Understanding the Practical Limitations of Applying Analog Compressed Sensing Systems to ECG Signals

Anna Marie Rogers Dixon

A dissertation submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2012

Reading Committee:

David J. Allstot, Chair

Robert B. Darling

Mani Soma

Program Authorized to Offer Degree:
UW Electrical Engineering

University of Washington

Abstract

Understanding the Practical Limitations of Applying Analog Compressed Sensing Systems to ECG Signals

Anna Marie Rogers Dixon

Chair of the Supervisory Committee:
Professor David J. Allstot
Electrical Engineering

Body area networks (BAN), networks of wearable and wireless physiological sensors, are expected to have a profound positive impact in healthcare. The bio-signal sensors are equipped with ultra-low power radios communicating to a BAN personal base station, and ultimately the healthcare provider. Most of the power dissipated in a state-of-the-art bio-signal sensor occurs when the RF power amplifier transmits data to the personal base station. Thus, a method is desired that decreases the amount of data to be transmitted which reduces the duty cycle of the power amplifier and increases the overall energy efficiency.

Compressed sensing (CS) is a compression scheme capable of significantly reducing a signal acquisition's data rate. CS requires only a few incoherent measurements to compress signals that are sparse in some domain. Since compressed sensing is still an emerging topic, only a handful of CS systems have been realized in hardware. These systems have shown promising and yet limited abilities. The objective of this research is to provide designers with a roadmap that enables them to more easily make correct decisions in designing analog CS encoders and decoders for bio-signals. By showing the impact of the considerations of this CS system on ECG signals, it will set up a framework for how to approach and/or analyze the design of these systems for all bio-signals. The CS roadmap accomplishes this goal this by demonstrating the importance of signal sparsity, guides the design of sensing matrix generation, addressing the impact of several analog CS imperfections on CS compression and guides the selection of proper CS reconstruction algorithms.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

ADC: Analog to Digital Converter

ASIC: Application Specific Integrated Circuit

BAN: Body Area Network

BP: Basis Pursuit

BPDN: Basis Pursuit Denoising

COSAMP: Compressive Sampling Matching Pursuit

CS: Compressed Sensing or Compressive Sampling

DAC: Digital to Analog Converter

DWT: Discrete Wavelet Transform

ECG: Electrocardiogram

EEG: Electroencephalogram

EMG: Electromyogram

HALT: High Accuracy Low Thresholding

HCT: High Compression and Thresholding

IID: Independent and Identically Distributed

LFSR: Linear Feedback Shift Register

LSB: Least Significant Bit

MDAC: Multiplying Digital to Analog Converter

MSE: Mean Squared Error

NIHT: Normalized Iterative Hard Thresholding

OMP: Orthogonal Matching Pursuit

PA: Power Amplifier

PDF: Probability Density Function

PMF: Probability Mass Function

PRD: Percentage Root Mean Square Difference

PRG: Pseudorandom Number Generator

SNR: Signal to Noise Ratio

SQNR: Signal to Quantization Noise Ratio

STFT: Short Time Fourier Transform

ACKNOWLEDGMENTS

First, I would like to acknowledge and thank my advisor, Prof. David Allstot. He took a chance on me for which I am eternally grateful. His support through the PhD process was unwavering. His confidence in my abilities often exceeded my own, which pushed me to achieve feats I didn't realize possible. He is a phenomenal researcher and mentor and working with him has been inspirational. Without him, none of this would have been possible.

I would like to thank Prof. Bruce Darling, Prof. Brian Otis and Prof. Mani Soma. They generously took the time to serve on my PhD committee and help improve my work. I am also grateful to have benefitted from their excellent instruction in class. I like to think that their guidance in the classroom and on this committee have shaped me into a better researcher and engineer. I would also like to thank Prof. Krishnamurthy for being the GSR. I am so thankful that he was willing to help at the last minute.

The UW SoC Lab was a great environment to work in these last few years. I am thankful all it's members and their support. I am especially grateful to those I had the honor to collaborate with: Emily Allstot, Andrew Chen, Daibashish Gangopadhyay, Heather Mitsuda and Aabeya Salman.

I would also like to acknowledge the Semiconductor Research Corporation. Their fellowship opportunity and research guidance have been instrumental to the PhD process.

Last but not least, my family including my husband, Colin, my parents, my brother and grandparents have always provided the endless love and support that makes anything possible.

DEDICATION

I dedicate this thesis to my dear husband, Colin, who always believed in me.

Chapter 1

INTRODUCTION

Soaring healthcare costs are becoming a burden on the economy, accounting for 15.2% of the gross domestic product in the United States in 2008 [3]. Moreover, healthcare costs are expected to continue growing. One method for improvement is more persistent patient monitoring. To be effective, this persistent monitoring needs to analyze a patient's physiological data during healthy and unhealthy periods (i.e. beyond just the healthcare provider's office). The long-term goal of persistent patient is personalization (and more effective) healthcare. In short, better fusion between engineering, technology and medicine is paramount.

Wearable health monitoring systems are an effective tool already used by the healthcare community. Patients who benefit from continuous health monitoring include those recovering from surgery, require monitoring of an ongoing illness or those in need of more rigorous diagnosis. However, these systems are far from perfect. A trade-off typically exists between high fidelity (accuracy) and longevity (power consumption). The existing health monitoring systems stand to benefit immensely from ongoing biomedical engineering research. This work aims to explore the advantages and disadvantages of one method, compressed sensing, as a solution for high-accuracy low-power ECG sensors.

1.1 Applying Compressed Sensing to Wireless Body Area Networks

Mobile personal health monitoring devices are already widely used by the healthcare community. The Holter monitor, a portable electrocardiography (ECG) monitor, is one such example. The Holter monitor records the electrical activity of the patient's heart through two, three or twelve electrodes (depending on patient needs). The signals are transmitted from the electrodes via wire to a recorder worn by the patient. The data is later uploaded to be analyzed by healthcare professionals. One major drawback of the Holter monitor system described is that the wires between the electrodes and the recorder are uncomfortable for the patient and may even alter the patient's movement and ac-

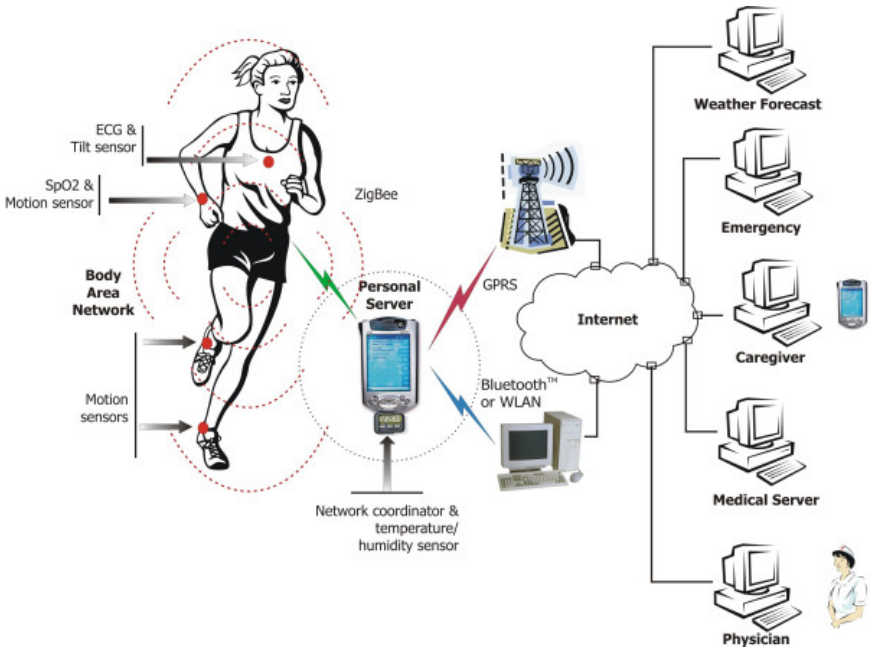


Figure 1.1: Wireless Body Area Network

tivity, potentially negatively impacting the measurements. A solution is to employ wireless sensors. Unfortunately, wireless sensors are power hungry. Another potential area of improvement for this personal health monitor is real-time data monitoring. In other words, rather than occasionally uploading physiological data for further analysis, it would be beneficial to have real-time monitoring to better detect medical emergencies.

A body area network (BAN) is a network of *wireless* physiological sensors used to monitor patient health. These sensors may include various motion, inertial and bio-signal (ECG, EEG, EMG, etc.), depending on the patients needs. Figure 1.1 is an example of a BAN [4]. Each sensor is equipped with a radio that communicates the biomedical data to a personal base station (i.e. a smartphone) and then to the healthcare provider via the internet. The BAN addresses the aforementioned drawbacks of the existing monitors. The wireless sensors provide continuous real-time patient monitoring with a higher level of patient comfort than it's wired counterpart. However, the benefits of BANs comes at the cost of increased power consumption. Increased power consumption at the sensor ultimately means increased heat dissipation (causing discomfort for the patient) and

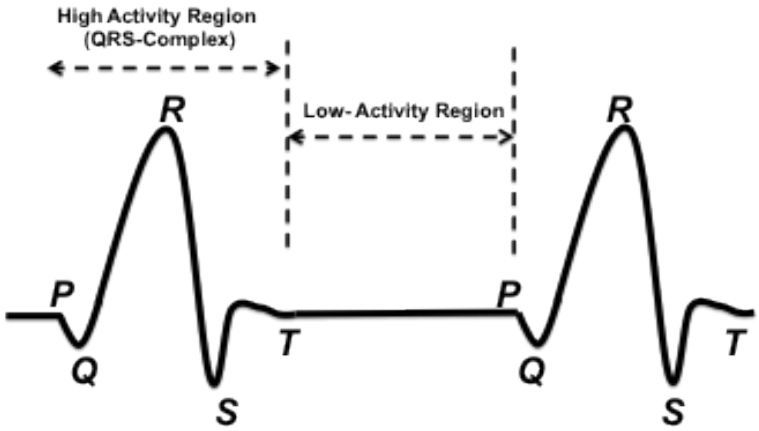
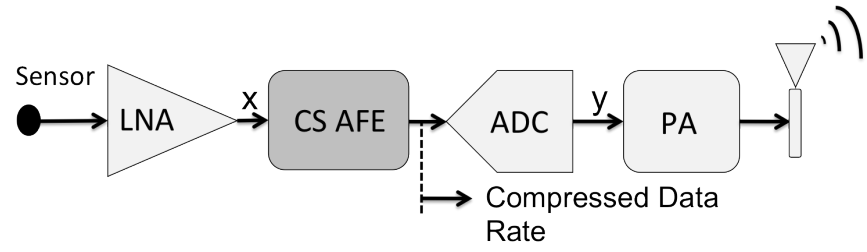


Figure 1.2: Time Domain Sparse ECG Waveform

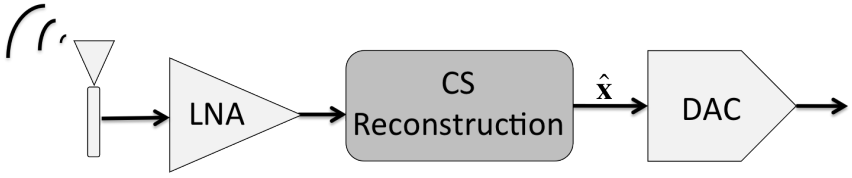
decreased battery life. As a consequence, much of the recent research into BANs focuses on how to reduce the power requirements of the BAN approach.

In the typical wireless bio-signal sensor, the bio-signal is first amplified by an ultra-low power low-noise amplifier. Following that, the ADC digitizes the signal and the RF power amplifier (PA) transmits the data to the personal base station. In this system, most of the power is dissipated when the RF PA is transmitting. For example, the PA of a BAN transmitter developed for the Medical Implant Communications Service (MICS) standard consumes 66.5μW (i.e., 74%) of the total power of 90μW [5]. Thus, the most significant power savings comes from decreasing the amount of data to be transmitted which reduces the duty cycle of the transmitter and increases the overall energy efficiency.

Compressed sensing (CS), or compressive sampling, is a data acquisition approach that requires only a few incoherent measurements to compress signals that are sparse in some domain [6]. For example, Figure 1.2 shows how a typical ECG bio-signal that comprises a high activity QRS complex followed by a low activity region between complexes can be considered sparse in the time domain. CS compresses N analog input samples into M ($<< N$) analog values. In doing so, it is possible to reduce the duty cycle and the corresponding energy consumption of a BAN radio [5] by the achieved compression factor $C = \frac{N}{M}$.



(a) CS Encoder



(b) CS Decoder

Figure 1.3: Bio-Signal CS CODEC

In a CS CODEC, the transmitter (Figure 1.3(a)) receives the bio-signal from a low-noise amplifier. The analog front-end (AFE) compresses the N-sample analog input vector, $\boldsymbol{x}$, into an M-sample ($M < N$) analog vector, $\boldsymbol{y}$. Consequently, the ADC and RF PA can operate with a lower duty cycle. The receiver (Figure 1.3(b)), which may reside in a smartphone, uses basis functions and optimization to recover the original signal.

CS is a promising solution to lower power BAN sensors. In fact, a few CS encoder designs and implementations already exist. However, the design details are highly variant. A more detailed description of these designs, their variations and performance is provided in Section 2.3. In fact, the existing designs are so disparate, that it is too difficult to discern which implementation details are important and why.

1.2 Thesis Contributions

Compressed sensing is a relatively new field of research with few working implementations (Section 2.3). Consequently, little is known about the failure modes and optimizations of working CS systems. The objective of this research is to provide designers with a roadmap that enables them to

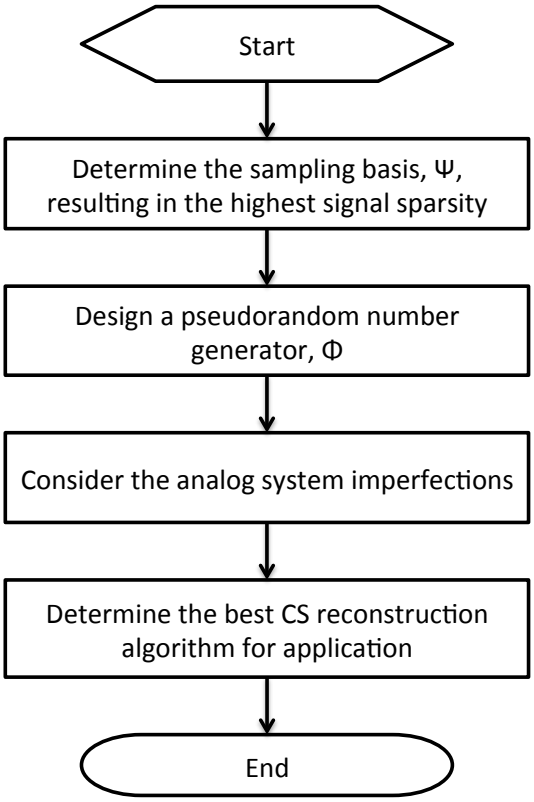


Figure 1.4: Analog CS System Design Roadmap

more easily make correct decisions in designing analog CS encoders and decoders for bio-signals. By showing the impact of the considerations of this CS system on ECG signals, it will set up a framework for how to approach and/or analyze the design of these systems for all bio-signals.

Figure 1.4 presents the analog CS system design roadmap that embodies this research. This thesis will show the importance of, and how to optimize for, the following core CS design concepts:

- **Sampling Basis** The success of compressed sensing is based on the requirement that the signal to be sparse. Fortunately, the sparsity restriction can apply in any reversible basis. This dissertation will show the impact of sampling basis selection on signal sparsity, and consequently the compression ratio.

- **Random Sensing Matrix Generation** Compressed sensing systems operate by multiplying

the signal by a random sensing matrix. CS dictates that the sensing matrix be incoherent with the sampling basis matrix (defined in Section 2.1.2). This thesis will explore the coherence between several sampling bases and random distribution functions. The work will then progress to explore the impact of the coherence on signal recovery accuracy. Existing CS implementations use on-chip pseudorandom number generators for sensing matrix generation. This dissertation will show the coherence achieved by various pseudorandom number generators and their overall effectiveness.

- **Analog System Imperfections** While CS is a proven effective mathematical tool, it is still unclear how circuit non-idealities will impact CS performance. This dissertation will work to uncover the challenges presented by the CS AFE design including quantization, offset, gain error and non-linearity.

- **CS Reconstruction** Several CS signal reconstruction algorithms have been presented. Convex optimization and greedy algorithms alike are used to recover CS compressed signals. This dissertation presents a thorough comparison of the state-of-the-art CS reconstruction algorithms and promotes the selection of certain algorithms based on the driving goal of the decoder (accuracy, computation time, etc.).

1.3 Organization

The organization of this thesis is as follows. Chapter 2 provides background material key to the reader's understanding of compressed sensing and existing CS encoders [7]. Each successive chapter covers the main thesis contributions. Chapter 3 presents proper sampling basis selection. This concept is demonstrated by presenting the effects of utilizing time and wavelet bases on ECG signals. Chapter 4 presents the impact of random sensing matrix generation. Chapter 5 discusses the challenges associated with the analog CS system. A comparison of various CS reconstruction algorithms are presented in Chapter 6. Chapter 7 draws conclusions from this dissertation and offers directions for future work.

Chapter 2

BACKGROUND

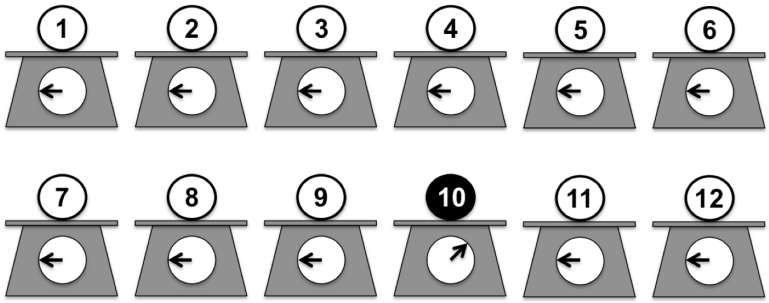
Compressed sensing is a revolutionary signal acquisition scheme that allows a signal to be acquired and accurately reconstructed with significantly fewer samples than required by Nyquist-rate sampling. Unlike Nyquist sampling, which relies on the maximum *rate-of-change* of a signal, compressed sensing relies on the maximum *rate-of-information* in a signal. Compressed sensing has been instrumental in research for low-power data acquisition methods. Section 2.1 reviews the basic concepts of CS. Section 2.1.1 provides an intuitive view of the CS concept and Section 2.1.2 gives the formal definition of CS. Section 2.2 presents and compares an alternative sampling technique, adaptive sampling, followed by a summary of the exiting CS encoders in Section 2.3. Finally, Section 2.4 introduces the performance metrics used throughout this thesis to measure CS success.

2.1 Compressed Sensing

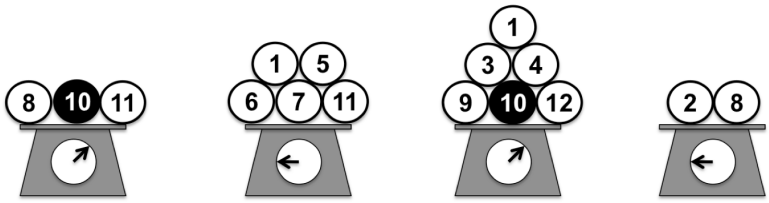
2.1.1 Intuitive View of Compressed Sensing

Imagine an experiment with many objects in which only one element is "different" (i.e., sparse). The goal is to identify the sparse object(s) with minimal effort. In traditional testing, each object is measured individually. In group testing, the large difference between the total number of objects in the set and the number of different objects is exploited by examining only a few groupings. Consequently, the sparse object is found with fewer tests. The traditional and group testing techniques are compared in Fig. 2.1. In this puzzle, there is a set of 12 balls wherein one is much heavier (100g) than the others (1g). The objective is to identify the heavy ball (Ball #10, the black ball). In order to present an appropriate analogy to sparse signal acquisition and the application of compressed sensing, the measurement protocol is assumed to be non-adaptive. That is, once the measurement process has commenced, it will not cease until completed even upon obvious identification of the heavy ball.

