) for different compression ratios and number of frames per symbol.

4.2. Performance Comparisons

The performance comparison can be divided into the following categorizes based on the cases:

Case 1: Compression ratio $\frac{M}{N} < 35\%$ and pilot symbol $N_p = 1$

In this case, the compression ratio M/N is kept below the approximate sparsity level $K = 35\%$ of the UWB noiseless frame. Figure 4.1 shows the performance comparison for both the conventional CS-UWB (CCS-UWB) and proposed CS-UWB (PCS-UWB). It is obvious that the conventional method would not be able to recover the estimated frame

at 5% of compression ratio because it fails to fulfill the criterion of compressed sensing reconstruction of $M \geq cK \log_{10}(\frac{N}{K})$. However, at 10% and 20% of the compression ratio,

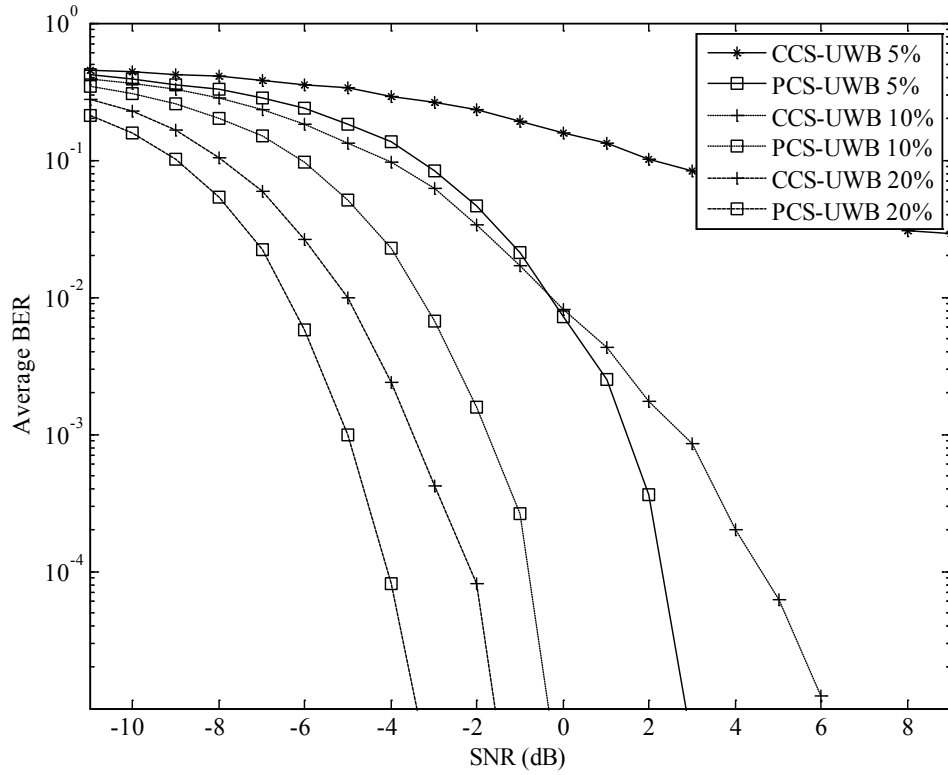


Figure 4.1: Performance graph for compression ratio less than 35%

the larger components of the frame are reconstructed which provide reasonable performance even at low compression region. However, the proposed method successfully performs even with low compression ratio of 5% as it increases its number of samples to MN_f by stacking M samples from each frame.

Thus, for lower compression ratio, the proposed method still successfully estimates the channel frame. The performance of both methods improves as we increase compression ratio from $M/N = 5\%$ to $M/N = 20\%$ which also shows that an increase in the number of projections improves the performance.

Case 2: Compression ratio $\frac{M}{N} > 35\%$ and pilot symbol $N_p = 1$

In this case, performance is compared for compression ratio M/N greater than 35%. Figure 4.2, shows that both methods successfully perform for increased compression ratio