Figure 2.1(a) demonstrates the conventional sampling approach, wherein each object is mea-



(a) Conventional Sampling



(b) Group Sampling

Figure 2.1: 12-Ball Example of Sampling Methods

sured individually. The corresponding matrix representation is:

$$\begin{bmatrix} 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 100g \\ 1g \\ 1g \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 100g \\ 1g \\ 1g \end{bmatrix} \tag{2.1}$$

That is, the output vector $\mathbf{y}$ equals the measurement matrix $\mathbf{\Phi}$ times the input vector $\mathbf{x}$. $\mathbf{\Phi}$ defines the measurement protocol, which is an identity matrix in this example because only one ball is measured at a time. Thus, traditional sampling requires N measurements for N balls.

Compressed sensing takes the group testing approach to the problem, illustrated in Figure 2.1(b), wherein groups of balls are measured instead of individual measurements. In group testing, the large difference between the total number of objects and the number of different objects (i.e., the sparsity) is exploited by examining only a few groupings. In matrix form, the compressed sensing approach to this example is given by:

$$\begin{bmatrix} 102g \\ 5g \\ 104g \\ 2g \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 1g \\ 100g \\ 1g \\ 1g \end{bmatrix} \tag{2.2}$$

The measurement matrix here implements random groupings of measurements. For very sparse signals, random measurements will accurately capture the required data with high probability. The CS measurement matrix $\mathbf{\Phi}$ comprises coefficients sampled from a random probability distribution function (pdf) such as Gaussian, Uniform, etc. The 1-bit random coefficients above are from a Bernoulli pdf where 0(1) represents the absence (presence) of a specific ball. The resulting number of measurements is much smaller than the number of elements in the signal. In fact, by implementing the group testing/compressed sensing method here, the number of measurements needed to identify and measure the "odd" ball has been decreased by a factor of 3!

At signal recovery, the decoder receives the compressed vector $\mathbf{y}$ and reconstructs the original

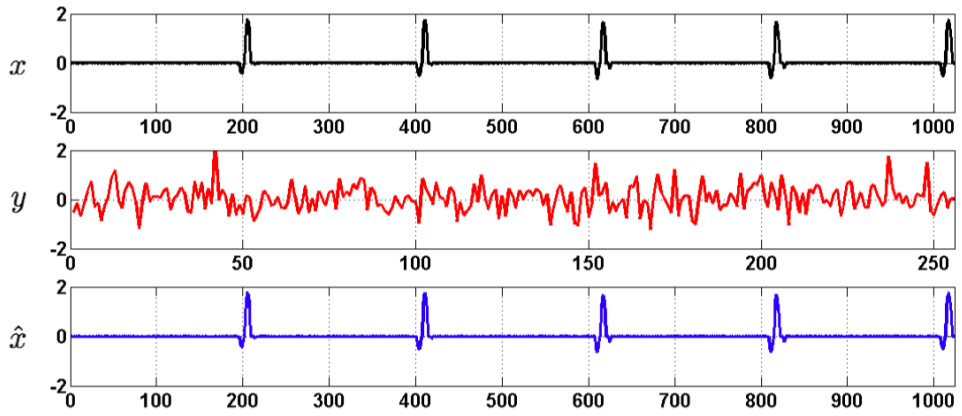


Figure 2.2: Compressed Sensing of an ECG Signal

signal. For example, here it can be observed that Ball #10 is the only commonality between the two measurements showing an abnormally large measurement. It is therefore deduced (at the decoder) from 4 total measurements of groups of balls that Ball #10 is the "odd" ball.

2.1.2 Formal Definition of CS

The compressed sensing system of Figure 1.3 performs two fundamental computational operations: encoding (or compression) and decoding (or reconstruction). Representative signals for the CS system of a typical ECG bio-signal (thresholded to reduce baseline drift and noise) after the acquisition, compression, and reconstruction stages are shown in Figure 2.2. In this example, the analog input vector, $\mathbf{x}$, comprises 1024 samples whereas and the compressed analog output vector, $\mathbf{y}$, has 256 samples after CS compression; a compression factor of C=4 is achieved. At the decoder, optimization algorithms are used in the decoder to reconstruct the input vector, $\hat{\mathbf{x}}$.

Compressed sensing is the non-adaptive sampling scheme embodied in the simple matrix equation

$$\mathbf{y} = \mathbf{\Phi x} \tag{2.3}$$

where $\mathbf{x}$ is the analog time-domain input sample vector of length N, $\mathbf{y}$ is the compressed analog

output vector of length M, and $\mathbf{\Phi}$ is the $M \times N$ measurement matrix [6]. CS is considered *non-adaptive* because $\mathbf{\Phi}$ remains constant.

In some cases, as described below, the analog input vector, $\mathbf{x}$, can be further expanded by:

$$\mathbf{x} = \mathbf{\Psi}\boldsymbol{\alpha} \tag{2.4}$$

where $\boldsymbol{\alpha}$ is the N-sample coefficient sequence and $\mathbf{\Psi}$ is the $N \times N$ sampling basis or sparsifying matrix [6]. For example, the ECG signal in Figure 2.2 was sampled and compressed in the time domain. The representation basis matrix, $\mathbf{\Psi}$, for the time-domain is simply the identity matrix. If however, $\mathbf{x}$ is able to achieve greater CS compression in the wavelet domain, $\mathbf{\Psi}$ is the inverse wavelet transform matrix that converts the wavelet coefficients, $\boldsymbol{\alpha}$, into it's corresponding input time signal. The concept can be extended to any sampling basis (Fourier, wavelets, chirplets, etc).

Sometimes it is useful to combine Equations 2.3 and 2.4:

$$\mathbf{y} = \mathbf{\Theta}\boldsymbol{\alpha} \tag{2.5}$$

where $\mathbf{\Theta} = \mathbf{\Phi}\mathbf{\Psi}$. $\mathbf{\Theta}$ is commonly referred to as the sensing matrix.

CS captures $M << N$ measurements from N samples using random linear projections [6]. The advantage of CS lies in the compression factor:

$$C = \frac{N}{M} \tag{2.6}$$

In order to implement a CS algorithm correctly, there are two key requirements:

1. The sampled signal, $\mathbf{x}$, must the sparse in some basis $\mathbf{\Psi}$.

2. The sampling matrix, $\mathbf{\Psi}$, and the measurement matrix, $\mathbf{\Phi}$, must be incoherent.

The signal sparsity constraint observes that for signals compressible by CS, there is a substantial difference between the signal's information rate and rate of change. The signal $\mathbf{x}$ is sparse when all but a few elements in the vector are zero (or very near to zero). If k elements of $\mathbf{x}$ are considerably larger than the other N-k elements, then $\mathbf{x_k}$ is formulated by keeping the k largest elements and setting all others equal to zero. The signal is said to be k-sparse when k elements are non-zero.

A broader definition of sparsity, used frequently in this work, is defined as the percentage of zero elements contained in the signal:

$$\text{Sparsity}(\%) = \frac{N-k}{N}(100\%) \tag{2.7}$$

The sparsity constraint is key because $\mathbf{x}$ is well approximated by $\mathbf{x_k}$ and the error $\|\mathbf{x} - \mathbf{x_k}\|_2$ is therefore small. The sparsity constraint explains the importance of the sampling basis matrix $\mathbf{\Psi}$. Increased signal sparsity results in higher CS compression. The sampling basis matrix is exploited to transform the input signal into the domain with the highest signal sparsity, thus achieving higher compression. Sparsity in some basis is extraordinarily common for bio-signals. Section 3 presents a further investigation into the optimum sampling basis for CS compression of ECG signals.

The CS incoherence requirement guides the $\mathbf{\Phi}$ formulation. Coherence between the sampling matrix $\mathbf{\Psi}$ and the measurement matrix $\mathbf{\Phi}$ is calculated as

$$\mu(\Psi, \Phi) = \sqrt{n} \cdot \underset{0 \leq j,k < n}{max} |\langle \psi_j, \phi_k \rangle| \tag{2.8}$$

where $\boldsymbol{\psi}$ is a column of $\mathbf{\Psi}$ and $\boldsymbol{\phi}$ is a row of $\mathbf{\Phi}$. If $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$ are unit vectors, then $\mu \in [1, \sqrt{n}]$. In other words, coherence measures the largest correlation between any column of $\mathbf{\Psi}$ and row of $\mathbf{\Phi}$. The incoherence between $\mathbf{\Psi}$ and $\mathbf{\Phi}$ spreads out the measurement to ensure maximal capture of the signal. As shown in Figure 2.2, the incoherence between $\mathbf{\Psi}$ and $\mathbf{\Phi}$ allows a very sparse signal in the $\mathbf{\Psi}$ domain to become densely populated (dense in information) as a compressed signal, $\mathbf{y}$. To achieve accurate and numerically stable compression and reconstruction, Candes recommends a minimum number of compressed measurements based on the sparsity and coherence constraints outlined [6]:

$$M \geq C \cdot \mu^2(\mathbf{\Psi}, \mathbf{\Phi}) \cdot k \cdot log(N) \tag{2.9}$$

where C is a small constant.

It is fortunate that random matrices are, with high probability, largely incoherent with any fixed basis $\mathbf{\Psi}$ [6], [8]. As a consequence, the measurement matrix can be populated with random values from many different probability density functions including Bernoulli, Gaussian and Uniform

distributions. Chapter 4 presents further investigation into the optimum measurement matrix and possible hardware implications for CS encoding of ECG signals.

Decoding or signal recovery is another key challenge in the implementation of a complete CS system. Ideally, reconstruction can be posed as solving $\hat{\mathbf{x}} = \mathbf{\Phi}^{-1}\mathbf{y}$. However, there are N unknown values in the reconstructed signal, $\hat{\mathbf{x}}$, but only M known values in the measured signal, $\mathbf{y}$ with $M < N$. Because $\mathbf{\Phi}$ is non-square and thus non-invertible, the problem is under-determined with many possible solutions. The answer to the CS under-sampling dilemma is that the solution must also be sparse, which greatly reduces the set of possible solutions. The ℓ_1-norm is a measure of sparsity; hence, the following minimization provides a reasonable CS signal recovery [6]:

$$min\|\hat{\boldsymbol{\alpha}}\| \text{ subject to } \mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\hat{\boldsymbol{\alpha}} \tag{2.10}$$

The sparsity and incoherence constraints ensure, with high probability, that the solution to Equation 2.10 is exact; i.e., $\hat{\boldsymbol{\alpha}} = \boldsymbol{\alpha_k}$.

2.2 Adaptive Sampling Versus Compressed Sensing

Conventional ECG monitoring systems use a constant sampling rate (typically 256 Hz) as determined by the rate of change of the signal in the QRS complex (Figure 1.2). Consequently, the traditional method accurately captures the high frequency information but performs unnecessary sampling on the low frequency information in the periods between QRS complexes. This section discusses an existing ECG compression technique, called adaptive sampling, and makes a comparison to CS.

Adaptive sampling is a method in which the sampling rate is adapted continuously to the activity level of the signal [9], [10]; Figure 2.3 exemplifies the technique applied to an ECG signal. An activity detection algorithm and circuits are used to identify the high frequency regions of the signal and adjust the sampling rate accordingly [10]; specifically,f_s, is either a low, f_l, or high, f_h, frequency as shown. Hence, the signal is effectively sampled at a lower rate, which reduces power dissipation and maintains adequate accuracy for clinical applications.

For ultra-low power implementations, the achievable data compression draws an important distinction between the two approaches. The effective digitization rate for adaptive sampling can be

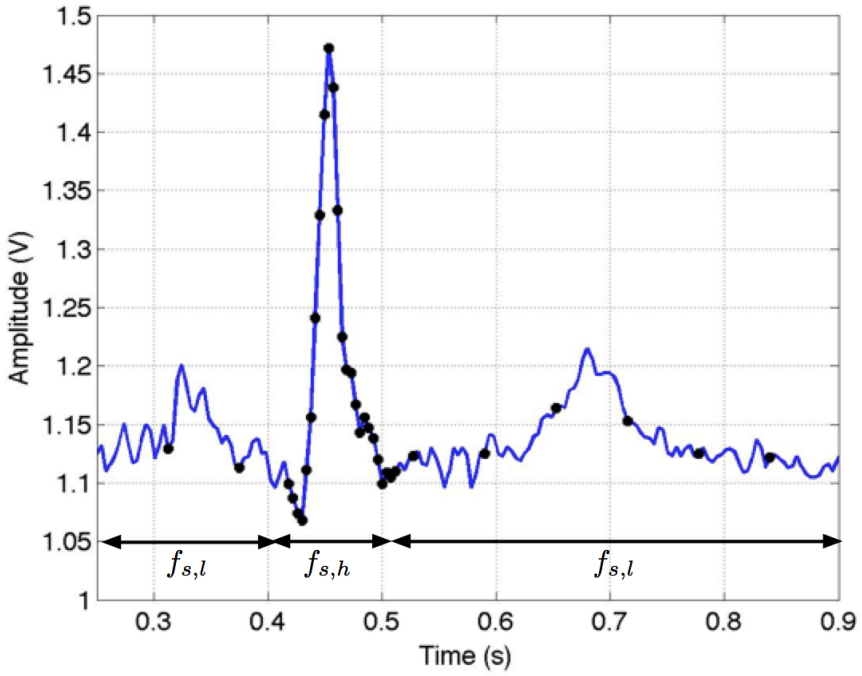


Figure 2.3: Adaptive Sampling applied to an ECG Signal

expressed as:

$$f_{s,eff} = x f_{s,h} + (1 - x) f_{s,l} \tag{2.11}$$

where x, the fraction of time in which an active event is detected, is tuned based on the desired level of detail. In contrast, the digitization rate for a CS implementation is fixed:

$$f_{s,eff} = \frac{M}{N} f_s \tag{2.12}$$

Unlike CS, an adaptive sampling system requires a-priori knowledge of the signal which ultimately its versatility; the activity detection heuristic should be designed to encompass the required range of sampling frequencies. On the other hand, CS operates with no a priori knowledge of the signal. Therefore, it is universally applicable to ECG, EMG, electroencephalogram (EEG), and other bio-signals.

	Adaptive Sampling	Compressed Sensing
Sampling Rate	$f_{s,eff} = xf_{s,h} + (1-x)f_{s,l}$	$f_{s,eff} = \frac{M}{N}f_s$
A-priori Knowledge	Yes	No
Additional Sensor Architecture	Activity Detection Circuit and Sampling Control Mechanism	Thresholding Circuit

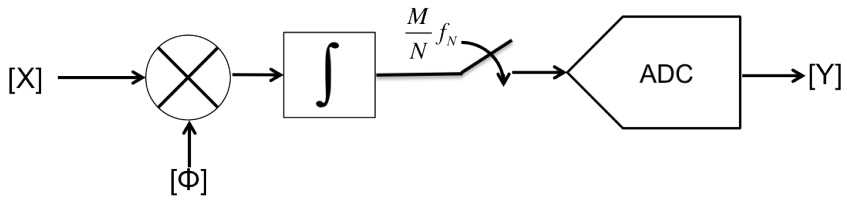
Table 2.1: CS/Adaptive Sampling Comparison Summary

A final consideration relates to the hardware differences between the two approaches. The adaptive sampling approach requires activity detection circuitry (e.g., a differentiator and a comparator [10]) before the digitization circuitry, and a microcontroller to tune the activity detector. In contrast, a CS implementation may require a thresholding block allowing a remote healthcare professional to send realtime feedback to the bio-sensor to trade off sparsity (i.e., acquisition accuracy) versus compression factor (i.e., energy efficiency). The CS encoder needs M multiply-and-accumulate (MAC) circuits to process the MxN measurement matrix with the N-sample analog input vector.

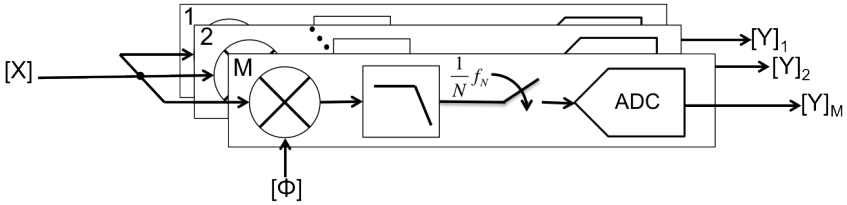
A summary of the implementation differences between adaptive sampling and CS is given in Table 2.1.

2.3 Existing CS Encoder Solutions

Compressed sensing has been proven to have many applications. Perhaps one of the earliest and most famous applications is the single pixel camera, achieved by taking the CS random linear projections using a digital micromirror array in conjunction with only a single detection element [11]. This section, however, will focus more particularly at CS implementations whose encoding all occurs on-chip. In other words, the CS encoder solutions all aim to perform the CS operation of Equation 2.3 electronically rather than employing a separate physical mechanism. First, the implementation issues and performance characteristics of two analog-domain CS architectures intended for spectral estimation (e.g., cognitive radios) are outlined in Section 2.3.1. Then Section 2.3.2 presents a digital-domain and an analong-domain CS encoder used for bio-signal compression.



(a) Random Demodulation [12], [13]



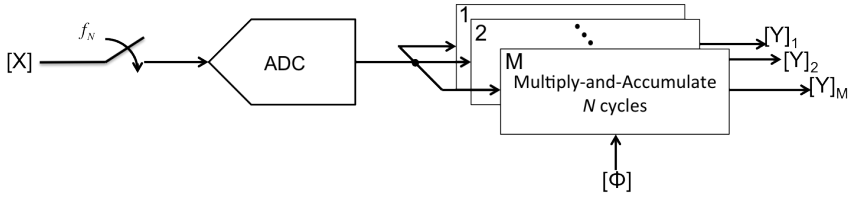
(b) Modulated Wideband Converter [14]

Figure 2.4: Existing Spectral Estimation CS Encoder Systems

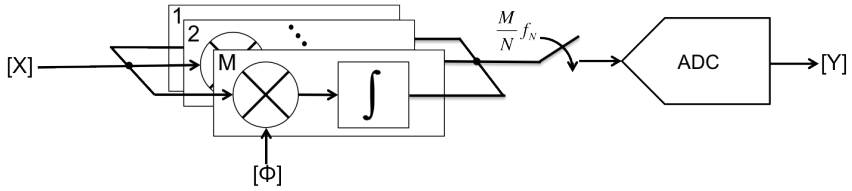
2.3.1 Spectral CS Encoders

Laska, et al. [12] presented an analog-domain CS system random demodulation (RD) (Figure 2.4(a)) and applied it to sparse multi-tone signals. The RD system first mixes the analog signal, $\boldsymbol{x}$, with a random signal, $\boldsymbol{\Phi}$, using a Gilbert multiplier. The resulting signal is then integrated over M successive windows using an active RC integrator. The output, $\boldsymbol{y}$, is digitized by an ADC. HSPICE simulations using 0.13 μm CMOS device models show a sampling rate 6X less than the Nyquist-rate for an AM signal with three tones. The approach was subsequently validated in a hardware prototype by Ragheb, et al. [13]. Measurements for the same three-tone AM input signal showed SNR values of 45.5 dB, 42.9 dB and 39.6 dB for sampling frequencies of 1/2, 1/4 and 1/8 of the Nyquist rate, respectively.

Mishali, et al. [14], [15] proposed another analog-domain CS system (Figure 2.4(b)), a modulated wideband converterter (MWC) - for the compression of sparse multiband signals. In the MWC, a sequence of N samples, $\boldsymbol{x}$, are applied in parallel to an array of M analog mixers which multiply $\boldsymbol{x}$ with a random signal $\boldsymbol{\Phi}$. Each mixer output is then integrated by a low-pass filter, and the output, $\boldsymbol{y}$, is digitized by an ADC. A hardware prototype processed input signals with a band-



(a) Digital CS [17], [18]



(b) Analog CS [1]

Figure 2.5: Existing Bio-Signal CS Encoder Systems

width of 120 MHz. The compressed output was accurately sampled at 280 MHz compared to the Nyquist sampling frequency of 2 GHz. A comparison of RD [12] and MWC [14] is given by Lexa, et al. [16].

2.3.2 Bio-Signal CS Encoders

Chen, et al. [17], [18] recently presented digital-domain compressed sensing chip for EEG and ECG signal compression (Figure 2.5(a)). The digital CS system first digitizes the conditioned bio-signal using a Nyquist rate ADC. A sequence of N samples, $\boldsymbol{x}$, is then applied in parallel to an array of M multiply-and-accumulate (MAC) stages to generate the M compressed output samples, $\boldsymbol{y}$. The 8-bit SAR ADC followed by 50 16-bit MACs integrated in 90 nm CMOS occupied an area of 200 x 550 μm^2. The power dissipation is only 1.9 μW (Vdd = 0.6 V) for EEG signals with compression factors from 10X to 40X.

D. Gangopadhyay [1] presented an analog-domain CS chip for bio-signal compression (Figure 2.5(b)). The analog CS system first compressed the analog bio-signal by passing a sequence of N samples, $\boldsymbol{x}$, through a parallel array of M multiply and integrate stages to generate the M compressed output samples, $\boldsymbol{y}$. The multiplication and integration step is performed by a multiplying-

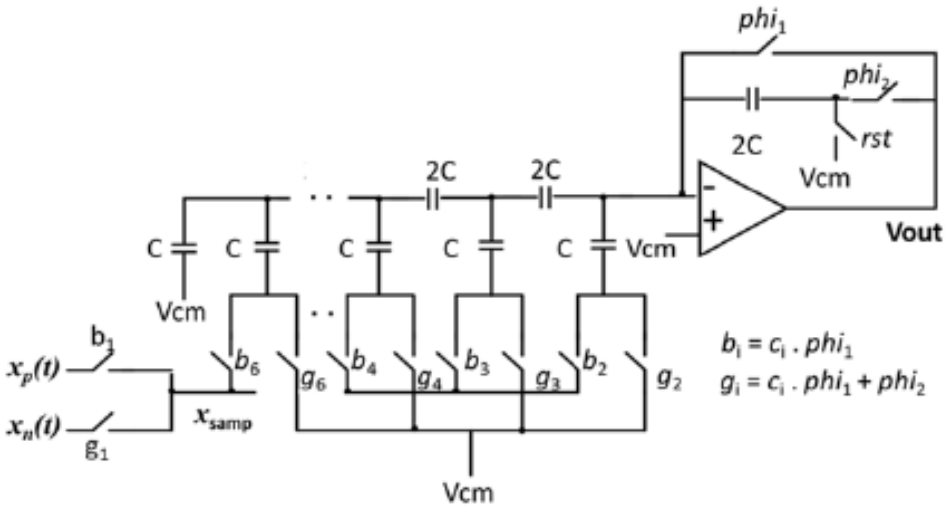


Figure 2.6: Analog CS MDAC/Integrator [1]

digital-to-anaolg-converter (MDAC). The MDAC, shown in Figure 2.6, uses switched capacitor operation to first multiply the input signal, $\mathbf{x}$ with the measurement matrix $\mathbf{\Phi}$ using a C-2C ladder, followed by integration. The compressed signal $\boldsymbol{y}$ is then digitized and transmitted. The design was composed of 64 parallel arrays of 6-bit C-2C MDAC/integrator and 10-bit C-2C SAR ADC units. It was implemented in 0.13 μm CMOS process and occupied an area of 3 x 3 mm^2. The power dissipation is 28 nW/row (1.8 μW maximum) from Vdd = 1.2V. The compression ratio is controlled by changing the number of rows, M. Consequently, higher compression ratios result in lower power dissipation. Compression ratios up to 16X for ECG signals are reported with SQNR > 52 dB between thresholded and reconstructed signals.

2.4 Compressed Sensing System Performance Metrics

Compressed sensing is a lossy compression scheme. Error in signal reconstruction will be present in several factors (sparsity control, reconstruction error, system noise, etc), many of which will be discussed further in later sections of this work. This presents the demand for a consistent performance metric. A common performance metric in lossy compression is mean squared error (MSE). MSE

provides the average power of error present. For example, the MSE between the CS reconstructed signal and the original signal is defined as:

$$\text{MSE}(\hat{x}, x) = \frac{1}{N}\sum_{i=1}^{N}[\hat{x}[i] - x[i]]^2 \tag{2.13}$$

The MSE measure is specified in Equation 2.13 in terms of the two signals being compared. This delineation is instilled in all performance metrics in this work to better define the source of error. It is useful to introduce another common metric, percentage root mean square difference (PRD), that measures the average error relative to the signal [19]. The PRD between the CS reconstructed signal and the original signal is defined as:

$$\text{PRD}(\hat{x}, x) = 100 \cdot \sqrt{\frac{\sum_{i=1}^{N}[\hat{x}[i] - x[i]]^2}{\sum_{i=1}^{N} x[i]^2}} \tag{2.14}$$

Finally, the PRD can be mapped to signal-to-noise-ratio (SNR) through the following equation:

$$\text{SNR}(\hat{x}, x) = 20\log_{10}\left(\frac{1}{\frac{1}{100}\cdot \text{PRD}(\hat{x}, x)}\right) \tag{2.15}$$

Equations 2.13, 2.14 and 2.15 are mostly employed in this work to make accurate comparison to previous works. Figure 2.7 shows the relationship between PRD and signal quality. PRD values of 0.1% and 1.0% were chosen because they represent SNR values of 60dB and 40dB, respectively. These are values close to those achieved in other compression works, lending to a fair comparison in compression ratios. A PRD value of 20% was chosen because the signal still shows the ECG diagnostic characteristics (P, Q, R, S and T waves). PRD will be used in this work to quantify accuracy of CS design considerations not closely related to hardware (Chapters 3 and 4). SNR is utilized to quantify CS accuracy in Chapters 5 and 6 to maintain consistency with hardware engineering practices.

The American National Standards Institute and the Association (ANSI) for the Advancement of Medical Instrumentation (AAMI) provide the industrial standards for cardiac monitoring devices [20] and ambulatory ECG devices [21]. These standards state the ambulatory ECG monitors shall have a "maximum deviation from the ideal of ± 20 percent or $\pm 100\mu$V, whichever is greater". Therefore, where relevant, this work will present the CS system's percent ANSI/AAMI compliance the ANSI/AAMI standards.

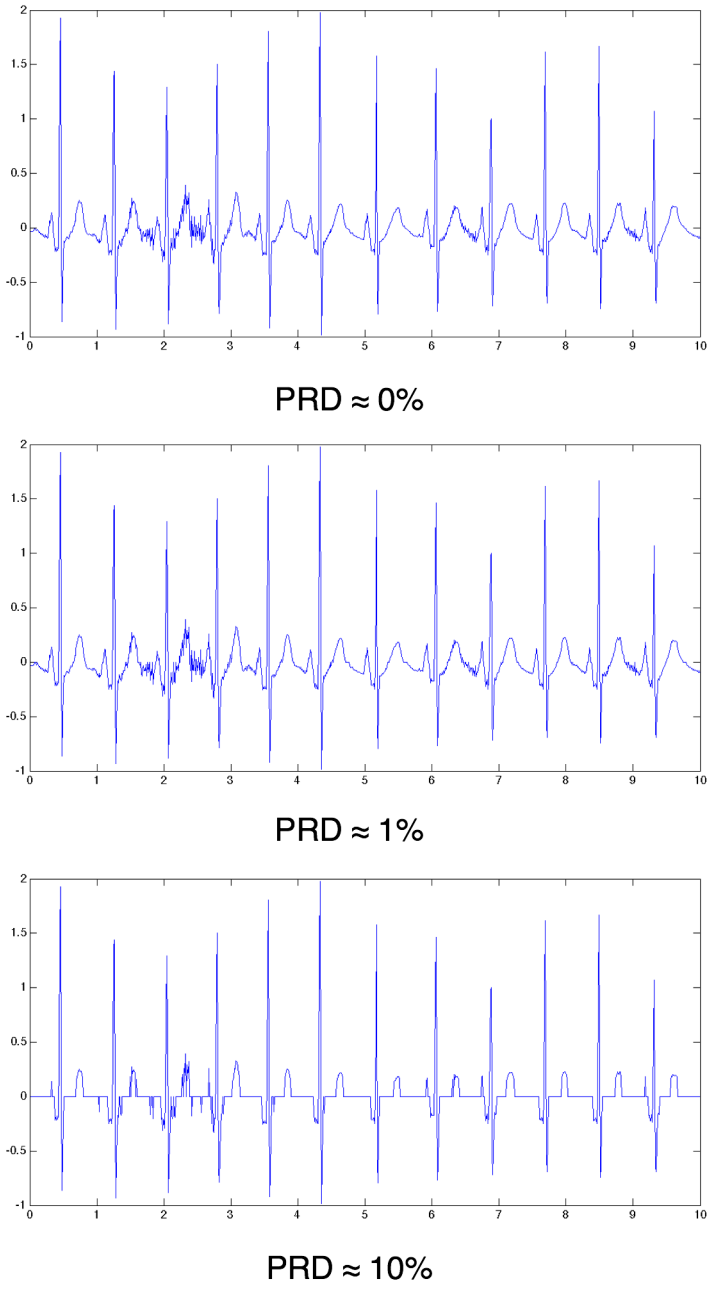


Figure 2.7: ECG Sample for Various PRD Values

Chapter 3

SPARSITY AND SAMPLING BASIS

The very first consideration in the compressed sensing system is the sampling domain. As discussed in Chapter 2, the number of compressed samples required for stable and accurate CS reconstruction is linearly related to the number of non-zero elements in the signal. In other words, a signal with higher sparsity can achieve a higher compression ratio.

Many bio-signals show sparsity in some sampling domain. First, this chapter emphasizes the importance of sparsity in CS in Section 3.1. Then, the advantages and disadvantages of different sampling domains for the ECG signal are explored. Most importantly, this work will address the trade-off between accuracy and sparsity. Since the sampling matrix is only used at reconstruction, no additional hardware is needed to implement different sampling bases. The sampling matrix does play a role in measurement matrix design (Chapter 4), and thus it's derivation is also discussed here. In particular, these topics will be addressed for the time and wavelet domains in Sections 3.2 and 3.3, respectively. The impact of thresholding and it's implications for the optimal sampling basis are addressed in Section 3.4. Section 3.5 will draw conclusions by reporting the best sampling domain section for ECG bio-signals and generalize these findings for applications in all CS system design decisions.

3.1 Impact of Sparsity

A requirement of compressed sensing is that the signal be sparse. In fact, the level of sparsity has a direct impact on the signal recovery accuracy, and thus compression ratio. Figure 3.1 helps demonstrate the importance of the signal sparsity. Time-domain ECG signals were thresholded to create varying levels of sparsity. The thresholded signals were then compressed with a $M \times N$ measurement matrix populated with independent and identically distributed (iid) Bernoulli values. Different compression ratios were achieved by changing M. Signals were recovered via convex optimization. Figure 3.1 shows the reconstruction error (PRD) between the thresholded signal and

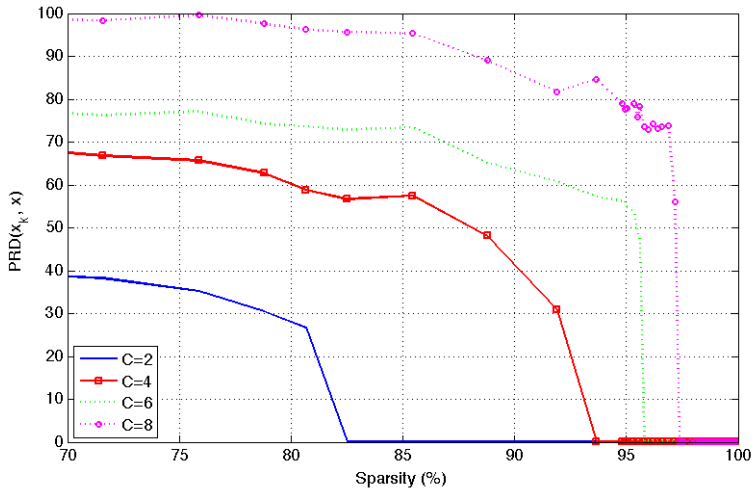


Figure 3.1: Sparsity vs PRD for CS Reconstructed Signal

the reconstructed signal with respect to sparsity. The different lines represent different compression ratios. It us clear that increased sparsity leads to less reconstruction error. Moreover, as the compression ratio increases, higher sparsity levels are required to achieve low reconstruction error. Perfect reconstruction (PRD=0%) for this set-up was achieved at 82.5% sparsity for C=2, 93.7% sparsity for C=4, 95.8% sparsity for C=6 and 97.4% sparsity for C=8.

Sparsity will play the largest role in CS success. As such, it is important to explore different sampling bases. In doing so, the goal is to find the most information dense sampling domain. An information dense sampling is one that expresses the signal most accurately with the fewest number of measurements.

3.2 Time-Domain Sampling Basis

The time domain is a straightforward sampling basis. Sampling the ECG signal in the time domain is achieved by taking evenly-distributed samples in time. The sampling basis matrix in the time domain is represented by the identity matrix.

To assess the performance of the time domain as a sampling basis, an experiment was conducted to measure sparsity. ECG data was collected from 20 different patients for one hour each. The raw

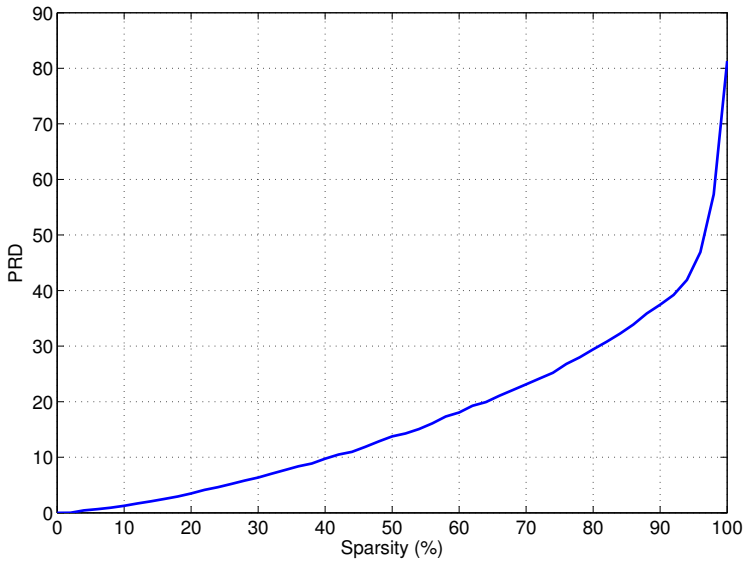


Figure 3.2: Sparsity vs PRD for Time-Domain ECG Signal

ECG time-based signal contains very few zero values. However, many of the samples have a very small amplitude and most the signal's energy can be captured in much fewer samples. For example, the QRS complex samples have a greater magnitude relative to the rest of the signal. It is important to determine the information density of the signal, giving rise to the need for quantification of the trade-off between accuracy and sparsity. The ECG data was divided into 1024-point frames (sampling frequency of 100 Hz) and thresholded using an absolute thresholding algorithm. The absolute thresholding algorithm simply compared each sample to a predetermined threshold value. If the amplitude was less than the threshold value, the value was set to zero. Otherwise, if the amplitude was greater than the threshold value, nothing changed.

Figure 3.2 presents the results of the ECG time-domain sparsity analysis. As anticipated, signal reconstruction error, PRD, increases with signal sparsity. This is the consequence of employing thresholding on a signal. An average PRD of 20% and 40% yields 64% and 92% sparsity, respectively.

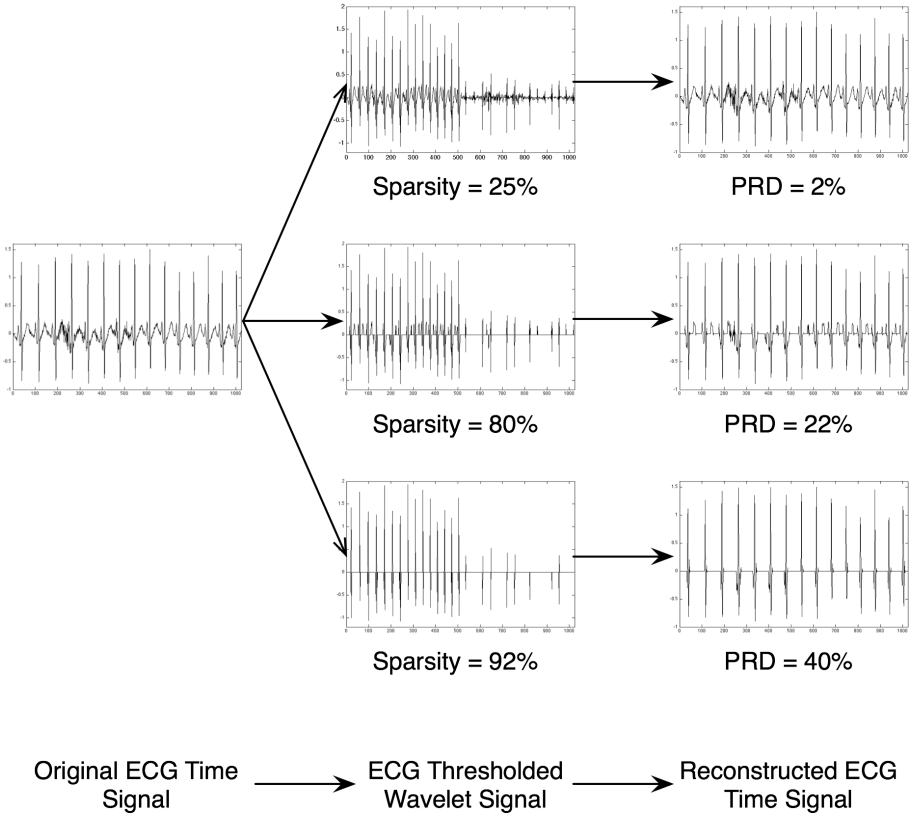


Figure 3.3: ECG Sparsity in the Wavelet Domain

3.3 Wavelet-Domain Sampling Basis

The wavelet domain is another potential sampling domain. The wavelet transform is the tool used to convert a signal into the wavelet domain and it is a common tool in ECG analysis. Applications of wavelet transform of ECG biosignals include ECG characteristic point detection for analysis automation [22], [23], denoising [24], and data compression [25].

The objective of exploring further sampling domains, such as wavelets, is to maximize sparsity of the signal. In doing so, the CS system will achieve a greater compression ratio. However, just as with the time domain, there is a trade-off between signal sparsity and accuracy. Figure 3.3 gives an example of the wavelet transform performed on an ECG signal. The ECG signal is transformed into "wavelet coefficients" by means of a wavelet transform. The signal is thresholded to maximize

sparsity. The thresholded signal is transformed back to an approximation of the original signal by means of an inverse wavelet transform. By enforcing sparsity constraints, signal content is permanently lost. Figure 3.3 demonstrates the effect of increasing signal sparsity in the wavelet domain on the ECG signal quality. For a very low sparsity level of 25%, the reconstructed time domain ECG signal shows no perceptual loss. For a higher sparsity level of 80%, the reconstructed signal still accurately shows many of the ECG characteristics (P wave, QRS complex, and some T waves). For a signal reconstructed from very high sparsity wavelet signals, only some of the ECG characteristics are still present. In this example, an ECG signal reconstructed from a 92% sparse ECG signal in the wavelet domain still contains details of the QRS complex but lacks information pertaining to the patient's P and T waves.