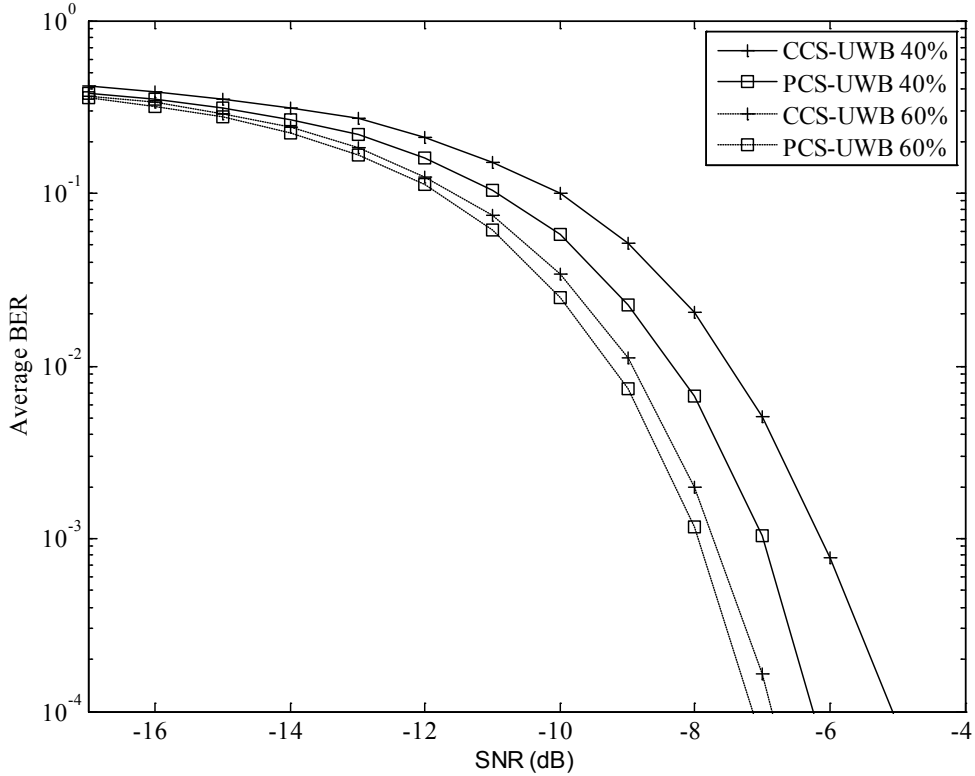


Figure 4.2: Performance graph for compression ratio more than 35%

as both of them fulfils the compressed sensing criteria. However, the key point to note here is that the performance difference between both methods for each compression.

For $M/N = 40%$, proposed method outperforms the conventional method. However, as the compression ratio increases from $M/N = 40%$ to $M/N = 60%$, proposed method still performs better but the performance gap decreases.

Case 3: Compression ratio $\frac{M}{N}=10%$ and $\frac{M}{N}=20%$ with $N_f = 5$ and 15 number of frames in each symbol for both compression ratios.

In this case, the compression ratio M/N is kept constant at 10% and 20%. However, the number of frames per symbol is varied for both compression ratios. Figure 4.3 shows

the performance comparison for both conventional CS-UWB and proposed CS-UWB under mentioned criteria. As the number of frames increases the performance of the proposed method and conventional CS method improves.

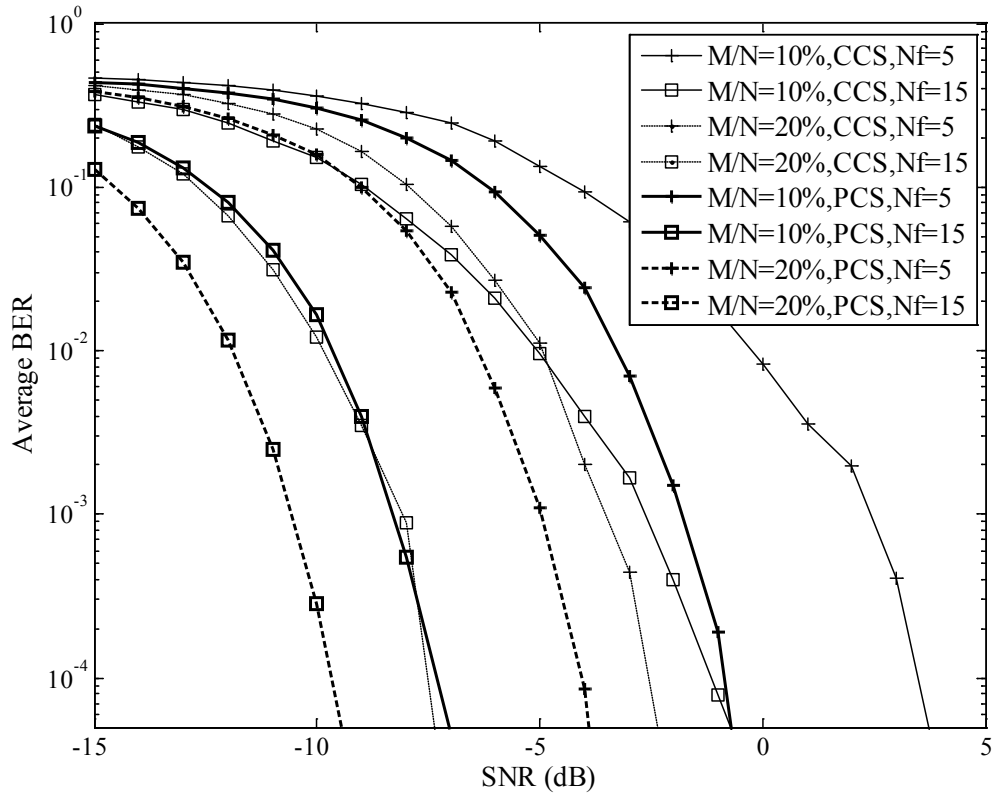


Figure 4.3: Performance graph for different number frames in each symbol

The figure above shows the comparison for frame size $N_f = 5$ or 15 . It is concluded that the performance can be increased for the same hardware complexity but at the cost of decreased data rate.

4.3. Hardware Comparison

The key advantage of the proposed method over conventional method is that the proposed method performs better even for low compression ratio. The following table shows the comparison of using different compression ratios on the hardware reduction and performance gain/reduction. The comparisons are made at approximately constant performance criterion of

$$\text{BER} \approx 10^{-2}$$

$$M_{min} = 440$$

$$\text{Parallel Structures' Reduction (\%)} = \frac{M_{min} - M}{M_{min}} \times 100$$

$$\text{Performance Improvement (dB)} = \text{SNR of PCS-UWB} - \text{SNR of conventional CCS-UWB}$$

The second column in the table 4.1 shows the reduction in hardware complexity achieved with reduced number of parallel structures. The third column shows performance comparison with respect to the conventional CS-UWB for same M, number of parallel branches. The performance gap between CCS-UWB and PCS-UWB decreases with increase in M. **Comparison of hardware complexity and performance in different compression ratios**

Table 4.1
Comparison of hardware complexity and performance in different compression ratios

M	Parallel Structures' Reduction compared to M_{min} (Approx %)	Performance Improvement compared to CCS-UWB for same M (Approx dB)
60	86	Very high
120	72	5
240	45	2

5. Conclusion and Future Work

This thesis provides an overview of UWB and CS-UWB receiver. An introduction to CS and its implementation challenges. The three main contributions of this thesis are as follows:

1. A new CS-UWB digital receiver design is proposed. It requires CS-UWB implementation at low-rate processing in the digital domain.
2. A study of CS hardware implementation, mainly of analog signals. The implementation methods are explained with their working, advantages, and challenges.
3. A new reconstruction algorithm is proposed for CS-UWB implementation which not only reduces the hardware complexity of the system but also improves the performance. It is an incremental advancement to reduce the hardware complexity of the system.