3.3.1 Discrete Wavelet Transform Definition

In order to intuitively understand wavelet analysis, it is beneficial to review common signal analysis techniques that led to it's existence. Figure 3.4 illustrates various sampling domain views of a signal. The most straightforward sampling domain is the time domain. In the time domain, the signal's amplitude is measured in evenly-distributed time intervals. The next sampling domain presented in Figure 3.4 is the frequency domain, or the Fourier analysis. The Fourier analysis measures the signal's frequency content over the entire time frame. Unfortunately, a couple factors make Fourier analysis unsuitable for ECG compression. First, time resolution does not really exist in Fourier analysis. It is not possible to recover exact time locations of certain events: a key requirement for any bio-signal acquisition system. Second, the Fourier analysis is not well suited for detecting less frequent events. This is a critical drawback for ECG bio-signal acquisition systems whose primary goal is to detect patient physiological anomalies. These drawbacks in the Fourier analysis led to the Short-Time Fourier Transform (STFT). The STFT measures the signal's frequency content in evenly distributed time windows. In doing so, the STFT copes with the time-resolution shortcomings of Fourier analysis. Unfortunately, time and frequency resolution can not be arbitrarily small. The time resolution - frequency resolution product is lower bounded. This concept is referred to as the Heisenberg inequality. It means that an increase in time resolution lends to a decrease in frequency resolution, and vice versa. Finally, wavelet analysis was introduced to surmount the aforemen-

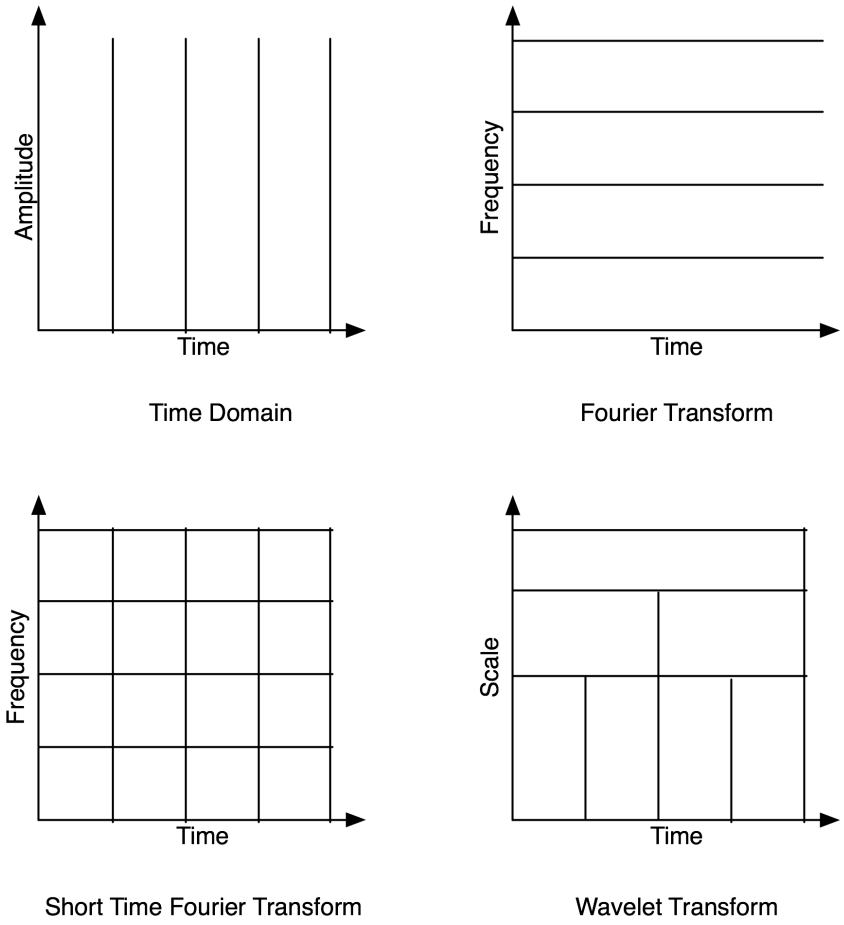


Figure 3.4: Different Sampling Views of a Time-Based Signal

tioned STFT resolution shortcomings. In wavelet analysis, the correlation between different scaled versions of a wavelet at varying time frames of the signal are calculated. The ability to vary the window size allows for better time-resolution and the ability to accurately capture less frequent signal anomalies.

The wavelet transform is a linear operation that decomposes a signal into multiresolution components. The oscillatory function $\psi(t)$ is considered a wavelet if it obeys the following condition:

$$\int_{-\infty}^{+\infty} \psi(t)\,\mathrm{d}t = 0 \tag{3.1}$$

This simply means the area of $\psi(t)$ is zero. This admissibility condition ensures a stable and invert-

ible wavelet transform. $\psi(t)$ is called the "mother wavelet", for which there are infinite possibilities.

A family of functions is derived from the mother wavelet by means of dyadic translations (shifting) and dilations (stretching):

$$\psi_{m,n}(t) = 2^{-m/2}\psi(2^{-m}t - n) \tag{3.2}$$

The family of functions form a new basis for the signal, $x(t)$, by performing a wavelet transform. The wavelet transform decomposes the signal into a set of wavelet coefficients, $c_{m,n}$, where (m,n) are the translation and dilation indices. Each wavelet coefficient is calculated by finding the inner product of the wavelet function at index (m,n), $\psi_{m,n}$, and the signal:

$$c_{m,n} = \langle x(t), \psi_{m,n}(t) \rangle = \int_{-\infty}^{+\infty} \psi_{m,n}(t)\alpha(t)\,\mathrm{d}t \tag{3.3}$$

The inverse wavelet transform recovers the original signal:

$$x(t) = \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} c_{m,n}\psi_{m,n}(t) \tag{3.4}$$

The discrete wavelet transform (DWT) is the wavelet transform of a signal at different frequency levels. It serves as the common tool to convert the time-domain signal into a wavelet-domain respresentation. The DWT utilizes a second key function, complimentary to the wavelet function, called the scaling function. Figure 3.5 shows a few common wavelet/scaling function pairs for ECG analysis. Each level of the DWT produces a set of low-frequency components, called "approximation coefficients", and high-frequency components, called "detail coefficients". The Fourier transform of scaling functions reveals a low-pass filter curve. The scaling function will produce the low-frequency approximation coefficients. The Fourier transform of wavelet functions reveals a band-pass filter curve. A combination of the scaling and wavelet functions will produce the high-frequency detail coefficients.

Let $\phi(t)$ be a scaling function. Dyadic translations (shifting) and dilations (stretching) form the scaling function set

$$\phi_{m,n}(t) = 2^{-m/2}\phi(2^{-m}t - n) \tag{3.5}$$

Because the scaling function represents a low-pass filter, the inner-product of the scaling function set and the signal produces a set of discrete approximation coefficients. At level L of the DWT, the

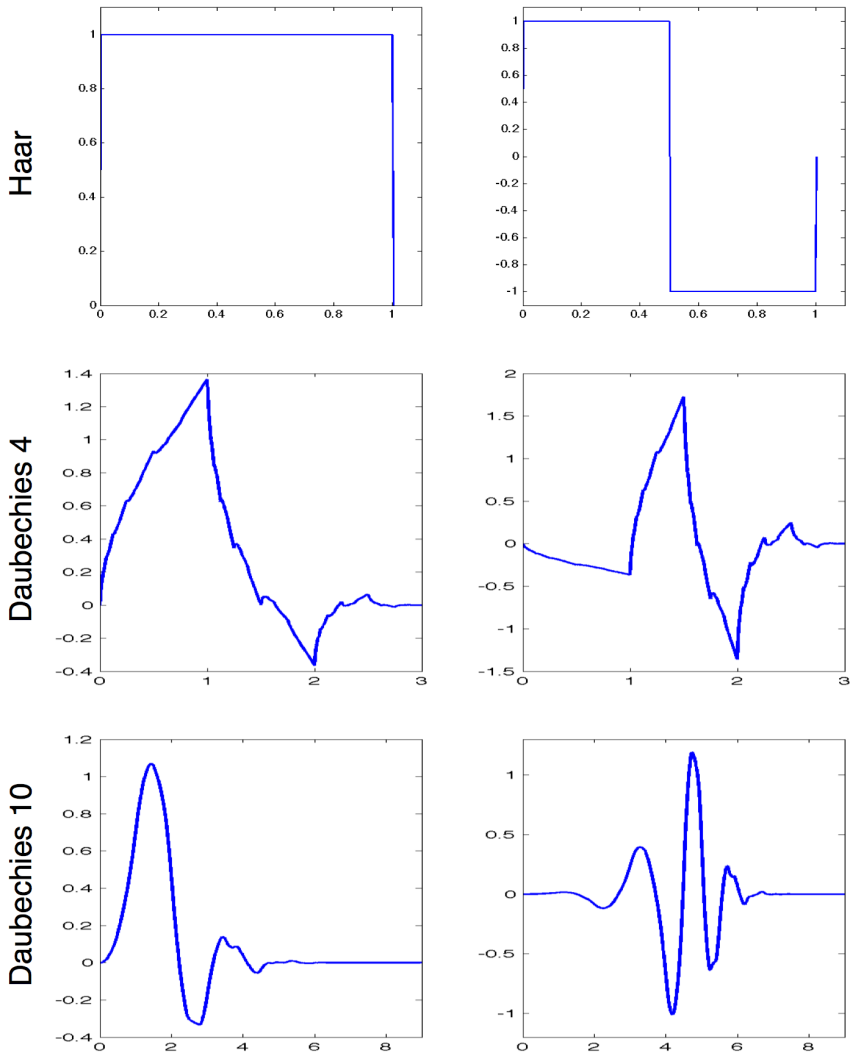


Figure 3.5: ECG Sparsity in the Wavelet Domain

discrete approximation coefficients are defined as

$$a_{L,n} = \langle x(t), \phi_{L,n}(t) \rangle = \int_{-\infty}^{+\infty} \phi_{L,n}(t) x(t) \, \mathrm{d}t \tag{3.6}$$

The discrete detail coefficients are calculated by finding the inner product of the wavelet function at index (L,n), $\psi_{L,n}$, and the signal:

$$d_{L,n} = \langle x(t), \psi_{L,n}(t) \rangle = \int_{-\infty}^{+\infty} \psi_{L,n}(t) x(t) \, \mathrm{d}t \tag{3.7}$$

Now, the multiresolution approximation of the signal is given as a function of both the scaling and wavelet functions

$$x(t) = \sum_{n=-\infty}^{+\infty} a_{L,n} \phi_{L,n}(t) + \sum_{m=-1}^{L} \sum_{n=-\infty}^{+\infty} d_{m,n} \psi_{m,n}(t) \tag{3.8}$$

The DWT system can be implemented as a filter bank [26]. At the first level of the system, the input signal is simultaneously passed through a symmetric low-pass and high-pass filter, and downsampled by 2. The low-pass filter produces the low-frequency approximation coefficients and the high-pass filter produces the high-frequency detail coefficients for that level. At each successive level, the low-pass filter and high-pass filter again produce that level's approximation and detail coefficients, respectively, but with the previous level approximation coefficients as the input. Let g(n) be the impulse response of the low-pass filter. The low-pass filter is related to the scaling functions through the following identities

$$g(n) = \langle \phi_{0,0}(t), \phi_{-1,n}(t) \rangle \tag{3.9}$$

$$\phi(t) = \sqrt{2} \sum_{n=-\infty}^{+\infty} g(n) \phi(2t - n) \tag{3.10}$$

Let h(n) be the impulse response of DWT system's high-pass filter. The high-pass filter is related to the scaling and wavelet functions through the following identities

$$h(n) = \langle \psi_{0,0}(t), \phi_{-1,n}(t) \rangle \tag{3.11}$$

$$\psi(t) = \sqrt{2} \sum_{n=-\infty}^{+\infty} h(n) \phi(2t - n) \tag{3.12}$$

The inverse discrete wavelet transform (IDWT) is performed by a reconstruction filter bank. The IDWT filter bank's filters are composed of a pair of low-pass and high-pass filters at each level that

compliment those of the DWT filter. Let $g'(n)$ be the impulse response of the reconstruction low-pass filter and let $h'(n)$ be the impulse response of the reconstruction high-pass filter. Their design is impacted by the wavelet family utilized. The IDWT filters for the Daubechies wavelets observed here obey the following identities

$$g'(n) = g(-n) \tag{3.13}$$

$$h'(n) = h(-n) \tag{3.14}$$

The impulse response of the Daubechies IDWT filters are simply the time reversal of their corresponding DWT filters.

3.3.2 Wavelet-Domain Sparsity vs. Accuracy

It is clear that further analysis is necessary to determine the most wavelet sampling domain effectiveness. An experiment was conducted to compare the sparsity and accuracy trade-offs for the DWT of the ECG signal. ECG data was collected from 20 different patients for one hour each. The ECG signals were decomposed into 1024-point samples (sampling frequency of 100 Hz) and the discrete wavelet transform was performed. The resulting ECG wavelet-domain signal was thresholded using an absolute thresholding algorithm. The inverse wavelet transform was then performed on the wavelet coefficients to determine the resulting reconstruction accuracy. The experiment was repeated for three proven effective wavelets for ECG signals: Haar, Daubechies 4, Daubechies 10.

Figure 3.7 presents the results of the ECG discrete wavelet transform sparsity analysis. The wavelet transform in this experiment was a one-level DWT only (i.e. only one pair of complimentary low-pass and high-pass filters). The results presented for the time domain and each wavelet are the average PRD (20 hours of ECG data spanning 20 different patients) for each level of sparsity. As previously discussed, higher signal sparsity leads to greater error in the signal reconstruction. Interestingly, the wavelet-domain ECG signal shows greater sparsity at a given signal accuracy than the time-domain ECG signal. Furthermore, analysis shows there is little difference in performance among the different discrete wavelet transform types. For example, an average PRD of 20% yields 64.1% sparsity in the time domain, 77.9% sparsity for Haar wavelet, 78.4% for Daubechies 4 wavelet and 79.1% sparsity for the Daubechies 10 wavelet. Another notable observation is the convergence of all the sampling domains at the higher sparsity/error levels. An average PRD of 40%

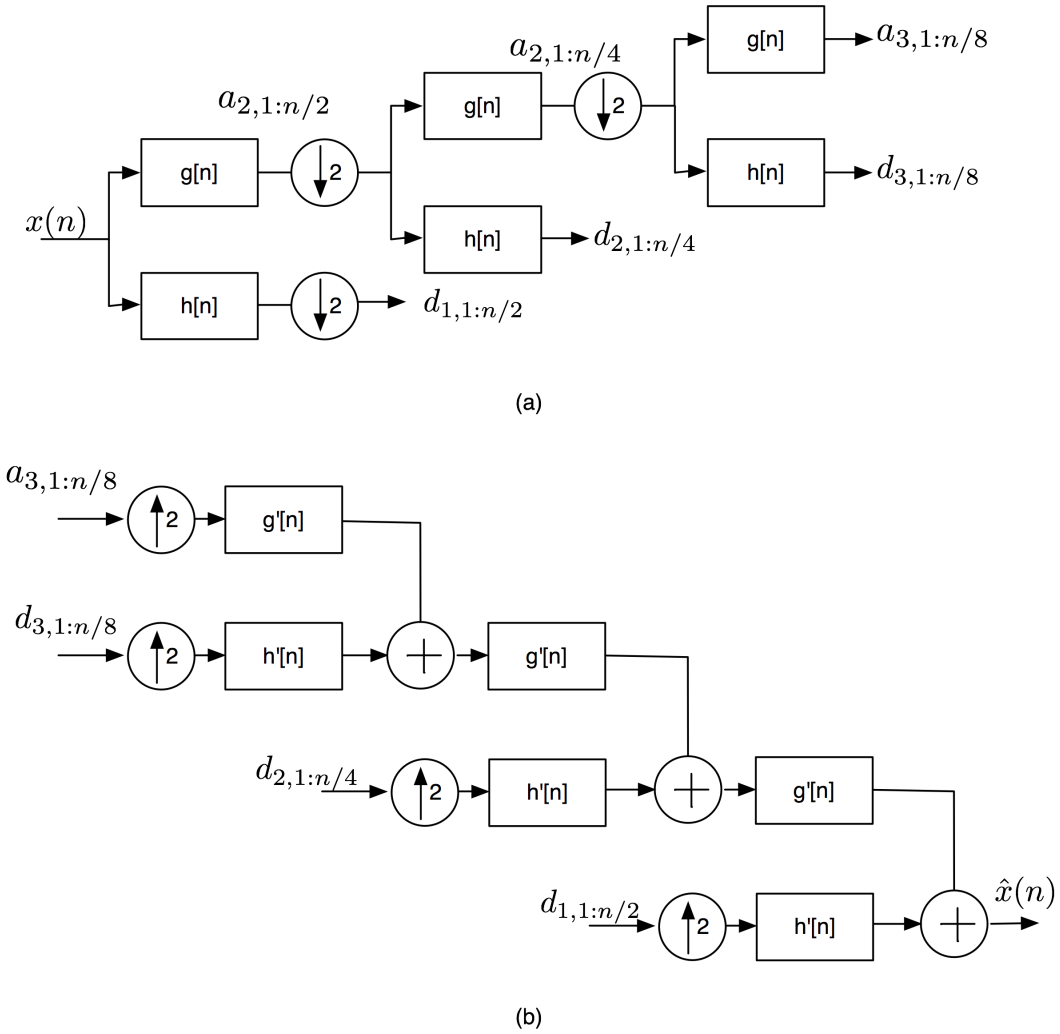


Figure 3.6: Filter Bank for (a) DWT and (b) IDWT

yields 92.6% sparsity in the time domain, 94.2% sparsity for Haar wavelet, 94.6% for Daubechies 4 wavelet and 94.3% sparsity for the Daubechies 10 wavelet.

It is intuitive that increasing the number of levels in the filter bank improves resolution. Thus, it is important to determine the extent of the increased signal reconstruction accuracy to select the appropriate filter bank depth. Further analysis was conducted to determine the impact of increasing the level of the DWT. The experiment was identical to that of Figure 3.7, but the filter bank depth

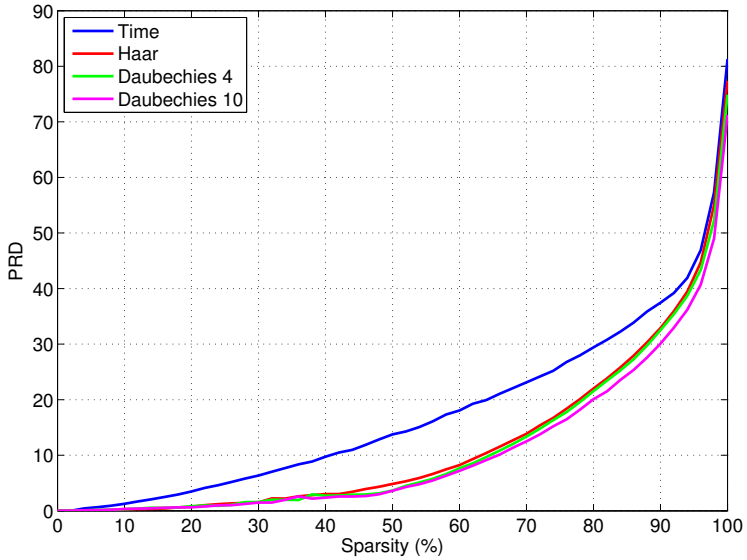
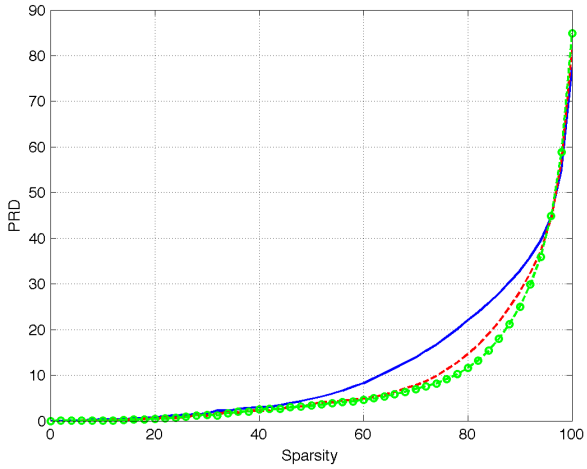


Figure 3.7: Sparsity vs PRD for Time and Wavelet Domains of ECG Signals

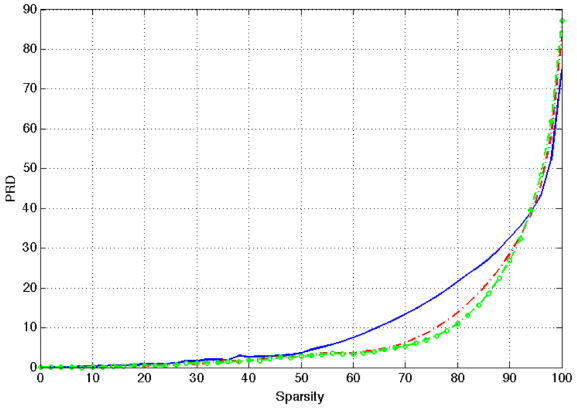
was varied. Figure 3.8 shows that for all wavelets tested here, signal accuracy is increased from a level-1 filter bank to a level-2 filter bank. Tables 3.1 and 3.2 summarize the performance of the different DWT system filter-bank levels at benchmarks PRD=20% and PRD=40%. For high accuracy systems (PRD=20%), Table 3.1 shows that a level-2 DWT for any wavelet shows higher sparsity for ECG signals. There is no benefit to level-3 or higher systems. For lower accuracy systems (PRD=40%), Table 3.2 shows there is no benefit for a higher level DWT filter bank.