Compressed sensing has applications in various fields of signal processing. CS-UWB also has its similarity to cognitive radio which requires very wide bandwidth receiver processing. Thus, the future work related to this thesis can be described below:

1. The proposed digital receiver design can be applied to other high bandwidth system where Nyquist rate sampling is a limitation.
2. The proposed reconstruction algorithm can be extended to CS-UWB multiuser scenario. However, it will also require more processing.
3. Similar to the UWB, other high bandwidth technologies such as cognitive radio, faces similar CS based hardware challenges. The proposed receiver design or reconstruction algorithm can also be applied to cognitive radio or wireless sensor networks with relevant modifications.

References

- [1] G. B. Giannakis, "Ultra-wideband communications: an idea whose time has come," in *Signal Processing Advances in Wireless Communications, 2003. SPAWC 2003. 4th IEEE Workshop on*, 2003, p. 3.
- [2] M. Z. Win and R. A. Scholtz, "Impulse radio: how it works," *Communications Letters, IEEE*, vol. 2, pp. 36-38, 1998.
- [3] H. Nikookar and R. Prasad, "UWB For Wireless Communications," in *Introduction to Ultra Wideband for Wireless Communications*, ed: Springer Netherlands, 2009, pp. 11-28.
- [4] W. P. Siriwongpairat and K. J. R. Liu, *MIMO Multiband OFDM*: John Wiley & Sons, Inc., 2007.
- [5] J. D. Choi and W. E. Stark, "Performance of ultra-wideband communications with suboptimal receivers in multipath channels," *Selected Areas in Communications, IEEE Journal on*, vol. 20, pp. 1754-1766, 2002.
- [6] K. Witrisal, *et al.*, "Noncoherent ultra-wideband systems," *Signal Processing Magazine, IEEE*, vol. 26, pp. 48-66, 2009.
- [7] J. L. Paredes, *et al.*, "Compressed Sensing for Ultrawideband Impulse Radio," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, 2007, pp. III-553-III-556.
- [8] Z. Peng, *et al.*, "A Compressed Sensing Based Ultra-Wideband Communication System," in *Communications, 2009. ICC '09. IEEE International Conference on*, 2009, pp. 1-5.
- [9] W. Zhongmin, *et al.*, "Compressed detection for ultra-wideband impulse radio," in *Signal Processing Advances in Wireless Communications, 2007. SPAWC 2007. IEEE 8th Workshop on*, 2007, pp. 1-5.
- [10] W. U. Bajwa, *et al.*, "Compressed channel sensing," in *Information Sciences and Systems, 2008. CISS 2008. 42nd Annual Conference on*, 2008, pp. 5-10.
- [11] Z. Tian and B. M. Sadler, "Weighted energy detection of ultra-wideband signals," in *Signal Processing Advances in Wireless Communications, 2005 IEEE 6th Workshop on*, 2005, pp. 1068-1072.
- [12] V. Lottici, *et al.*, "Channel estimation for ultra-wideband communications," *Selected Areas in Communications, IEEE Journal on*, vol. 20, pp. 1638-1645, 2002.

- [13] J. N. Laska, *et al.*, "Theory and Implementation of an Analog-to-Information Converter using Random Demodulation," in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, 2007, pp. 1959-1962.
- [14] E. J. Candes and M. B. Wakin, "An Introduction To Compressive Sampling," *Signal Processing Magazine, IEEE*, vol. 25, pp. 21-30, 2008.
- [15] D. L. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, pp. 1289-1306, 2006.
- [16] R. G. Baraniuk, "Compressive Sensing [Lecture Notes]," *Signal Processing Magazine, IEEE*, vol. 24, pp. 118-121, 2007.
- [17] Y. Depeng, *et al.*, "Compressed sensing based UWB receiver: Hardware compressing and FPGA reconstruction," in *Information Sciences and Systems, 2009. CISS 2009. 43rd Annual Conference on*, 2009, pp. 198-201.
- [18] M. Jia, *et al.*, "Sampling rate reduction for 60 GHz UWB communication using compressive sensing," in *Signals, Systems and Computers, 2009 Conference Record of the Forty-Third Asilomar Conference on*, 2009, pp. 1125-1129.
- [19] F. M. Naini, *et al.*, "Compressive sampling of pulse trains: Spread the spectrum!," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 2009, pp. 2877-2880.
- [20] J.R.Foerster, "Channel Modeling Sub-committee Report Final.IEEE P802.12-02/368r5SG3a," *IEEE P802.15 Working group for WPAN*, 2002.
- [21] W. Zhongmin, *et al.*, "Compressed Detection for Pilot Assisted Ultra-Wideband Impulse Radio," in *Ultra-Wideband, 2007. ICUWB 2007. IEEE International Conference on*, 2007, pp. 393-398.
- [22] J. L. Paredes, *et al.*, "Ultra-Wideband Compressed Sensing: Channel Estimation," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 1, pp. 383-395, 2007.
- [23] L. Zhilin, *et al.*, "Sparse Signal Recovery via Optimized Orthogonal Matching Pursuit," in *Image and Signal Processing, 2009. CISP '09. 2nd International Congress on*, 2009, pp. 1-4.
- [24] K. Sami, *et al.*, "Analog-to-Information Conversion via Random Demodulation," in *Design, Applications, Integration and Software, 2006 IEEE Dallas/CAS Workshop on*, 2006, pp. 71-74.

- [25] R. J. Marks, Ed., *Introduction to Shannon Sampling and Interpolation Theory*. New York:Springer-Verlag, 1991, p.^pp. Pages.
- [26] J. A. Tropp, *et al.*, "Random Filters for Compressive Sampling and Reconstruction," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, 2006, pp. III-III.
- [27] L. Jason, *et al.*, "Random Sampling for Analog-to-Information Conversion of Wideband Signals," in *Design, Applications, Integration and Software, 2006 IEEE Dallas/CAS Workshop on*, 2006, pp. 119-122.
- [28] O. Taheri and S. A. Vorobyov, "Segmented Compressed Sampling for Analog-to-Information Conversion: Method and Performance Analysis," *Signal Processing, IEEE Transactions on*, vol. 59, pp. 554-572, 2011.
- [29] T. Ragheb, *et al.*, "Implementation models for analog-to-information conversion via random sampling," in *Circuits and Systems, 2007. MWSCAS 2007. 50th Midwest Symposium on*, 2007, pp. 325-328.
- [30] Y. Zhuizhuan, *et al.*, "Mixed-signal parallel compressed sensing and reception for cognitive radio," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 3861-3864.
- [31] D.Donoho. Sparselab [Online]. Available: <http://sparselab.stanford.edu/>.

Appendices

Appendix A

MATLAB script files used for the simulation for case 1

```
function [SNR cber pber]=
CSUWB_BELOW_SPARSITY(crs,Np,Nf,CMN,num_channels)

%CSUWB_BELOW_SPARSITY(cr,Np,Nf,ch,num_channel)
%CSUWB_BELOW_SPARSITY: Compressed Sensing based Ultra-wideband system
with
%          sensing matrix operating below the sparsity level of 35%
% Inputs
%crs      Compression Ratio (M/N), default [.05,.10,.20]
%Np      Number of pilot symbols, default 1
%Nf      Number of frames in each symbol, default 5
%CMN     Type of channel according to the Reference 21,default 1
%num_channel  Number of channel realizations, default 4000
%Outputs
% Bit Error rate performance of CCS-UWB and PCS-UWB for
% different compression ratios
%SNR     The range of Signal to Noise ratio
%cber    CCS-UWB bit error rate for different compression ratios
%pber    PCS-UWB bit error rate for different compression ratios
%Description
%CSUWB_BELOW_SPARSITY gives BER performance graph for conventional
%compressed sensing ultra-wideband (CCS-UWB) and proposed compressed
%sensing ultra-wideband (PCS-UWB) for different compression ratios (M/N)

if nargin<5
    num_channels=4000;
end

if nargin<4
    CMN=1;
end
if nargin<3
    Nf=5;
end
if nargin<2
```

```

    Np=1
end
if nargin<1
    crs=[0.05 0.1 0.2]
end

#####PARAMETERS #####

Fs=20e9;          %Digital frequency
fp=2e9;          %Pulse frequency
Tc=10e-9;        %Chip Width
Tf=200;          %Frame duration in Nanoseconds
Ni=1000;         %Number of information symbols
Ns=Np+Ni;        %Total number of symbols
SNR=[-11:1:9];   % SNR in dB

#####PULSE GENERATION #####

tc = gmonopuls('cutoff', fp); % Pulse width parameter
Tp = -2*tc : 1/Fs : 2*tc;     % Actual pulse width
p = gmonopuls(Tp,fp);        % UWB Pulse

#####CHIP GENERATION #####

chiptime=0:1/Fs:Tp-1/Fs;     % Chip samples
chip=zeros(length(chiptime),1);
chip(1:length(p))=(p);      % UWB Chip

##### UWB CHANNEL #####

uwb_h=uwb_channel(CMN,num_channels); % Reference 21

##### SIMULATION #####

for cr=1:length(crs)          % Iteration for each compression ratio

for snrdb=1:length(SNR)      % Iteration for each SNR
    for z=1:num_channels     % Number of channel realizations

##### FRAME GENERATION #####

frame1=conv(uwb_h(:,num_channels),chip);

```

```

frame=frame1(1:1200);
N=length(frame);

%##### PILOT SIGNAL #####

sig=zeros(Nf*N,1);

for j=1:Nf

    sig((j-1)*N+1:j*N,1)=frame;
end

for n=1:Np
    s2(1:length(sig),n)=sig;
end

%##### TRANSMITTED SIGNAL #####

s=single(zeros(Ns*length(sig),1));
s(1:Np*length(sig))=s2(:);
Mod=(sign(randn(Ni,1)));          % Modulation
s3=single(sig*Mod(1:Ni,1)');
s(Np*length(sig)+1:Ns*length(sig),1)=s3(:); % Modulated TX signal
clear s3;

%##### RX SIGNAL GENERATION #####
rt=SNR(snrdb);                    % SNR value
len=length(s);                    % Length of TX signal
awg=awgnnoise(rt,s,len,fp,Fs);    %AWG Noise generation
r=s+awg;                          % RX signal with AWG Noise
clear s;

%#####Reconstruction Methods#####

M=single(round(crs(cr)*N));        % Number of reduced dimation

cy=zeros(M,1);
py=zeros(M*Nf,1);
cfrmrec=zeros(N,1);
pfrmrec=zeros(N,1);
pPhi=zeros(M*Nf,N);
cPhi=randn(M,N);