Wavelet	Level-1	Level-2	Level-3
Haar	77.9%	84.7%	87.2%
Daubechies 4	78.4%	85.0%	86.7%
Daubechies 10	79.1%	82.8%	82.9%

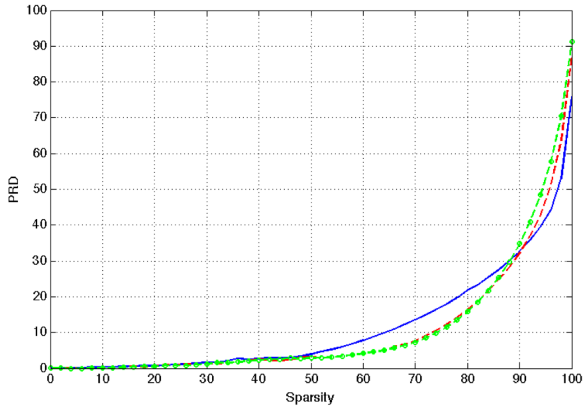
Table 3.1: Sparsity of Varying DWT System Depths for PRD=20%



Haar



Daubechies 4



Daubechies 10

Figure 3.8: Sparsity vs PRD for Varying DWT System Depths

Wavelet	Level-1	Level-2	Level-3
Haar	94.2%	94.8%	94.9%
Daubechies 4	94.6%	94.5%	94.1%
Daubechies 10	94.3%	93.0%	91.7%

Table 3.2: Sparsity of Varying DWT System Depths for PRD=40%

3.3.3 Wavelet-Domain Sampling Matrix Derivation

As noted in Section 3.3.2, the CS system performance for ECG signals is superior in the wavelet-domain. Therefore, it is necessary to derive the sampling matrix for the wavelet transform. Recall from Equation 2.4 that the N-length time-domain signal, x, is simply a product of a $N \times N$ sampling matrix, Ψ, and the N-length sparse signal of the desired sampling domain, α. The wavelet sampling matrix is derived from the wavelet and scaling functions. There wavelet transform matrix also depends on the input signal's length, N. The wavelet transform matrix will be derived for the Haar wavelet for a length-8 signal ($N = 8$). The example can be extended to develop any sampling wavelet matrix.

An example of wavelet transform matrix formulation will be developed for the Haar wavelet. For ease of explanation, the forward wavelet transform matrix is shown. The sampling matrix, $\mathbf{\Psi}$ is actually the inverse wavelet transform matrix. For the wavelets presented in this work (Haar and Daubechies family wavelets), the inverse wavelet matrix is simply the transpose of the wavelet matrix. The Haar wavelet and scaling functions are given by the following piecewise functions

$$\psi(t) = \begin{cases} 1 & 0 \leq t < 1/2 \\ -1 & 1/2 \leq t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.15}$$

$$\phi(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.16}$$

Combining Equations 3.9 and 3.16, the Haar wavelet low-pass filter impulse response is calculated as

$$g[n] = \begin{cases} \frac{1}{\sqrt{2}} & n = -1, 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.17}$$

Similarly, combining Equations 3.11 and 3.15, the Haar wavelet high-pass filter impulse response is calculated as

$$h[n] = \begin{cases} \frac{1}{\sqrt{2}} & n = 0 \\ -\frac{1}{\sqrt{2}} & n = -1 \\ 0 & \text{otherwise} \end{cases} \tag{3.18}$$

As described in Section 3.3.1, the approximation and detail coefficients are the convolution of the input signal and the low-pass and high-pass filters, receptively, followed by a downsampling of 2. Let $x[k]$ be the input time-domin signal. The level-1 approximation and detail coefficients are given by

$$a_{1,n} = \sum_{k=-\infty}^{\infty} x[k]g[2n-k] \tag{3.19}$$

$$d_{1,n} = \sum_{k=-\infty}^{\infty} x[k]h[2n-k] \tag{3.20}$$

Now, let $x[0:7] = [x_0 \quad x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7]^T$ be a 8-point time-domain input. The level-1 approximation and detail coefficients can be calculated from the wavelet's low-pass and high-pass filter impulse responses as

$$a_{1,n} = \begin{bmatrix} g[0] & g[-1] & g[-2] & g[-3] & g[-4] & g[-5] & g[-6] & g[-7] \\ g[-6] & g[-7] & g[0] & g[-1] & g[-2] & g[-3] & g[-4] & g[-5] \\ g[-4] & g[-5] & g[-6] & g[-7] & g[0] & g[-1] & g[-2] & g[-3] \\ g[-2] & g[-3] & g[-4] & g[-5] & g[-6] & g[-7] & g[0] & g[-1] \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} \tag{3.21}$$

$$d_{1,n} = \begin{bmatrix} h[0] & h[-1] & h[-2] & h[-3] & h[-4] & h[-5] & h[-6] & h[-7] \\ h[-6] & h[-7] & h[0] & h[-1] & h[-2] & h[-3] & h[-4] & h[-5] \\ h[-4] & h[-5] & h[-6] & h[-7] & h[0] & h[-1] & h[-2] & h[-3] \\ h[-2] & h[-3] & h[-4] & h[-5] & h[-6] & h[-7] & h[0] & h[-1] \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} \tag{3.22}$$

Let the matrices containing the low-pass and high-pass filter coefficients be $\mathbf{G_{l,N}}$ and $\mathbf{H_{l,N}}$, respectively, where l is that filter's level in the filter bank and N is the number of points in the input signal.

$$\mathbf{G_{1,8}} = \begin{bmatrix} g[0] & g[-1] & g[-2] & g[-3] & g[-4] & g[-5] & g[-6] & g[-7] \\ g[-6] & g[-7] & g[0] & g[-1] & g[-2] & g[-3] & g[-4] & g[-5] \\ g[-4] & g[-5] & g[-6] & g[-7] & g[0] & g[-1] & g[-2] & g[-3] \\ g[-2] & g[-3] & g[-4] & g[-5] & g[-6] & g[-7] & g[0] & g[-1] \end{bmatrix} \tag{3.23}$$

$$\mathbf{H_{1,8}} = \begin{bmatrix} h[0] & h[-1] & h[-2] & h[-3] & h[-4] & h[-5] & h[-6] & h[-7] \\ h[-6] & h[-7] & h[0] & h[-1] & h[-2] & h[-3] & h[-4] & h[-5] \\ h[-4] & h[-5] & h[-6] & h[-7] & h[0] & h[-1] & h[-2] & h[-3] \\ h[-2] & h[-3] & h[-4] & h[-5] & h[-6] & h[-7] & h[0] & h[-1] \end{bmatrix} \tag{3.24}$$

The output for a level-1 DWT is defined as a matrix containing both the level-1 approximate and detail coefficients, $\alpha = \begin{bmatrix} a_{1,n} \\ d_{1,n} \end{bmatrix}$. It is deduced from Equations 3.21 and 3.22 that the wavelet coefficients are simply linear combinations of the input signal. The level-1 wavelet transform matrix, $\mathbf{W_{1,8}}$, for an 8-point input signal can be calculated as

$$\mathbf{W_{1,8}} = \begin{bmatrix} \mathbf{G_{1,8}} \\ \mathbf{H_{1,8}} \end{bmatrix} \tag{3.25}$$

For the example of the Haar wavelet, Equation 3.25 yields the level-1 Haar wavelet transform matrix for an 8-point input signal as

$$\mathbf{W_{1,8}} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \tag{3.26}$$

Since the sampling matrix $\mathbf{\Psi}$ is the inverse wavelet matrix and the inverse wavelet matrix is simply the transpose of the wavelet matrix, the level-1 Haar sampling matric for an 8-point input signal is

$$\mathbf{\Psi} = \mathbf{W_{1,8}}' = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{3.27}$$

Next, the wavelet sampling matrix is developed for a multi-level DWT system (Figure 3.6). Consider a level-2 DWT filter bank (L=2). The detail coefficients derived for the level-1 DWT system are still an output. However, the approximation coefficients, $a_{1,n}$, must be further decomposed to yield the level-2 approximation coefficients, $a_{2,n}$, and detail coefficients, $d_{2,n}$. The filtering and subsampling operations have not changed, but they are now operated on a signal that is half the length ($\frac{N}{2}$). The level-2 coefficients are calculated as

$$a_{2,n} = \begin{bmatrix} g[0] & g[-1] & g[-2] & g[-3] \\ g[-6] & g[-7] & g[0] & g[-1] \end{bmatrix} \begin{bmatrix} a_{1,1} \\ a_{1,2} \\ a_{1,3} \\ a_{1,4} \end{bmatrix} \tag{3.28}$$

$$d_{2,n} = \begin{bmatrix} h[0] & h[-1] & h[-2] & h[-3] \\ h[-6] & h[-7] & h[0] & h[-1] \end{bmatrix} \begin{bmatrix} a_{1,1} \\ a_{1,2} \\ a_{1,3} \\ a_{1,4} \end{bmatrix} \tag{3.29}$$

$\mathbf{G_{2,4}}$ and $\mathbf{H_{2,4}}$ define the matrices containing the level-2 filter coefficients and $\mathbf{I_{4x4}}$ and $\mathbf{0_{2x4}}$ are 4x4 identity and 2x4 zero matrices, respectively. If the output for a level-2 DWT is defined as a matrix containing the level-1 detail, level-2 approximate and level-2 detail coefficients $x = \begin{bmatrix} a_{2,n} \\ d_{2,n} \\ d_{1,n} \end{bmatrix}$, then the level-2 Haar wavelet transform matrix, $\mathbf{W_{2,8}}$, for an 8-point input signal can be calculated as

$$\mathbf{W_{2,8}} = \begin{bmatrix} \mathbf{G_{2,4}} & \mathbf{0_{2x4}} \\ \mathbf{H_{2,4}} & \mathbf{0_{2x4}} \\ \mathbf{0_{4x4}} & \mathbf{I_{4x4}} \end{bmatrix} \begin{bmatrix} \mathbf{G_{1,8}} \\ \mathbf{H_{1,8}} \end{bmatrix} \tag{3.30}$$

Returning again to the Haar wavelet example, Equation 3.30 yields the level-2 Haar wavelet transform matrix of an 8-point input signal as

$$\mathbf{W_{2,8}} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \tag{3.31}$$

Finally, this yields a level-2 Haar wavelet sampling matrix for an 8-point input signal as

$$\mathbf{\Psi} = \mathbf{W_{2,8}}' = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 11 \end{bmatrix} \tag{3.32}$$

The wavelet sampling matrix is dependent on the type of wavelet function and the length of the input signal. An example wavelet sampling matrix was derived for a level-1 and level-2 Haar wavelet and an 8-point input signal. The principles can be extended to change the depth of the filter bank or formulate the Daubechies 4 and Daubechies 10 wavelet matrices of any size.

3.4 Thresholding and Sparsity

In Chapter 5, this dissertation will present two different types of CS systems: one with aggressive thresholding and one with no or very little thresholding. The system with no or very little thresholding perform compressed sensing for very low compression ratios. This system treats the lack of high sparsity levels as compression noise. As such, it is important that the decoder solves for a sparse signal, $\boldsymbol{\alpha}$, in a sampling domain that is information dense (better accuracy with fewer number of points). In this way, the noise perceived by the decoder is less detrimental to the signal recovery accuracy. Research conducted in Sections 3.2 and 3.3 show that a thresholded $\boldsymbol{\alpha_k}$ in the wavelet sampling domain produced a more accurate representation of the time-domain ECG signal, $\mathbf{x}$, than $\boldsymbol{\alpha_k}$ in the time domain. This dissertation will later show that this information is crucial to the success of the CS system with no or very little thresholding.

The other type of CS system, that with aggressive thresholding, targets much higher compression ratios. In order to keep encoder hardware complexity to a minimum, the thresholding must occur in the time domain. This will eliminate the need for extra hardware to transform the input time domain signal into the desired sampling domain [27], [28], [29]. However, thresholding $\mathbf{x}$ presents

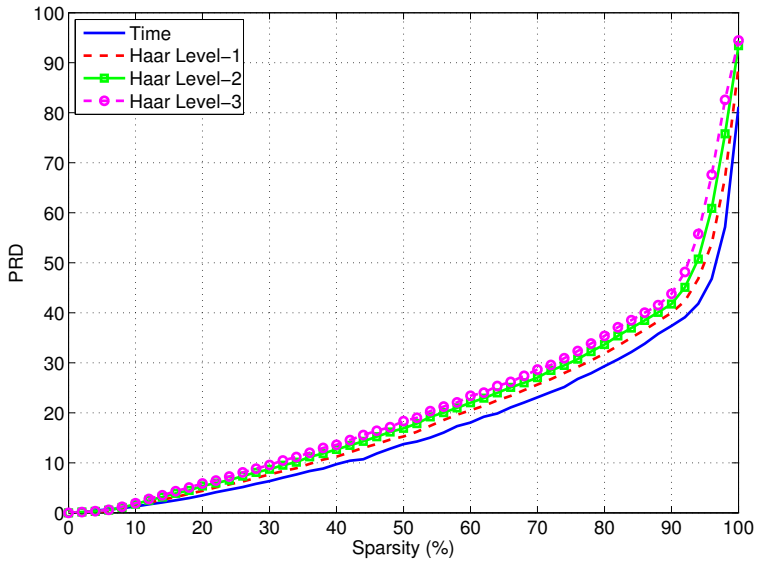


Figure 3.9: Sparsity vs PRD for Time and Wavelet Domains of Time-Thresholded ECG Signals

different sparsity results.

Figure 3.9 shows the average accuracy of an ECG signal for a given sparsity level in the time and Haar wavelet domains after thresholding in the time domain. ECG data was collected from 20 different patients for one hour each. The ECG signals were decomposed into 1024-point samples (sampling frequency of 100 Hz) and thresholded in the time domain using an absolute thresholding algorithm, just as would be done in the encoder designed here. The resulting signal is $\mathbf{x_k}$. The accuracy (PRD) of the signal is directly comparing $\mathbf{x_k}$ to $\mathbf{x}$ and the time domain's sparsity is calculated by simply summing the number of non-zero entries in $\mathbf{x_k}$. To determine the wavelet representation of $\mathbf{x_k}$'s sparsity, $\mathbf{x_k}$ was multiplied by the wavelet transform matrix to obtain $\boldsymbol{\alpha}$ and the number of non-zero entries were summed. The results show that, for a time domain thresholded signal, there is little difference in sparsity among wavelet and time domains. In fact, the Haar wavelet domain and increasing the DWT depth produces slightly worse results than the time domain. For example, an average PRD of 20% yields 64.2% sparsity in the time domain and 58.9%, 55.9% and 53.5% in the Haar wavelet domain for DWT depths of 1, 2 and 3, respectively.

3.5 Sampling Domain Conclusions

The key to a successful CS system is signal sparsity. High signal sparsity achieves a higher CS compression ratio, resulting in higher power savings. Many bio-signals show sparsity in some sampling domain and this work specifically explores the time and wavelet sampling bases of ECG signals.

This work first shows that without any thresholding of the ECG signal, the wavelet domain exhibits enough sparsity for successful CS operation. This is the first indication of the importance of sampling basis selection, as the time domain did not perform adequately for a system without thresholding. It is very important to note that the ECG CS system that does not employ thresholding will achieve very low compression ratios, regardless of the sampling basis. This work demonstrates a compression ratio of C=1.5 for the ECG signal without thresholding.

Next, the work shows the sparsity of the ECG signal when thresholded in the appropriate domain. The results of the study show that for medium accuracy systems (PRD=20%), ECG signals exhibit mush higher sparsity in the wavelet domain than the time domain. The study further concluded that there was negligible difference in sparsity for the type of wavelet under scrutiny in this study (Haar, Daubechies 4 and Daubechies 10). The level-2 and level-3 DWT showed superior results to the level-1 DWT. A method to calculate the sampling matrix to be incorporated in the CS decoder was presented. The results of the study also reveal that for low accuracy systems (PRD=40%), such as those used in ambulatory ECG monitoring, there is little difference in sparsity between the time and wavelet domains. To sum up, the wavelet domain works better when thresholding is conducted in the wavelet domain.

Finally, this work shows the impact of sampling basis selection when only time thresholding is considered. Time thresholding is the most practical approach for CS encoder hardware simplicity. When the ECG signal is thresholded in the time domain, the time sampling basis sparsity outperforms the wavelet sampling basis.

In general, CS performance is most affected by signal sparsity. It is imperative in any CS design to recreate the investigation shown here for the intended application. Various sampling domains should be tested to find which has the highest information density for the intended thresholding scheme. For the case of ECG signals achieving high compression ratios, the time sampling basis is

best because thresholding in the wavelet domain is expensive.

Chapter 4

THE SENSING MATRIX

The keystone to CS success is that the sensing matrix, Φ, be a random matrix. In doing so, Φ becomes nearly orthogonal. As discussed in Section 2.1.2, the incoherence between the sampling basis Ψ and the sensing basis Φ is a key criterion for CS success. CS literature states that a random Φ will, with high probability, be incoherent to any sampling basis. Candes and Wakin [6] promote the use of any random distribution function, but make a special note of uniform, normal and Bernoulli distributions. The dominant choice among existing CS architectures is the Bernoulli distribution, due to it's one bit implementation. While these works show functional CS solutions, there is nothing to indicate an optimal random distribution was selected. What's more is that these architectures use on-chip pseudorandom number generators without any indication that they will produce the desired random properties. This section strives to help guide sensing matrix design decisions. Section 4.1 will open with an investigation into the calculated incoherence of different random probability distributions with time and wavelet domain sampling bases. It will also confirm, via simulations performed on ECG signals, the relationship between incoherence and signal reconstruction quality and ultimately achievable compression. Section 4.2 discusses the use of on-chip pseudorandom number generators, including calculating their coherence and mutual coherence to determine if they are in fact a viable solution to CS encoder sensing matrix formulation.

4.1 Impact of Random Sensing Matrix Incoherence with Sampling Matrix

The key concept behind compressed sensing is that random linear projections significantly reduces the number of required measurements of a sparse signal. CS literature argues that incoherence between the sampling and sensing bases has a direct impact on CS performance. Core concepts and equations are repeated to better convey the work, but Section 2.1.2 provides a more detailed discussion of the CS incoherence criterion. Coherence, a measure of correlation between the rows of $\mathbf{\Phi}$ and the columns of $\mathbf{\Psi}$, is given as

Random Distribution Name	PDF/PMF
Bernoulli	$f(x) = \left\{ \begin{array}{l} \frac{1}{2} \text{ for } x = \frac{1}{\sqrt{m}} \\ \frac{1}{2} \text{ for } x = -\frac{1}{\sqrt{m}} \end{array} \right\}$
Sparse Bernoulli	$f(x) = \left\{ \begin{array}{l} \frac{1}{2} \text{ for } x = 0 \\ \frac{1}{2} \text{ for } x = 1 \end{array} \right\}$
Uniform	$f(x) = \left\{ \begin{array}{l} 1 \text{ for } x \in [0,1] \\ 0 \text{ otherwise} \end{array} \right\}$
Normal	$f(x) = \frac{\sqrt{m}}{\sqrt{2\pi}} e^{-\frac{m \cdot x^2}{2}}$
Exponential	$f(x) = e^{-x}$
LogN Low σ	$f(x) = \frac{1}{x\sqrt{2\pi}} e^{\frac{-(\text{Ln}(x)-2)^2}{2}}$
LogN High σ	$f(x) = \frac{4}{3x\sqrt{2\pi}} e^{\frac{-8(\text{Ln}(x)-2)^2}{9}}$

Table 4.1: Experimental $\mathbf{\Phi}$ Random Distributions

$$\mu(\Psi, \Phi) = \sqrt{N} \cdot \max_{0 \leq j,k < N} |\langle \psi_j, \phi_k \rangle| \tag{4.1}$$

where $\boldsymbol{\psi}$ is a column of $\mathbf{\Psi}$ and $\boldsymbol{\phi}$ is a row of $\mathbf{\Phi}$. It is assumed throughout this work that $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$ are unit vectors ($||\boldsymbol{\psi}||_2 = ||\boldsymbol{\phi}||_2 = 1$). Consequently, the range of coherence is given by $\mu \in [1, \sqrt{N}]$. Based on this definition of coherence, CS performance bounds are given as

$$M \geq C \cdot \mu^2(\mathbf{\Psi}, \mathbf{\Phi}) \cdot k \cdot log(N) \tag{4.2}$$

where C is a small constant.

An investigation was conducted to determine the coherence of several different random distributions with time and wavelet sampling matrices. The random distributions tested are given in Table 4.1. They are defined here by their probability density function (pdf) for the continuous random variable case and the probability mass function (pmf) for the discrete random variable case. The Bernoulli, uniform and normal distributions were included for their mention in [6]. Furthermore, the Bernoulli solution is popular among existing implementations [12], [17]. The exponential and log-normal random distributions were chosen to examine the effects of random distribution families

with a much wider spread. The "Sparse Bernoulli" function is considered due to ease of implementation. The Sparse Bernoulli distribution function is one with equal probability for entries 0/1. The "Exponential" function used here is an exponential random distribution with $\mu = 1$. The "LogN Low σ" function is a log-normal random distribution with $\mu = 2$ and $\sigma = 0.75$. The "LogN High σ" function is a log-normal random distribution with $\mu = 2$ and $\sigma = 1$.

4.1.1 Incoherence of Time vs. Wavelets Sampling Basis

First, it is important to calculate the coherence of different random distributions. Coherence is calculated using Equation 4.1. However, since we are dealing with random values, coherence varies for each $\mathbf{\Phi}$. An experiment was conducted to measure the coherence of several $\mathbf{\Phi}$ matrices with the various random distributions of Table 4.1. In the experiment, 1,000 $\mathbf{\Phi}$ matrices of each random distribution were formulated for each of the following dimensions: 128×1024, 170×1024 and 256×1024 (for compression ratios 8X, 6X and 4X, respectively). Bernoulli and Sparse Bernoulli are one-bit implementations while all other random distributions are double-precision floating point. In total, 3,000 $\mathbf{\Phi}$ matrices were formulated for each random distribution. The coherence between each $\mathbf{\Phi}$ and the time domain $\mathbf{\Psi}$ ($\mathbf{\Psi}$ is the identity matrix for a time-domain signal) was calculated. Since the range of coherence lies within $[1, \sqrt{n}]$, the range for coherence in this experiment was $[1, 32]$.

The results of the time domain coherence experiment are depicted by the histograms in Figure 4.1. The mean and standard deviation of the coherence values are also summarized in Table 4.2. Bernoulli exhibits the lowest coherence, with a constant value of $\mu = 1$. The Sparse Bernoulli and Uniform distributions also exhibited very low coherence values, with $\mu = 1.48 \pm 0.01$ and $\mu = 1.80 \pm 0.01$, respectively. The Normal, Exponential, LogN Low σ and LogN High σ all have increasing coherence value, with average values of 4.64, 8.70, 15.06 and 23.16, respectively. Furthermore, these random distributions also exhibit much greater variation, as evidenced by the higher standard deviations.

The coherence experiment was repeated for the wavelet domain. Section 3.3 proved that Haar, Daubechies 4 and Daubechies 10 wavelets yielded similar superior sparsity performance than the time domain for ECG signals. It is crucial to determine if the incoherence between the random dis-

	Time ($\mu \pm \sigma$)	Haar ($\mu \pm \sigma$)	Daubechies 4 ($\mu \pm \sigma$)
Bernoulli	1 ± 0	1.41 ± 0	1.67 ± 0
Sparse Bernoulli	1.48 ± 0.01	2.28 ± 0.03	2.28 ± 0.03
Uniform	1.80 ± 0.01	2.63 ± 0.03	2.63 ± 0.03
Normal	4.64 ± 0.27	4.62 ± 0.26	4.62 ± 0.26
Exponential	8.70 ± 0.83	7.73 ± 0.70	7.72 ± 0.71
LogN Low σ	15.06 ± 2.35	12.87 ± 2.00	12.87 ± 1.92
LogN High σ	23.16 ± 2.62	19.60 ± 2.22	19.59 ± 2.21

Table 4.2: Experimental $\mathbf{\Phi}$, $\mathbf{\Psi}$ Coherence Summary

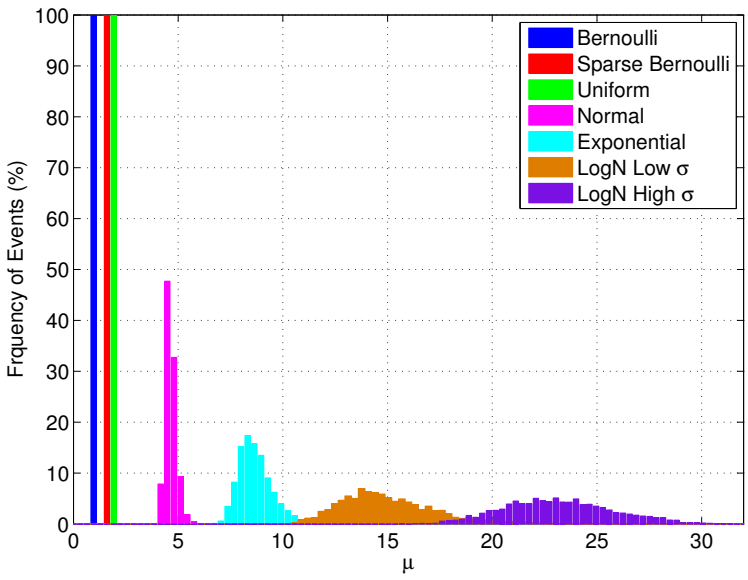
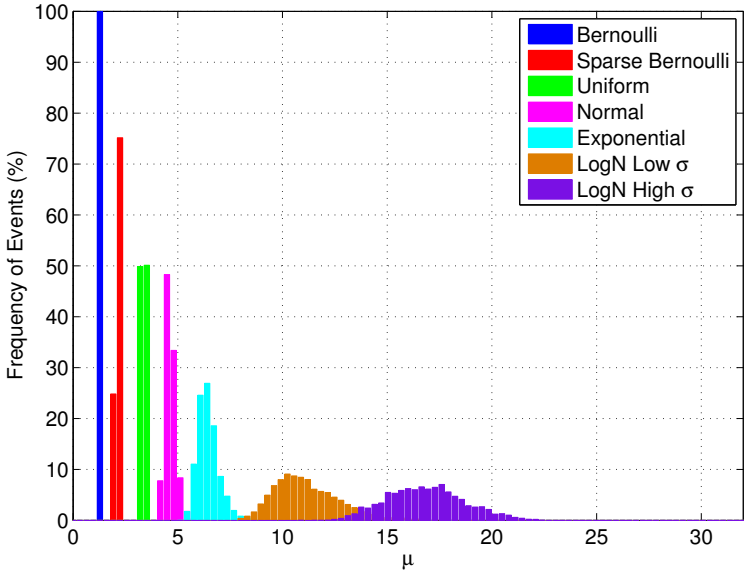
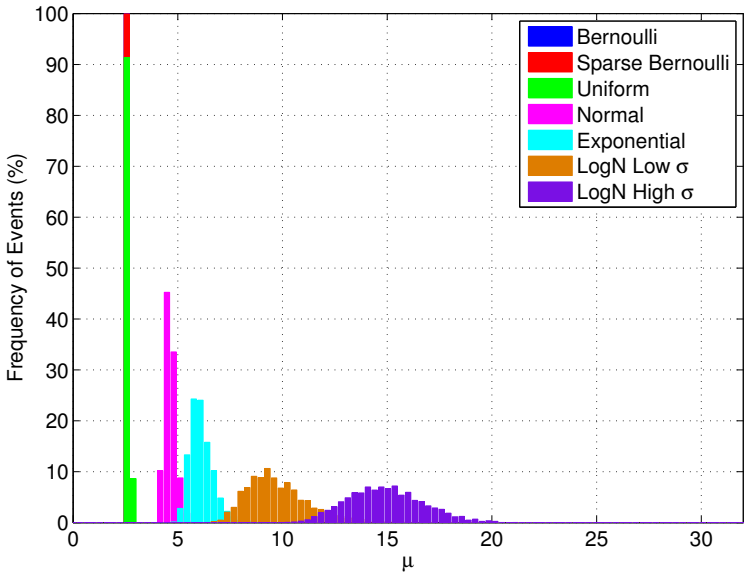


Figure 4.1: Time Domain Coherence Histograms

tributions and their sampling basis matrices is also high. The distribution of the wavelet coherences with the $\mathbf{\Phi}$ of Table 4.1 are presented by the histograms in Figure 4.2. The results are summarized



(a) Level-1 Haar Wavelets Coherence Histograms



(b) Level-1 Daubechies 4 Wavelets Coherence Histograms

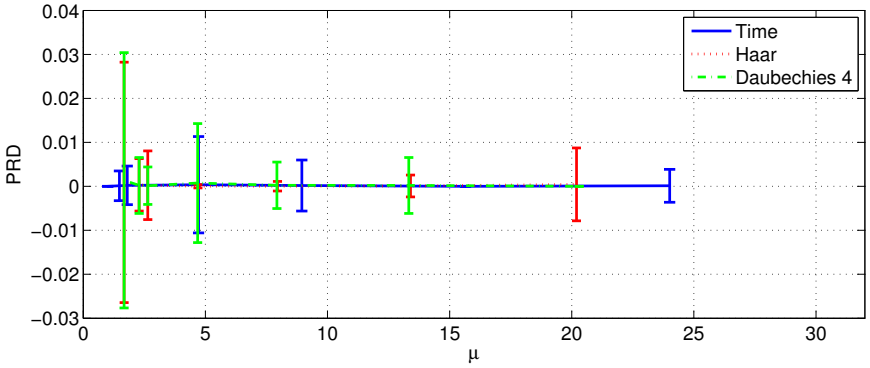
Figure 4.2: Wavelet Domain Coherence Histograms

in Table 4.2. It is interesting to note the similarity in trend between the different random probability distributions in both the time and wavelet domains. The order of increasing coherence for the different random distributions is the same for both sampling domains. For the Bernoulli, Sparse Bernoulli and Uniform random distributions, the wavelets had slightly higher coherence (both mean and standard deviation). The coherence for the Normal distribution is roughly equivalent for both time and wavelet domains. Interestingly, Exponential, LogN Low σ and LogN High σ are more incoherent in the wavelet bases. Albeit, their coherence is still significantly greater than the coherence demonstrated in the aforementioned random distributions.

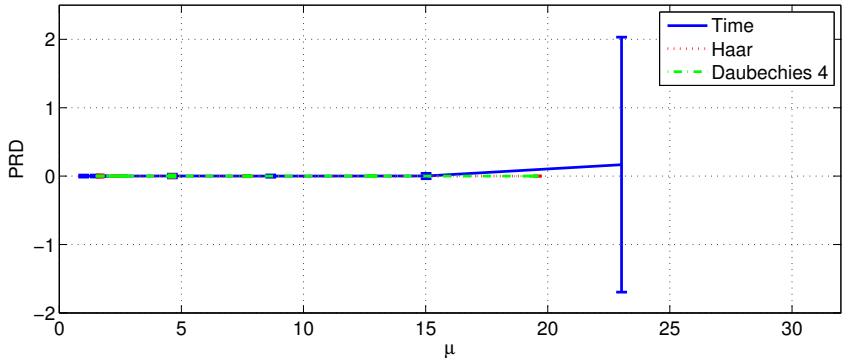
4.1.2 The Relationship Between Incoherence and Accuracy

The next logical progression is to confirm the relationship between coherence and compressibility. The experiment was run for the time, level-1 Haar wavelet and level-1 Daubechies 4 wavelet domains. For each domain the sparse signal, $\boldsymbol{\alpha}$, was thresholded to maintain 98% sparsity, $\boldsymbol{\alpha}_k$. The signal was compressed with each of the random distributions given in Table 4.1. Three different compression levels were tested: C=4X, 6X and 8X. Figure 4.3 shows signal reconstruction error, PRD, between the time-domain signal, $\boldsymbol{x}$, and the reconstructed signal, $\hat{\boldsymbol{x}}$. For C=4, the PRD between $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ is very low. The mean plus standard deviation for the time $\boldsymbol{\Psi}$ never exceeds 0.005. The mean plus standard deviation for the Haar and Daubechies 4 $\boldsymbol{\Psi}$ never exceeds 0.035. For C=6, error is still minimal for all sampling domains. Time $\boldsymbol{\Psi}$ mean plus stand deviation PRD for coherence less than 15.06 ± 2.35 is less than 0.05. Haar $\boldsymbol{\Psi}$ and Daubechies 4 $\boldsymbol{\Psi}$ mean plus stand deviation PRD for coherence less than 12.87 ± 2.00 never exceeds 0.02. For C=8, the PRD gets substantially larger. The trend is more clear for C=8, as coherence increases, so does reconstruction error. Time $\boldsymbol{\Psi}$ mean plus stand deviation PRD for coherence less than 4.64 ± 0.27 is less than 15, and only less than 55 for coherence up to 23.16 ± 2.62. Haar and Daubechies 4 $\boldsymbol{\Psi}$ mean plus stand deviation PRD for coherence ranging from 1.41 ± 0 to 19.60 ± 2.22 is less than 30.

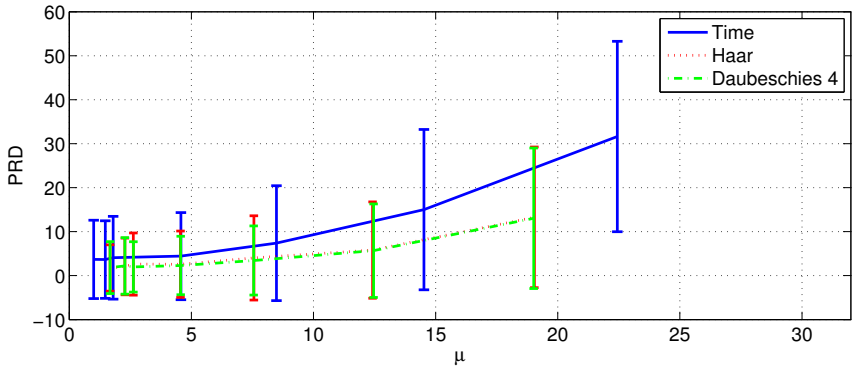
It is important to note that this experiment does confirm the existence of a relationship between coherence and accuracy. As coherence increases, reconstruction error increases. Interestingly enough, the most important finding of this research is that μ is only a proxy for performance. There is not a strict mapping between the coherence between a sensing matrix and any sampling matrix and



(a) Coherence vs. PRD for C=4



(b) Coherence vs. PRD for C=6



(c) Coherence vs. PRD for C=8

Figure 4.3: Coherence vs. PRD for Time and Wavelet Sampling Bases

accuracy. Figure 4.3(c) shows that the sampling basis also has an impact on performance. For example, a similar coherence in the wavelet domains achieves better accuracy than in the time domain. Referring to Equation 4.2, there is a small constant, C, that helps define the achievable compression. The experiment shows that the constant is related to the sampling basis. Within a sampling basis, however, lower coherence always yielded better results which is to be considered in all CS encoder design.

4.1.3 Sensing Matrix Resolution and One-Bit Sensing Matrix

Another key consideration is sensing matrix resolution. Previously in Section 4.1, experiments showed the error of CS reconstruction for sensing matrices constructed using various probability distribution functions. However, the resolution for most of the sensing matrix values were double-precision floating-point (the exception was, of course, the Bernoulli and Sparse Bernoulli distributions which were only one-bit). This is not practical for the analog CS design of [1] as it would both consume a large area and be power-hungry. Studies have been taken to determine the minimum acceptable resolution [30]. In [30] and [31], our group reported that a more reasonable 6 bits were required for Uniform and Gaussian distributions to achieve accurate CS reconstruction.

The implications of Figure 4.3 regarding resolution are intriguing. Figure 4.3 reveals that the one-bit sensing matrices, Bernoulli and Sparse Bernoulli random distributions, are a valid design choice for the sensing matrix. In fact the one-bit solutions performed better than most other double-precision floating-point solutions. The only exception was the uniform distribution which demonstrated very similar performance results. The significance of this discovery is the profound benefit on hardware. The one-bit sensing matrix solution will be both smaller in area (by decreasing the number of capacitors in the C-2C ladder) and consume less power. As such, for this work, it is the recommendation that one-bit CS encoders are the optimal design choice.

4.2 One-Bit Sensing Matrix Generation Hardware

The overarching goal of this work is to aid in better CS system design. Thus, it is important to apply these findings to hardware implementation concepts. Evidence presented in previous sections clearly indicated that, for at least the time and wavelet domains scrutinized for ECG applications,

one-bit random matrices often perform better than multi-bit solutions. Due to superior performance and the obvious savings in both area and power, only one-bit random matrices are discussed here. This section will discuss the advantages and disadvantages of a few different random matrix generation methods. The one-bit solutions presented here will pertain to a random distribution containing only +1/-1 entries, similar to the Bernoulli random distribution. This distribution is explored because of it's use to implement a differential input CS analog encoder. However, the analysis presented in this section can be easily applied to a random distribution containing on +1/0 entries, similar to the Sparse Bernoulli distribution.

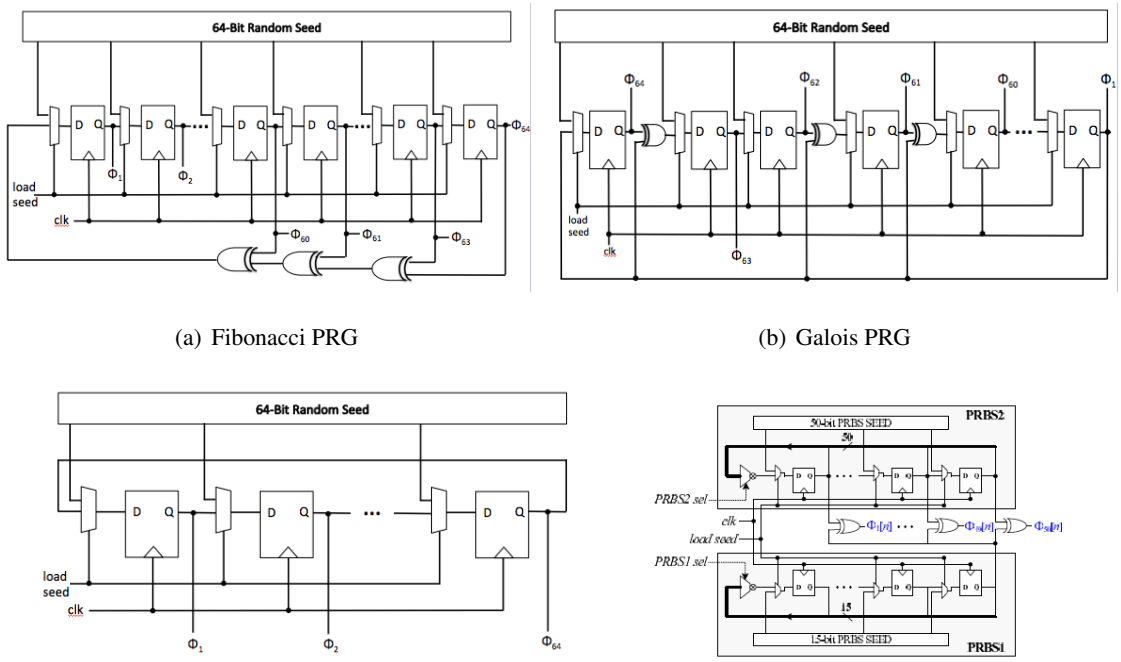
4.2.1 Look-Up Table

The most straightforward random matrix generation is a look-up table. The advantage to this approach is user/designer control of the random matrix. If a particular random sensing matrix is known to produce especially good results, that matrix is simply loaded into the on-chip memory. However, the size and area of such memory is significantly greater than that of the rest of the encoder. For example, if the CS encoder from [1] is used, then the maximum number of rows of the sensing matrix is M=64. Recall that this encoder adjusted the number of rows to achieve the desired compression ratio. The compression ratio corresponding to M=64 was C=2, so the maximum sensing matrix size is 64×128. Therefore, the size of the memory required would be 8.192kB. In summary, on-chip SRAM sensing matrix solutions substantially add to the encoder's overall area.

4.2.2 Pseudorandom Number Generators

A preferable alternative to the look-up table memory approach is dynamic random matrix generation. Existing CS encoders perform column-wise matrix multiplication for optimum area efficiency. Consequently, only one column of the sensing matrix is needed per sample. This approach incorporates a pseudorandom number generator (PRG) to produce a M-bit random bit sequence per sample. The existing ECG CS encoders both employ PRG's for sensing matrix generation [17], [1].