```



```

for l=1:Nf
    if l==1;
        pPhi((l-1)*M+1:l*M,1:N)=cPhi;
    else
        pPhi((l-1)*M+1:l*M,1:N)=randn(M,N);
    end
end

%##### RECEIVER #####

for n=1:Np
    for j=1:Nf

        rp(1:N,1)=r((n-1)*length(sig)+(j-1)*N+1:(n-1)*length(sig)+j*N,1);

        cy=cy+cPhi*rp; % CCS-UWB
        py((j-1)*M+1:j*M,1)= pPhi((j-1)*M+1:j*M,1:N)*rp(1:N); % PCS-UWB

%##### RECONSTRUCTION #####
end
end

[cfmrec(1:N,1),iters, activationHist]=SolveOMP(cPhi,cy/Nf,N,[],0,0);
[pfrmrec(1:N,1),iters, activationHist]=SolveOMP(pPhi,py,N,300,[],0,0,1e-4);

%#####Demodulation#####

for n=1:Ni
    for j=1:Nf

        cfrm(1:N,j)=r((n+Np-1)*length(sig)+(j-1)*N+1:(n+Np-1)*length(sig)+j*N).*cfmrec;
        cfrmsum(1,j)=sum(cfrm(:,j));

        pfrm(1:N,j)=r((n+Np-1)*length(sig)+(j-1)*N+1:(n+Np-1)*length(sig)+j*N).*pfrmrec;
        pfrmsum(1,j)=sum(pfrm(:,j));

    end

    csymb(n)=sign(sum(cfrmsum(1,:)/N));

```

```

    psymb(n)=sign(sum(pfrmsum(1,:)/N));

end

clear r;
clear ynew;
clear frmrec;
clear frmrec2;

%##### ERROR CALCULATION #####

ccnterr(z,snrdb)=length(find(Mod-csymb'));

pcnterr(z,snrdb)=length(find(Mod-psymb'));

end

end

%##### BIT ERROR RATE #####

cber(cr,1:snrdb)=sum(ccnterr(:,:))/(z*(Ni));    %BER for CCS-UWB

pber(cr,1:snrdb)=sum(pcnterr(:,:))/(z*(Ni));    %BER for PCS-UWB
end

%##### PERFORMANCE PLOTS #####

figure,
semilogy(SNR,cber(1,1:snrdb),'k-*',SNR,pber(1,1:snrdb),'k-
s',SNR,cber(2,1:snrdb),'k:+',SNR,pber(2,1:snrdb),'k:s',SNR,cber(3,1:snrdb),'k--
+',SNR,pber(3,1:snrdb),'k--s')
title('BER Performance with Compression ratio (M/N) less than 35%')
xlabel('SNR (dB)')
ylabel('Average BER')
legend('CCS-UWB 5%', 'PCS-UWB 5%', 'CCS-UWB 10%', 'PCS-UWB 10%', 'CCS-
UWB 20%', 'PCS-UWB 20%')
axis([-11 9 1e-5 1])

```

Appendix B

MATLAB script files used for the simulation for case 2

```
function [SNR cber pber]=
CSUWB_ABOVE_SPARSITY(crs,Np,Nf,CMN,num_channels)

%CSUWB_ABOVE_SPARSITY(cr,Np,Nf,ch,num_channel)
%CSUWB_ABOVE_SPARSITY: Compressed Sensing based Ultra-wideband system
with
%          sensing matrix operating above the sparsity level of 35%
% Inputs
%crs      Compression Ratio (M/N), default [0.4 0.6]
%Np      Number of pilot symbols, default 1
%Nf      Number of frames in each symbol, default 5
%CMN     Type of channel according to the Reference 21, default 1
%num_channel  Number of channel realizations, default 1000
%Outputs
% Bit Error rate performance of CCS-UWB and PCS-UWB for
% different compression ratios
%SNR     The range of Signal to Noise ratio
%cber    CCS-UWB bit error rate for different compression ratios
%pber    PCS-UWB bit error rate for different compression ratios
%Description
%CSUWB_ABOVE_SPARSITY gives BER performance graph for conventional
%compressed sensing ultra-wideband (CCS-UWB) and proposed compressed
%sensing ultra-wideband (PCS-UWB) for different compression ratios (M/N)
if nargin<5
    num_channels=2;
end

if nargin<4
    CMN=1;
end
if nargin<3
    Nf=5;
end
if nargin<2
    Np=1
end
if nargin<1
    crs=[0.4 0.6]
end
```

```

#####PARAMETERS #####

Fs=20e9;          %Digital frequency
fp=2e9;          %Pulse frequency
Tc=10e-9;        %Chip Width
Tf=200;          %Frame duration in Nanoseconds
Ni=2;           %Number of information symbols
Ns=Np+Ni;        %Total number of symbols
SNR=[-17:1:-3];  % SNR in dB

#####PULSE GENERATION #####

tc = gmonopuls('cutoff', fp);  % Pulse width parameter
Tp = -2*tc : 1/Fs : 2*tc;      % Actual pulse width
p = gmonopuls(Tp,fp);          % UWB Pulse

#####CHIP GENERATION #####

chiptime=0:1/Fs:Tp-1/Fs;       % Chip samples
chip=zeros(length(chiptime),1);
chip(1:length(p))=(p');        % UWB Chip

##### UWB CHANNEL #####

uwb_h=uwb_channel(CMN,num_channels);

##### SIMULATION #####

for cr=1:length(crs)           % Iteration for each compression ratio

for snrdb=1:length(SNR)        % Iteration for each SNR
    for z=1:num_channels        % Number of channel realizations

##### FRAME GENERATION #####

frame1=conv(uwb_h(:,num_channels),chip);
frame=frame1(1:1200);
N=length(frame);

##### PILOT SIGNAL #####

sig=zeros(Nf*N,1);

for j=1:Nf

```

```

    sig((j-1)*N+1:j*N,1)=frame;
end

for n=1:Np
    s2(1:length(sig),n)=sig;
end

%##### TRANSMITTED SIGNAL #####

s=single(zeros(Ns*length(sig),1));
s(1:Np*length(sig))=s2(:);
Mod=(sign(randn(Ni,1)));          % Modulation
s3=single(sig*Mod(1:Ni,1)');
s(Np*length(sig)+1:Ns*length(sig),1)=s3(:); % Modulated TX signal
clear s3;

%##### RX SIGNAL GENERATION #####
rt=SNR(snrdB);                    % SNR value
len=length(s);                    % Length of TX signal
awg=awgnnoise(rt,s,len,fp,Fs);    %AWG Noise generation
r=s+awg;                          % RX signal with AWG Noise
clear s;

%#####Reconstruction Methods#####

M=single(round(crs(cr)*N));        % Number of reduced dimation

cy=zeros(M,1);
py=zeros(M*Nf,1);
cfmrec=zeros(N,1);
pfrmrec=zeros(N,1);
pPhi=zeros(M*Nf,N);
cPhi=randn(M,N);

for l=1:Nf
    if l==1;
        pPhi((l-1)*M+1:l*M,1:N)=cPhi;
    else
        pPhi((l-1)*M+1:l*M,1:N)=randn(M,N);
    end
end
end

```

```

##### RECEIVER #####

for n=1:Np
    for j=1:Nf

        rp(1:N,1)=r((n-1)*length(sig)+(j-1)*N+1:(n-1)*length(sig)+j*N,1);

        %%%%%%%%% Compressed Sampling for ALL
        Methods%%%%%%%%
        cy=cy+cPhi*rp; % CCS-UWB
        rp((j-1)*M+1:j*M,1)= pPhi((j-1)*M+1:j*M,1:N)*rp(1:N); % PCS-UWB

        ##### RECONSTRUCTION #####
    end
end

[cfmrec(1:N,1),iters, activationHist]=SolveOMP(cPhi,cy/Nf,N,[],0,0);
[pfrmrec(1:N,1),iters, activationHist]=SolveOMP(pPhi,py,N,300,[],0,0,1e-4);

#####Demodulation#####

for n=1:Ni
    for j=1:Nf

        cfrm(1:N,j)=r((n+Np-1)*length(sig)+(j-1)*N+1:(n+Np-1)*length(sig)+j*N).*cfmrec;
        cfrmsum(1,j)=sum(cfrm(:,j));

        pfrm(1:N,j)=r((n+Np-1)*length(sig)+(j-1)*N+1:(n+Np-1)*length(sig)+j*N).*pfrmrec;
        pfrmsum(1,j)=sum(pfrm(:,j));

    end

    csymb(n)=sign(sum(cfrmsum(1,:)/N));

    psymb(n)=sign(sum(pfrmsum(1,:)/N));

end

clear r;

```

```

clear ynew;
clear frmrec;
clear frmrec2;

%##### ERROR CALCULATION
#####

ccnterr(z,snrdb)=length(find(Mod-csymb'));

pcnterr(z,snrdb)=length(find(Mod-psymb'));

end

end

%##### BIT ERROR RATE
#####

cber(cr,1:snrdb)=sum(ccnterr(:,:))/(z*(Ni));    %BER for CCS-UWB

pber(cr,1:snrdb)=sum(pcnterr(:,:))/(z*(Ni));    %BER for PCS-UWB
end

%##### PERFORMANCE PLOTS
#####

figure,
semilogy(SNR,cber(1,1:snrdb),'k-*',SNR,pber(1,1:snrdb),'k-
s',SNR,cber(2,1:snrdb),'k:+',SNR,pber(2,1:snrdb))
title('BER Performance for Compression Ratio(M/N) more than 35%')
xlabel('SNR (dB)')
ylabel('Average BER')
legend('CCS-UWB 40%', 'PCS-UWB 40%', 'CCS-UWB 60%', 'PCS-UWB 60%')
axis([-17 -4 .0001 1])