Section 4.1 proved the success of a CS random matrix $\mathbf{\Phi}$ is directly linked to it's incoherence with the sampling matrix $\mathbf{\Psi}$. The chief concern of PRG design is whether or not it is "random enough". LFSR's are not truly random because their sequence eventually repeats. A select few



(a) Fibonacci PRG

(b) Galois PRG

(c) Circulant PRG

(d) Dual-PRG [17]

Figure 4.4: Hardware Pseudorandom Number Generators

popular pseudorandom number generators are presented and studied here. A few common PRG's are studied here to demonstrate the importance and determining factors behind proper PRG design. Figure 4.4 displays the PRG designs considered. Each PRG produces exactly 64 bits per sample. The input sample length was fixed at N=128. Although the full PRG will always produce a 64-bit sequence, subsets of the sequence are selected for use. For example, a compression ratio of C=4 would only use the first 32 bits of the PRG's output.

The first PRG solution considered in the Fibonacci LFSR. The Fibonacci LFSR produces a special type of matrix referred to as a Toeplitz matrix. A Toeplitz matrixone in which every left-to-right diagonal is constanthas the following form:

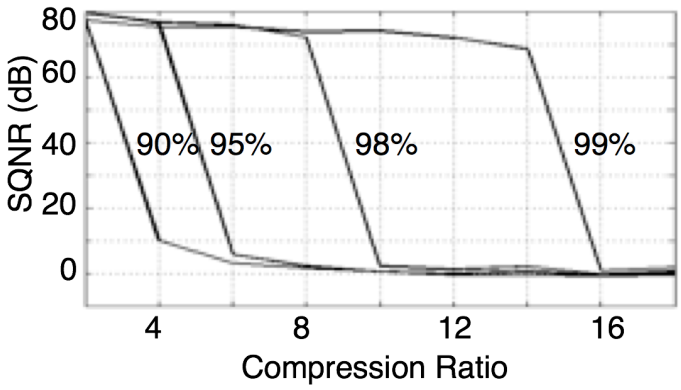
$$\mathbf{\Phi} = \begin{bmatrix} \phi_n & \phi_{n-1} & \cdots & \phi_1 \\ \phi_{n+1} & \phi_n & \cdots & \phi_2 \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{2n} & \phi_{2n-1} & \cdots & \phi_n \end{bmatrix} \tag{4.3}$$

The performance of general Toeplitz matrices was tested for the analog CS design [32]. The analysis, whose results are shown in Figure 4.5, measured the signal-to-quantization-noise ratio (SQNR) at different compression ratios for the independent and identically distributed (iid) Bernoulli matrix and the one-bit Toeplitx matrix. Each line in the figures represents a different sparsity level and the increasing sparsity achieves higher compression, as expected. The analysis confirms the viability of the Toeplitz sensing matrix.

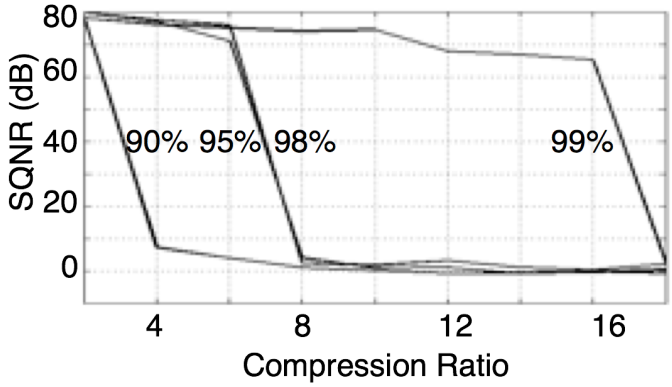
The Fibonacci LFSR implementation tested here is given in Figure 4.4(a). It is a string of *M* D flip flops with external XOR gates feedback. A random seed is loaded for the first iteration. A maximal-length Fibonacci LFSR is capable of producing 2^{n-1} different sequences (before repeating a previous sequence) where n is the number of bits in the LFSR. In this example, a maximal-length LFSR produces 2^{63} unique random sequences. Maximal-length LFSR's are achieved through selection of the "taps", or bits whose state influence the next LFSR state through feedback. A maximal-length Fibonacci LFSR for M=64 includes XOR feedback from registers 64, 63, 61 and 60 [33].

The next PRG solution considered is the Galois LFSR. The Galois LFSR implementation tested here is given in Figure 4.4(b). The Galois LFSR is similar to the Fibonacci sequence; it is a string of *M* D flip flops with feedback achieved using XOR gates. However, in the Galois solution, the XOR gates are internal rather than external. A random seed is loaded for the first iteration. Just like the Fibonacci LFSR, the Galois is capable of producing 2^{n-1} different sequences (before repeating a previous sequence) where n is the number of bits in the LFSR. The taps for the 64-bit Galois LFSR are the same as the Fibonacci LFSR.

The next PRG considered is the Circulant PRG. The circulant matrix is a derivative of the Toepltz matrix wherein each row is rotated one column to the right relative to the previous row:



(a) SQNR vs. Compression Ratio for the Bernoulli Matrix



(b) SQNR vs. Compression Ratio for the Toeplitz Matrix

Figure 4.5: SQNR vs. Compression Ratio for Different One-Bit Matrix Structures

$$\mathbf{\Phi} = \begin{bmatrix} \phi_n & \phi_{n-1} & \cdots & \phi_1 \\ \phi_1 & \phi_n & \cdots & \phi_2 \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{n-1} & \phi_{n-2} & \cdots & \phi_n \end{bmatrix} \tag{4.4}$$

The performance of circulant matrices were also tested for the analog CS design [32]. The circulant matrix was constructed by first forming a N-bit random Bernoulli column vector and rotating it N times to produce a $N \times N$ matrix. The $M \times N$ sensing matrix was then constructed from the first M rows of the first matrix constructed (in essence constructing another Toeplitz matrix). The results

yielded similar results to the Toeplitz results shown in Figure 4.5.

The circulant PRG implementation tested here us given in Figure 4.4(c). It is the simplest implementation, consisting of *M* D flip-flops with the final flip-flop acting as direct feedback to the first flip-flop. The LFSR is initially loaded with a random Bernoulli seed. This design differs from that tested previously in [32] because the maximum number of unique sequences is only M, where the previous experiment could achieve a maximum number of unique sequences of N. The importance of this difference and how it affects the viability of the circulant matrix implementation will be further discussed in Section 4.2.4.

The final PRG design considered is the dual-PRG configuration [17]. The Dual-PRG configuration (Figure 4.4(d)) consists of two separate PRG's [17]. The first PRG consists of M D flip-flops. The length of the first PRG is adjusted from 50 in the previous work to 64 for better comparison here. The feedback is only to the first flip-flop and is a randomly selected output from any of the 64 flip-flops. The second PRG consists of 15 D flip-flops. The feedback is again only input to the first flip-flop and is a randomly selected output from any of the 15 flip-flops. The $\mathbf{\Phi}$ entries are the output of each of the flip-flops of the 64-bit PRG XORed with the last bit of the 15-bit PRG.

4.2.3 Pseudorandom Number Generator Sensing Matrix Incoherence with the Sampling Matrix

First, it is important to note that the coherence between the $\mathbf{\Phi}$ of PRG's of Figure 4.4 and the time and wavelet sampling matrices presented in Chapter 3. Because the entries of the sensing matrix are only composed of +1/-1, the coherence of the sensing matrices developed by the one-bit PRG's is identical to that of the Bernoulli matrix. Coherence between the Bernoulli distribution and the time and wavelet matrices were given in Table 4.2.

It is natural to assume that because the coherence of the one-bit PRG's sensing matrix and the Bernoulli sensing matrix are identical, that the accuracy performance would be similar as well. The accuracy of the CS signal reconstruction was measured for the PRG's of Figure 4.4. The experiment was run for the time, level-1 Haar wavelet an level-1 Daubechies 4 wavelet. For each domain of the sparse signal, $\boldsymbol{\alpha}$ was thresholded to maintain 98% sparsity, $\boldsymbol{\alpha}_k$. The signal was compressed with each of the PRG's of Figure 4.4. For each sampling matrix/PRG combination, 1000 different sensing matrices were tested. Only compression C=2 was observed, and reconstruction was achieved via

	Fibonacci	Galois	Circulant	Dual-PRG
Time	0 ± 0.00	0 ± 0.00	70.72 ± 0.01	78.61 ± 6.32
Haar	0 ± 0.00	0 ± 0.00	70.71 ± 0.01	59.36 ± 25.32
Daubechies 4	0 ± 0.00	0 ± 0.00	64.64 ± 0.02	64.64 ± 33.13

Table 4.3: Experimental PRG $\mathbf{\Phi}$ Accuracy Summary

convex optimization.

Table 4.3 summarizes the accuracy results for the experiment, showing the PRD for each scenario (mean $\pm$ standard deviation). The results are alarming because the incoherence between the sensing matrix and sampling matrix seems to have no impact on performance. The Fibonacci and Galois LFSR's performed well in all sampling domains. For example, the Fibonacci and Galois LFSR's achieved $PRD = 0 \pm 0.00$ in the time domain, respectively. For all practical purposes, the Fibonacci and Galois LFSR's achieve perfect reconstruction. The circulant and dual-PRG configurations performed noticeably worse with $PRD = 70.72 \pm 0.01$ and $PRD = 78.61 \pm 6.32$ in the time domain, respectively. PRD values of such great magnitude mean the circulant and dual-PRG designs fail to reconstruct the compressed signal. At first, the finding may seem unintuitive since all PRG's achieved high incoherence with the sampling matrices tested. However, there is a more important factor at play called mutual incoherence. The impact of mutual incoherence is discussed in Section 4.2.4.

4.2.4 Pseudorandom Number Generator Sensing Matrix Mutual Incoherence

Pseudorandom number generators are the preferable hardware solution for sensing matrix generation, but their design is not without complications. In Section 4.1, this worked showed that the main design consideration for a sensing matrix populated with iid probability distribution functions was the coherence between the sensing matrix and the sampling matrix. However, Section 4.2.3 went on to show that the incoherence between the sensing matrix formulated by a pseudorandom number generator and the sampling matrix was not the sole basis for performance. In fact, there seemed to be little connection at all. That is because the pseudorandom number generator, by definition, defies

the principles of CS. The pseudorandom number generator is not truly random because it's next state is predictable and the sequence of states typically repeat after a certain number of iterations. The predictability of the PRG is actually a benefit to CS, giving the decoder knowledge of the sensing matrix if the PRG and random seed are known. The second condition, the repetition of the PRG, is detrimental to it's functionality as a CS sensing matrix generator. This does not mean that PRG's are a bad option, but rather certain measures must be taken to ensure the PRG is "random enough".

As an example, let's consider the circulant matrix. The M-long circulant PRG produces M different states before repeating itself. Therefore, the sensing matrix produces $\frac{N}{M}$ copies of itself. The previous experiment employed a 64-long circulant PRG to produce a 64×128 sensing matrix (C=2). The resulting sensing matrix would take the following form:

$$\mathbf{\Phi} = \begin{bmatrix} \phi_n & \phi_{n-1} & \cdots & \phi_1 & \phi_n & \phi_{n-1} & \cdots & \phi_1 \\ \phi_1 & \phi_n & \cdots & \phi_2 & \phi_1 & \phi_n & \cdots & \phi_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_{n-1} & \phi_{n-2} & \cdots & \phi_n & \phi_{n-1} & \phi_{n-2} & \cdots & \phi_n \end{bmatrix} \tag{4.5}$$

The result is a sensing matrix that can not produce enough unique measurements. In mathematical terms, the columns of the sensing matrix are no longer nearly orthogonal (the columns of the sensing matrix can never be exactly orthogonal because there are more columns than rows).

In order to determine whether a PRG's sensing matrix is "random enough", this work measures the sensing matrix's mutual coherence. Mutual coherence measures the coherence of the columns of the measurement matrix to each other [34], [35]. Mutual coherence is defined formally as

$$\mu_{mutual} = \underset{1 \leq j \neq k < N}{max} |\langle \theta_j, \theta_k \rangle| \tag{4.6}$$

where θ is a column of $\mathbf{\Theta}$ (recall from Chapter 2 that $\mathbf{\Theta} = \mathbf{\Phi\Psi}$). It is assumed throughout this work that $\boldsymbol{\theta}$ is a unit vector ($||\boldsymbol{\theta}||_2 = 1$). Consequently, the range of mutual coherence is given by $\mu_{mutual} \in [\frac{1}{\sqrt{M}}, 1]$. If the mutual coherence is close to 1, then it has highly correlated columns and is thus not nearly orthogonal.

The mutual coherence was evaluated for the sensing matrices produced by the PRG's of Figure 4.4 and the time and wavelet sampling matrices presented in Chapter 3. The experiment was

	Fibonacci	Galois	Circulant	Dual-PRG
Time	0.43 ± 0.05	0.43 ± 0.04	1.00 ± 0.00	1.00 ± 0.00
Haar	0.50 ± 0.04	0.50 ± 0.04	1.00 ± 0.00	1.00 ± 0.00
Daubechies 4	0.86 ± 0.03	0.87 ± 0.03	1.00 ± 0.00	1.00 ± 0.01

Table 4.4: Experimental PRG $\mathbf{\Phi}$, $\mathbf{\Psi}$ Mutual Coherence Summary

conducted such that 1000 different sensing matrices (controlled by varying the initial seed) for the PRG's of Figure 4.4 were formulated.

Table 4.4 summarizes the mutual coherence of $\mathbf{\Theta}$ for different PRG sensing matrix/sampling matrix pairs. The mutual coherence produced by circulant and dual-PRG solutions are both equal to 1. Due to the high correlation between the measurement matrix's columns (some are equivalent in fact to produce $\mu_{mutual} = 1$), they are unacceptable choices for sensing matrix generators. This was evidenced by their performance presented in Table 4.3. The Fibonacci and Galois mutual coherence was not close to 1, and thus they were better capable of perfect reconstruction. The Fibonacci and Galois mutual coherence were very close in value.

4.3 Sensing Matrix Conclusions

The success of compressed sensing is attributed to random incoherent sampling. Random incoherent sampling guarantees that the measurements are nearly orthogonal (a factor made more important by the undersampled nature of CS).

A thorough study calculated the coherence between several different random probability distributions and the time and wavelet sampling domains of Chapter 3. Further, the results confirm the relationship between incoherence and CS performance for these iid randomly distributed sensing matrices. This study showed that one-bit random sensing matrices work as well or better than multi-bit sensing matrices. This finding is significant because the analog CS encoder for a one-bit random sensing matrix is substantially smaller and simpler than that of the multi-bit implementation.

Next, this work discussed various different hardware solutions for sensing matrix generation and concludes that pseudorandom number generators are efficient sensing matrix generators. In previous

CS implementations, it has often been taken for granted that any pseudorandom number generator will suffice. This work proves that the analog CS designer must consider both coherence between the sensing and sampling matrices and the sensing matrix mutual coherence of the pseudorandom number generator.

Ultimately, for the analog CS encoder with ECG applications, the Bernoulli random matrix generated by a Galois or Fibonacci LFSR leads to a good analog CS design.

Chapter 5

CS ANALOG ENCODER SYSTEM DESIGN

As evidenced from the existing CS solutions, there are many system design decisions that go into creating an analog CS system. Thus far, no existing work explores the impact of these different decisions. This chapter will step through various design choices for an analog CS ECG compression system. In doing so, it will present many factors that impact performance of the analog CS encoders. The work will show that some decisions are of little importance while others have resounding impact on CS system viability. Section 5.1 describes an ideal model for this dissertation's analog CS model, discusses some high-level design decisions and presents performance results for this model. Section 5.2 presents the impact of thresholding on diagnostic quality and compression capability. Sections 5.3 and 5.4 discuss the impacts of the typical errors seen in the ADC and analog CS row operator. Finally, Section 5.5 draws several conclusions about analog CS encoder design decisions.

5.1 The Analog CS Model

There are many different CS solutions. However, this work focuses exclusively on analog CS systems. The section will describe the model and architecture of the analog CS system developed for ECG applications. The block diagram of the model is given in Figure 5.1. The first subsystem, CS Pre-processing, is the only optional subsystem. The CS-Preprocessing is optional because it involves signal pre-amplification (specific to the ECG application) and thresholding. As will be later demonstrated, the thresholding is not mandatory, but results in higher signal sparsity and consequently higher compression ratio capability. The remainder of the system is fairly straightforward, consisting of three separate subsystems: measurement matrix generator, CS operator and the quantizer. This section will describe each subsystem, including their underlying design decisions for the ECG analog CS system.

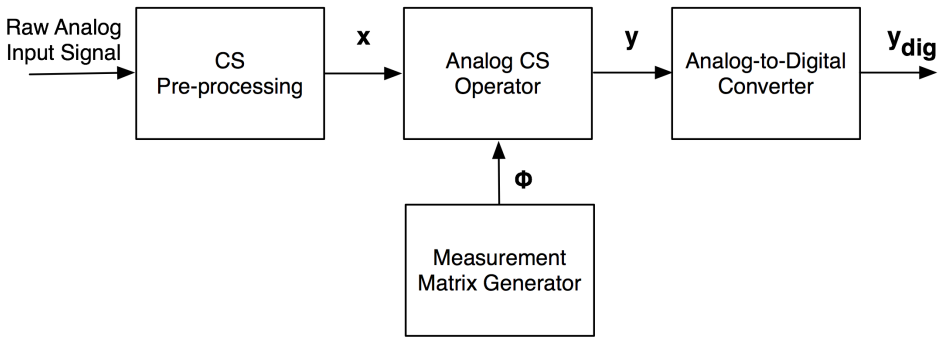


Figure 5.1: Analog CS Block Diagram

5.1.1 Analog CS Operator

The first mandatory subsystem to be considered is the CS operator. This dissertation aims to design the analog CS encoder of Figure 2.5(b). The analog CS system compresses the analog bio-signal by passing a sequence of N samples, $\mathbf{x}$, through a parallel array of M multiply and integrate stages to generate the M compressed output samples, $\mathbf{y}$. The system will fix the number of samples per frame, N. Thus, adjusting the number of rows utilized, M, achieves different compression ratios. The primary advantage of this approach over the alternative (fixing the number of rows, M, and altering the frame size N) is power savings. CS operator power consumption is dictated by the number of rows operating. Consequently, utilizing a smaller number of rows to achieve a higher compression ratio will also consume less power.

The maximum M on-chip is a major design consideration for the CS operator. It impacts both the maximum power consumption and the chip area. The following should be the guideline for maximum M:

$$M_{max} = \frac{N_{max}}{C_{min}} \tag{5.1}$$

The minimum desirable compression ratio, C_{min} and maximum frame size, N_{max}, constrain M_{max}. For this ECG system, we will design for $C_{min} = 1.5$ (for reasons discussed in Section 5.2.1). The frame size is dependent on the signal's nature and the sampling frequency. For periodic signals, such as ECG, the ideal frame size is given by

$$N = kTf_s \tag{5.2}$$

where k is an integer, T is the period of $\mathbf{x}$ and f_s is the sampling frequency of $\mathbf{x}$. Since Tf_s is the number of points contained in one period of the signal, the ideal N is simply an integer multiple of the number of points within the signal's period. In doing so, the signal sparsity is consistent. Since the hardware design goal is to minimize M_{max}, it is logical to set $k = 1$, making N the number of points within one signal period. For the ECG system model in this work, the data is sampled at $f_s = 100$Hz. The average adult human heart rate ranges from 60 to 100 beats per minute, equivalent to $T = 1 - 0.60$ sec respectively. Therefore, the N_{max} needed for this ECG system model is $N_{max} = 100$, corresponding to $M_{max} \approx 67$. However, Chapter 3 proved the tremendous benefit from using a level-3 wavelet sampling basis for the ECG signal and the level-3 wavelet basis is only fully represented by a sampling matrix whose size ($N \times N$) is an integer multiple of 2^3. Therefore, this design will aim for $N_{max} = 104$ with $M_{max} = 69$. Despite the fixed frame size operation, it is not necessary to use the maximum N_{max} outlined. In fact, it is advisable to calibrate N on a patient-by-patient basis according to their average heart rate. Doing so will achieve two benefits: the ideal frame size N will optimize system accuracy performance and using a more appropriate value for N, rather than N_{max}, will consume less power for the equivalent compression ratios.