```

Appendix C

MATLAB script files used for the simulation for case 3

```
function [SNR cberNf1 pberNf1 cberNf2
pberNf2]=CSUWB_DIFF_FRAMES(crs,Np,Nf1,Nf2,CMN,num_channels)

%CSUWB_DIFF_FRAMES(cr,Np,Nf1,Nf2,ch,num_channel)
%CSUWB_DIFF_FRAMES: Compressed Sensing based Ultra-wideband system with
%      sensing matrix operating for different number of
%      frames per symbol for different compression ratios
%Inputs
%crs      Compression Ratio (M/N), default [0.1 0.2]
%Np      Number of pilot symbols, default 1
%Nf1     Number of frames in each symbol, default 15
%Nf2     Number of frames in each symbol, default 5, Nf1>Nf2
%CMN     Type of channel according to the Reference 21,default 1
%num_channel  Number of channel realizations, default 4000
%Outputs
% Bit Error rate performance of CCS-UWB and PCS-UWB for
% different compression ratios
%SNR     The range of Signal to Noise ratio
%cberNf1  CCS-UWB Nf1 bit error rate for different compression ratios
%pberNf1  PCS-UWB Nf1 bit error rate for different compression ratios
%cberNf2  CCS-UWB Nf2 bit error rate for different compression ratios
%pberNf2  PCS-UWB Nf2 bit error rate for different compression ratios
%Description
%CSUWB_DIFF_FRAMES gives BER performance graph for conventional
%compressed sensing ultra-wideband (CCS-UWB) and proposed compressed
%sensing ultra-wideband (PCS-UWB) for different frames per symbols Nf1 and Nf2
%for different compression ratios (M/N)

if nargin<6
    num_channels=4;
end

if nargin<5
    CMN=1;
end
if nargin<4
    Nf2=5;
end
```



```

if nargin<3
    Nf1=15;
end
if nargin<2
    Np=1
end
if nargin<1
    crs=[0.1 0.2]
end
%num_channels=4;CMN=1; Nf1=15;Nf2=5;Np=1;crs=[0.01 0.05]

%#####PARAMETERS #####

Fs=20e9;          %Digital frequency
fp=2e9;          %Pulse frequency
Tc=10e-9;        %Chip Width
Tf=200;          %Frame duration in Nanoseconds
Ni=10;           %Number of information symbols
Ns=Np+Ni;        %Total number of symbols
SNR=[-15 :1:5];  % SNR in dB

%#####PULSE GENERATION #####

tc = gmonopuls('cutoff', fp); % Pulse width parameter
Tp = -2*tc : 1/Fs : 2*tc;    % Actual pulse width
p = gmonopuls(Tp,fp);        % UWB Pulse

%#####CHIP GENERATION #####

chiptime=0:1/Fs:Tp-1/Fs;     % Chip samples
chip=zeros(length(chiptime),1);
chip(1:length(p))=(p');      % UWB Chip

%##### UWB CHANNEL #####

uwb_h=uwb_channel(CMN,num_channels); % Reference 21

%##### SIMULATION #####

for cr=1:length(crs)          % Iteration for each compression ratio

for snrdb=1:length(SNR)      % Iteration for each SNR
    for z=1:num_channels     % Number of channel realizations

```

```

%##### FRAME GENERATION #####

frame1=conv(uwb_h(:,num_channels),chip);
frame=frame1(1:1200);
N=length(frame);

%##### PILOT SIGNAL #####

sig=zeros(Nf1*N,1);

for j=1:Nf1

    sig((j-1)*N+1:j*N,1)=frame;
end

for n=1:Np
    s2(1:length(sig),n)=sig;
end

%##### TRANSMITTED SIGNAL #####

s=single(zeros(Ns*length(sig),1));
s(1:Np*length(sig))=s2(:);
Mod=(sign(randn(Ni,1)));          % Modulation
s3=single(sig*Mod(1:Ni,1)');
s(Np*length(sig)+1:Ns*length(sig),1)=s3(:); % Modulated TX signal
clear s3;

%#####RX SIGNAL GENERATION #####
rt=SNR(snrdb);                    % SNR value
len=length(s);                    % Length of TX signal
awg=awgnnoise(rt,s,len,fp,Fs);    %AWG Noise generation
r=s+awg;                          % RX signal with AWG Noise
clear s;

%#####Reconstruction Methods#####

M=single(round(crs(cr)*N));        % Number of reduced dimation

cyNf1=zeros(M,1);
pyNf1=zeros(M*Nf1,1);

```

```

cyNf2=zeros(M,1);
pyNf2=zeros(M*Nf2,1);

cfmrecNf1=zeros(N,1);
pfrmrecNf1=zeros(N,1);
cfmrecNf2=zeros(N,1);
pfrmrecNf2=zeros(N,1);

pPhi=zeros(M*Nf1,N);

cPhi=randn(M,N);

for l=1:Nf1

    if l==1;
        pPhi((l-1)*M+1:l*M,1:N)=cPhi;

    else
        pPhi((l-1)*M+1:l*M,1:N)=randn(M,N);
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% RECEIVER %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for n=1:Np

    for j=1:Nf1
        rp(1:N,1)=r((n-1)*length(sig)+(j-1)*N+1:(n-1)*length(sig)+j*N,1);

        %%%%%%%%%%%%%%% Compressed Sampling for ALL
        Methods%%%%%%%%%%%%%%
        if j<=Nf2
            cyNf2=cyNf2+cPhi*rp;           % Conventional method's Compressed Sampled output

            pyNf2((j-1)*M+1:j*M,1)= pPhi((j-1)*M+1:j*M,1:N)*rp(1:N);
            Phinf((j-1)*M+1:j*M,1:N)=pPhi((j-1)*M+1:j*M,1:N);
        end

            cyNf1=cyNf1+cPhi*rp;           % Conventional method's Compressed Sampled output

            pyNf1((j-1)*M+1:j*M,1)= pPhi((j-1)*M+1:j*M,1:N)*rp(1:N);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% RECONSTRUCTION %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

end
end
[cfmrecNf2(1:N,1),iters, activationHist]=SolveOMP(cPhi,cyNf2/Nf2,N,[],0,0);
% Reconstructed frame (CM)
[pfrmrecNf2(1:N,1),iters, activationHist]=SolveOMP(Phinf,pyNf2,N,300,[],0,0,1e-4);

[cfmrecNf1(1:N,1),iters, activationHist]=SolveOMP(cPhi,cyNf1/Nf1,N,[],0,0);
% Reconstructed frame (CM)
[pfrmrecNf1(1:N,1),iters, activationHist]=SolveOMP(pPhi,pyNf1,N,300,[],0,0,1e-4);

%#####Demodulation#####

for n=1:Ni
    for j=1:Nf1

        if j<=Nf2

            cfmNf2(1:N,j)=r((n+Np-1)*length(sig)+(j-1)*N+1:(n+Np-
1)*length(sig)+j*N).*cfmrecNf2;
            cfmsumNf2(1,j)=sum(cfmNf2(:,j));

            pfrmNf2(1:N,j)=r((n+Np-1)*length(sig)+(j-1)*N+1:(n+Np-
1)*length(sig)+j*N).*pfrmrecNf2;
            pfrmsumNf2(1,j)=sum(pfrmNf2(:,j));

            cfmNf1(1:N,j)=r((n+Np-1)*length(sig)+(j-1)*N+1:(n+Np-
1)*length(sig)+j*N).*cfmrecNf1;
            cfmsumNf1(1,j)=sum(cfmNf1(:,j));

            pfrmNf1(1:N,j)=r((n+Np-1)*length(sig)+(j-1)*N+1:(n+Np-
1)*length(sig)+j*N).*pfrmrecNf1;
            pfrmsumNf1(1,j)=sum(pfrmNf1(:,j));

        end

    end

    csymbNf1(n)=sign(sum(cfmsumNf1(1,:)/N));
    psymbNf1(n)=sign(sum(pfrmsumNf1(1,:)/N));

    csymbNf2(n)=sign(sum(cfmsumNf2(1,:)/N));
    psymbNf2(n)=sign(sum(pfrmsumNf2(1,:)/N));

```

```

end

clear r;

%#####ERROR CALCULATION #####

ccnterrNf1(z,snrdb)=length(find(Mod-csymbNf1'));
pcnterrNf1(z,snrdb)=length(find(Mod-psymbNf1'));

ccnterrNf2(z,snrdb)=length(find(Mod-csymbNf2'));
pcnterrNf2(z,snrdb)=length(find(Mod-psymbNf2'));

end

end

%##### BIT ERROR RATE #####

cberNf1(cr,1:snrdb)=sum(ccnterrNf1(:,:))/(z*(Ni));

pberNf1(cr,1:snrdb)=sum(pcnterrNf1(:,:))/(z*(Ni));
cberNf2(cr,1:snrdb)=sum(ccnterrNf2(:,:))/(z*(Ni));

pberNf2(cr,1:snrdb)=sum(pcnterrNf2(:,:))/(z*(Ni));
end

semilogy(SNR,cberNf2(1,1:snrdb),'k-+',SNR,cberNf1(1,1:snrdb),'k-
s',SNR,cberNf2(2,1:snrdb),'k:+',SNR,cberNf1(2,1:snrdb),'k:s',SNR,pberNf2(1,1:snrdb),'k
--+',SNR,pberNf1(1,1:snrdb),'k--s',SNR,pberNf2(2,1:snrdb),'k-
.+ ',SNR,pberNf1(2,1:snrdb),'k-.s')
title('10%')
xlabel('SNR (dB)')
ylabel('Average BER')
axis([-15 5 .00005 1])

```

Appendix D

Permission for using figure 1.1 in the thesis

Zimbra Collaboration Suite

jodhbirs@mtu.edu

Re: Request for the Copyright
Permission

Wednesday, January 19, 2011 2:46:45
PM

From: pavlenko.artem@gmail.com

To: jodhbirs@mtu.edu

Hello Jodhbir,

Yes, you may reprint the figure. Just please mention that it's taken from our website.

Good luck on your thesis!

Best regards,
Artem Pavlenko,
Managing Editor,
iXBTLabs.com (former www.Digit-Life.com)

On Wed, Jan 19, 2011 at 22:38, Jodhbir Singh <jodhbirs@mtu.edu> wrote:

Hello

I am Jodhbir Singh, completing a Master's thesis at Michigan Technological University entitled "Compressed Sensing Based Ultra-Wideband Communication Hardware Reduction." I would like your or **Alexander Medvedev** (unclesam@ixbt.com) permission to reprint in my thesis figure from **Alexander Medvedev's** following article:

Medvedev, A. (2003) *Following the IDF: Ultra Wide Band Wireless Data Transfer Technology*, www.digit-life.com/articles2/uwb/?11789.

The figure to be reproduced is *Emitted Signal Power vs. Frequency(Hz)*

The given e-mail address of **Alexander Medvedev** is not working. In case I should request permission directly from him, then please provide me with his new e-mail address or forward this e-mail to him/her.

The requested permission extends to any future revisions and editions of my thesis, including non-exclusive world rights in all languages, and to the prospective publication of my thesis by UMI (ProQuest). These rights will in no way restrict republication of the material in any other form by you or by others authorized by you. Your or **Alexander Medvedev** permission will also confirm that you own [or Byrds Research & Publishing Ltd] the copyright to the above-described material.

If these arrangements meet with your approval, please reply to this e-mail with agreement. Thank you very much.

Sincerely,

Jodhbir Singh

MS candidate
Department of Electrical and Computer Engineering
Michigan Technological University
Houghton, MI 49931