It is appropriate to now take a step deeper into the system architecture and discuss the CS row operator. The row operator will be based on the analog CS design of [1], described in Section 2.3.2. This analog CS row operator for this design is given in Figure 5.2. The main difference between the design presented in Figure 5.2 and that of [1] is the length of the C-2C ladder. Section 4.1 demonstrated the one-bit measurement matrices (Bernoulli and Sparse Bernoulli) had a lower coherence with the commonly used ECG sampling matrices (time and wavelet domains) than that of the multi-bit measurement matrices, resulting in superior CS accuracy performance. Consequently, the row operator in Figure 5.2 does not implement a full C-2C ladder, but rather a simple switch capacitor design. This design not only ensures better performance but also consumes less area.

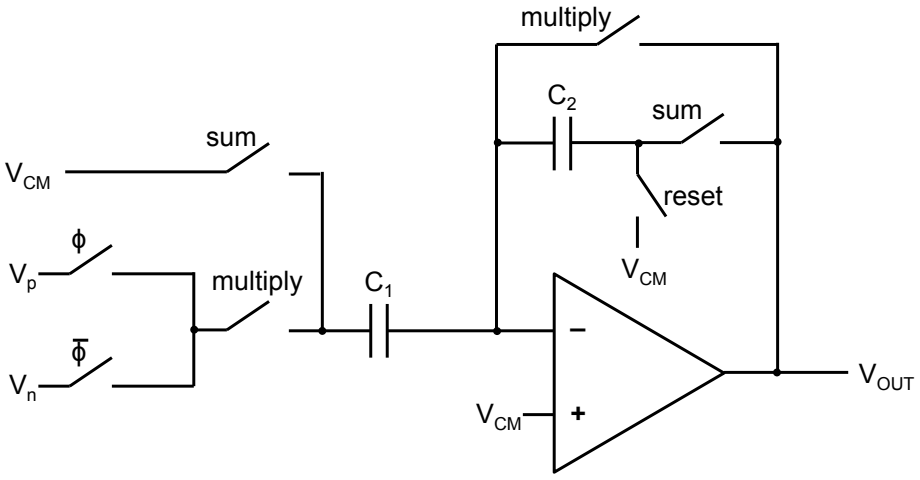


Figure 5.2: Analog CS Row Operator

5.1.2 Measurement Matrix Generator

The next system of the analog CS model is the measurement matrix generator. The sole function of the measurement matrix generator is to produce an effective measurement matrix for CS operation. The success of a measurement matrix is based largely on two factors: incoherence between the measurement and sampling matrix and mutual incoherence within the sensing matrix. Chapter 4 presents a thorough discussion, and study, on finding the optimal measurement matrix. Section 4.1 showed the impact of incoherence between truly random measurement matrices and sampling matrices, resulting in the revelation that one-bit matrices demonstrated larger incoherence with the commonly used ECG sampling matrices (time and wavelet domains) than that of the multi-bit measurement matrices. In Section 4.2.4, the study emphasized the importance of correct one-bit measurement matrix generator design. For ECG applications, Galois and Fibonacci LFSR's were the most effective designs. There was a negligible difference between the Galois and Fibonacci LFSR's. Therefore, this work implements the Galois LFSR due to it's smaller propagation time. The Galois LFSR will be similar to that of Figure 4.4(b), except that it will consist of 69 total bits (one bit each for the 69 row operators as determined in Section 5.1.1), with taps located at registers 69, 67, 42 and 40 [33]. At initialization, the LFSR is loaded with a 69-bit random seed.

5.1.3 Analog-to-Digital Converter

The final step in the analog CS model is quantization, performed by an analog-to-digital converter (ADC). Each CS row operator contains it's own ADC that converts the compressed signal, $\mathbf{y}$, into the digital compressed signal, $\mathbf{y}_{dig}$, ready for wireless transmission. The aim of this work will be to achieve similar performance as if the signal had been quantized (without compression) by an 8-10 bit ADC. In other words, this work will strive for 48-60 dB SNR. For an adequately sparse signal, the SNR of the CS reconstructed signal is similar in magnitude to that of the compressed signal. This will be further verified throughout this chapter. Therefore, the design specifications for the CS ADC are driven by the desired SNR of the compressed signal. The analog CS model for this application is composed of 10 bits.

5.1.4 CS Pre-Processing

The CS-Preprocessing block in the CS system is optional and it's design is highly dependent on application. For the ECG application developed here, the CS-Preprocessing includes pre-amplification and optional thresholding.

The pre-amplification simply amplifies the signal prior to CS operation. The pre-amplifier would be present even in the non-CS system. However, the gain specification will vary between non-CS and CS systems. For example, the pre-amplifier of a non-CS system would amplify the signal to meet the dynamic input range of the ADC. The typical ECG signal has a range of approximately -5mV to +5mV. Therefore, for an ADC with maximum input dynamic range of 600mV, the preamplifier gain of a non-CS system should be approximately 36 dB. On the other hand, for the CS system, the gain of the pre-amplifier needs to amplify the signal such that the compressed signal $\mathbf{y}$ meets the dynamic input range of the ADC. This will depend on several factors: frame size (N), measurement matrix type and magnitude, etc. To address this issue, the range of the compressed signal was determined experimentally. Twenty hours of ECG data, spanning 20 different patients, was compressed using CS. The ECG signals from the patients ranged from -5mV to $+5$mV. The compression parameters are consistent with those outlined in Sections 5.1.1 and 5.1.2. The frame size, N, is calibrated per patient to cover approximately one period of the signal. The compression ratio is C=1.5 to emulate the maximum possible compressed signal magnitude discussed here. Com-

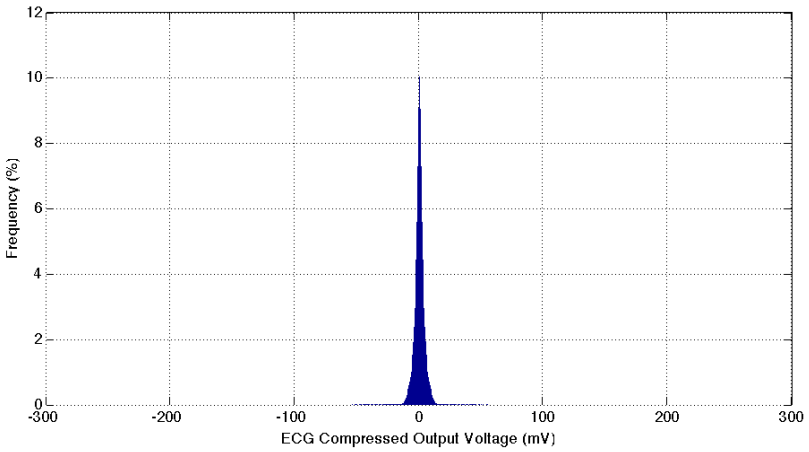


Figure 5.3: ECG Compressed Output Magnitude Distribution

pression was achieved by multiplying the ECG input with a random +1/-1 sequence produced by a 69-bit Galois LFSR initially loaded with a random seed. Figure 5.3 shows the distribution of the compressed signal magnitude of this experiment. The histogram shows that the compressed output for a ECG signal compressed using these system parameters spans about -97mV to +68mV. For an ADC with a maximum input dynamic range of 600mV, it is instinctive to assume that the needed pre-amplification is either nonexistent or very small ($\approx$ 9.5dB). However, upon further investigation, it is also clear that the vast majority of the distribution is much smaller than the maximum and minimum limits. In fact, 98% of the distribution lies in the -9.4mV to +9.4mV range. For the time being, for this system to span the 600mV ADC range, the model employs a pre-amplification of 32X (30dB). However, Section 5.3.1 will go on to show the importance of pre-amplifier gain patient calibration. Again, pre-amplification is highly dependent on the particular CS system parameters and should be checked to determine the necessity of the system.

CS pre-processing also includes optional thresholding. Section 5.2 presents two possible ECG analog CS models: one with thresholding and one without (or very little) thresholding. The system with minimal thresholding aims to satisfy the requirements of the AAMI/ANSI ECG monitoring device standards (outlined in Section 2.4). This system will be referred to as the "High Accuracy Low Thresholding" (HALT) system. It should be noted that previous CS implementations do not

attempt to cover the possibility of a HALT system. The system that incorporates thresholding optimizes for compressibility. Thresholding yields higher signal sparsity, and thus higher compression ratios. This system will be referred to as the "High Compression and Thresholding" (HCT) system. While the CS reconstruction is capable of perfect signal recovery, signal accuracy is diminished by the thresholding itself. Thus, the HCT system strives to achieve the highest compression possible while maintaining signal integrity. For the case of the ECG signal, that means trying to maintain as many of the diagnostic features as possible. This is the popular method of CS methodology, implemented in the previous CS systems discussed in Section 2.3. Despite the popularity of thresholding in previous CS systems, it's hardware realization is not discussed. There are several potential thresholding methods. Since the thresholding mechanism must exist in the hardware before the rest of the CS encoder, this work promotes a "simplicity is key" ideology. As such, all thresholding will be achieved using absolute thresholding. Absolute thresholding simply compares each input ECG signal value to a predetermined threshold limit. If the values is larger than the threshold limit, the ECG signal is passed. Otherwise, the signal point is set to zero. The absolute threshold hardware is realized simply through the use of two comparators (for positive and negative threshold limits). Further, the thresholding will be highly patient dependent and should be calibrated to the patient at initialization.

5.1.5 The Ideal Analog CS System Model Performance

This section addresses the success and limitations of the compressed sensing system based solely on the CS operation, before any other non-idealities are introduced.

The ideal model of the system was modeled in Matlab. Twenty hours of ECG data (one hour each from 20 different patients) was broken into the ideal frame size (Section 5.1.1) and thresholded using the absolute thresholding algorithm. A 69-bit Galois LFSR compresses the signal. The ADC was simply modeled by adding white noise to the compressed signal. Since the ADC described in Section 5.1.3 consists of 10 bits, the white noise was added such that the SNR of $\mathbf{y}$ was $\approx$ 60dB. This experiment tests compression ratios of 2X, 4X and 8X. The signals were reconstructed using convex optimization with double precision. Figure 5.4 shows the SNR of the reconstructed signal with respect to input signal sparsity. As expected, the accuracy (SNR) of CS reconstruction increases

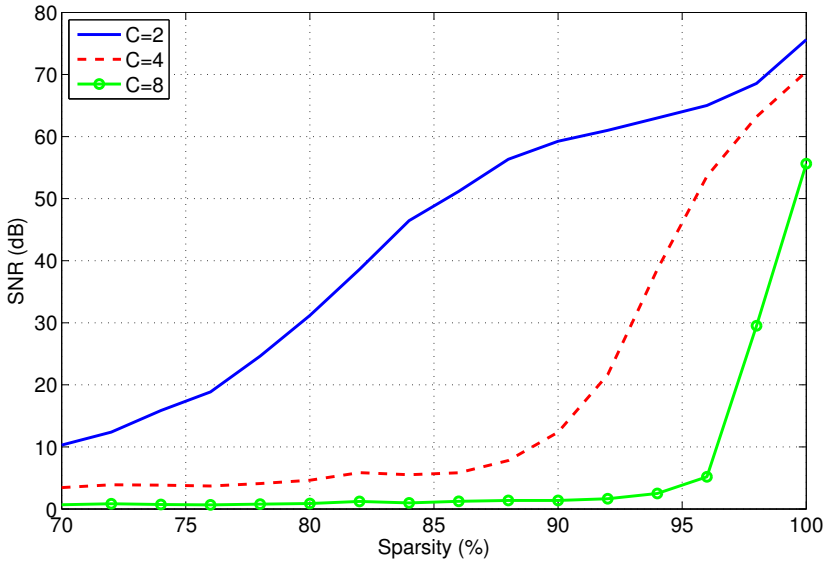


Figure 5.4: Analog CS System with ADC Noise Model ECG Sparsity vs. SNR

with signal sparsity. Further, the SNR increases sooner for lower compression ratios. For example, assume the target SNR were 48 dB (comparable to the input signal simply being quantized by an 8-bit ADC with with CS algorithm). At that SNR specification, the C=2X compression is achievable for signal sparsity $> 84.7\%$, the C=4X compression is achievable for signal sparsity $> 94.3\%$ and the C=8X compression is achievable for signal sparsity $> 99.4\%$. The maximum achievable SNR is also noteworthy. Despite being reconstructed using a double-precision value, the maximum SNR of the CS reconstructed signal is comparable to the SNR of the compressed signal! This finding is of significant importance, and is used to guide all design decisions regarding the analog CS ADC.

5.2 Thresholding and Diagnostic Quality

Thresholding is an important topic in compressed sensing. A crucial trade-off exists between signal sparsity and compression capability. Thresholding controls that balance. Figure 5.5 demonstrates the impact of thresholding on the ECG's diagnostic quality. As the absolute thresholding limit increases, the signal sparsity also increases. In Figure 5.5, minimal thresholding results in a signal

with 68% sparsity. Further, this signal is mostly still intact. The distinguishing features of the P, Q, R, S and T waves are still present. Increasing the thresholding limit increases sparsity even further but at the cost of losing ECG features. In this example, the signal thresholded to 93% sparsity provides information regarding the patients QRS complex and the signal thresholded to 99% sparsity provides only an ECG R peak. Thresholding limits needed to produce signals with certain diagnostic features are patient specific. However, the sparsity levels containing the various diagnostic features using this thresholding algorithm are fairly consistent. This section will address two very different approaches to the thresholding dilemma. First, the High Accuracy Low Thresholding (HALT) system is presented in Section 5.2.1. This system aims to implement minimal thresholding and provide an accurate representation of the original ECG signal. The success of the HALT system is measured in the strictest possible terms: ANSI/AAMI portable ECG device standards compliance. Then the High Compression and Thresholding (HCT) system is presented in Section 5.2.2. This system recognizes the value of the power savings in a higher compression CS system. Consequently, it will implement higher thresholding to achieve higher sparsity. While signal accuracy is seemingly less significant than that of the HALT system, the HCT system recognizes that key ECG diagnostic features (such as the QRS complex) can still be delivered. Because sparsity is greatly increased, CS has a higher guaranteed performance. Consequently, the HCT system performance is measured by the SNR of the thresholded signal and the reconstructed signal.

5.2.1 High Accuracy Low Thresholding Ideal System Performance

First, this work will discuss the performance of the ideal analog CS encoder without any CS preprocessing thresholding. For the strict definition of sparsity employed here where only those values with a value of zero are included in the sparsity count, the ECG signal shows very low sparsity in both the time and wavelets domains. By that logic, the very straightforward view of CS would indicate very poor performance for a system without thresholding. Regardless, CS is resilient to noise, so work was completed to test the system's capabilities without thresholding nonetheless. The same twenty hours of ECG data from twenty patients was amplified but not thresholded. The signal was then compressed using the ideal analog CS model of Section 5.1. Since the unthresholded signal does not show significant sparsity, the HALT system only uses a low compression factor of C=1.5.

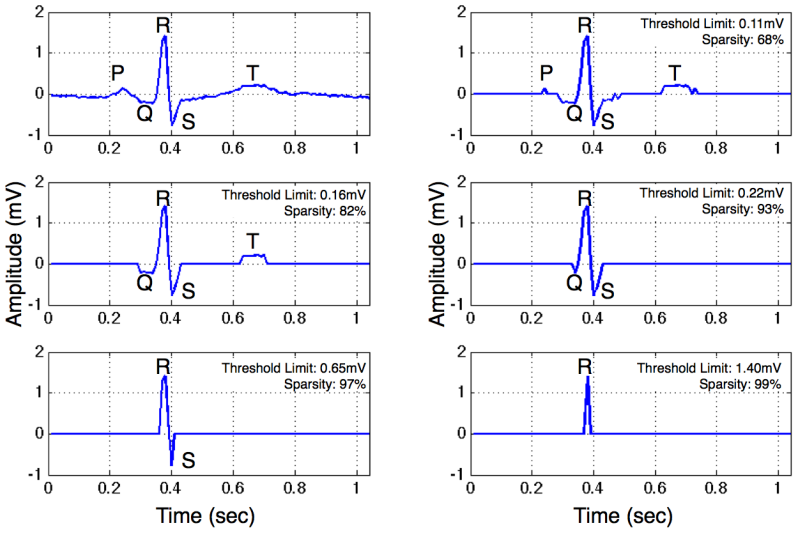


Figure 5.5: Impact of Thresholding on ECG Diagnostic Quality

CS reconstruction was achieved via basis pursuit convex optimization. Table 5.1 shows the performance of the experiment using both the time and level-3 Haar wavelet sparsifying domains. For an unthresholded signal, the level-3 Haar wavelet basis is superior by every metric. The main objective for the system without thresholding is to achieve ANSI/AAMI compliance. As demonstrated in Table 5.1, this is highly dependent on the patient. Of the twenty patients tested, eight patients achieved ANSI/AAMI compliance more than 90% of the time! The ANSI/AAMI compliance for the system without thresholding and C=1.5 over twenty patients was 68.71% and 84.31% for the time and level-3 Haar wavelets, respectively. Table 5.1 also shows the framed and total SNR performance. These values are significantly lower than those achieved with thresholding. However, that has little bearing on the ANSI/AAMI compliance and is of lesser consequence for the HALT system.

The analysis naturally progresses to show the impact of thresholding on the ideal HALT system. Because the ANSI/AAMI standards state that an error under $100\mu V$ is acceptable, it is logical to next observe the system's performance for a system with thresholding limits at $\pm 100\mu V$. Table 5.2 shows the experiment repeated for $\pm 100\mu V$ absolute thresholding. The thresholding showed some improvement in ANSI/AAMI compliance for CS in the time domain. SNR performance was also

	Overall SNR (dB)		SNR ($\sigma \pm \mu$,dB)		ANSI/AAMI Compliant (%)	
Patient	Time	Haar	Time	Haar	Time	Haar
1	9.09	12.31	9.40 ± 2.27	13.24 ± 3.23	80.08	91.94
2	4.93	8.13	5.39 ± 2.01	9.31 ± 2.83	86.41	95.70
3	11.60	14.24	11.80 ± 2.30	15.22 ± 3.15	65.72	81.12
4	12.35	15.52	13.14 ± 3.45	17.21 ± 4.43	85.38	94.34
5	5.08	10.14	10.99 ± 4.12	15.39 ± 4.24	56.71	76.40
6	8.79	10.93	9.89 ± 2.52	11.96 ± 3.04	97.77	98.97
7	9.82	14.44	11.87 ± 3.49	16.39 ± 3.99	52.80	76.02
8	10.33	15.91	12.01 ± 3.94	18.18 ± 4.56	50.54	79.93
9	8.24	11.62	8.54 ± 1.70	12.45 ± 2.76	86.74	95.99
10	6.09	10.89	9.38 ± 2.49	13.10 ± 3.10	72.29	87.50
11	7.90	11.78	8.17 ± 1.68	12.59 ± 2.78	64.35	85.25
12	8.54	9.97	11.36 ± 3.44	12.37 ± 3.65	99.39	99.45
13	7.70	13.06	10.84 ± 4.04	15.90 ± 4.29	46.85	71.50
14	8.45	11.86	8.84 ± 2.41	12.87 ± 3.39	78.52	91.12
15	2.74	11.55	9.80 ± 4.50	15.54 ± 5.06	49.86	76.27
16	4.32	10.10	12.62 ± 4.25	17.18 ± 4.49	55.79	78.11
17	4.66	8.81	5.80 ± 2.61	10.02 ± 3.22	40.48	60.93
18	8.24	11.55	8.58 ± 1.72	12.45 ± 2.79	87.57	96.34
19	4.59	11.87	11.55 ± 4.92	15.56 ± 4.54	56.65	74.25
20	2.61	7.08	5.72 ± 2.95	9.01 ± 3.41	60.30	75.11

Table 5.1: Unthresholded ECG CS System Model Performance at C=1.5

	Overall SNR (dB)		SNR ($\sigma \pm \mu$,dB)		ANSI/AAMI Compliant (%)	
Patient	Time	Haar	Time	Haar	Time	Haar
1	12.46	14.83	11.61 ± 3.03	12.30 ± 2.44	88.93	93.00
2	9.79	10.24	7.32 ± 2.12	7.22 ± 1.71	94.62	94.70
3	12.73	15.07	12.80 ± 2.85	14.91 ± 2.91	70.36	81.94
4	15.83	17.83	14.63 ± 3.75	14.91 ± 3.36	91.96	93.69
5	4.94	11.57	11.94 ± 5.08	14.98 ± 3.93	59.73	74.91
6	16.85	20.17	9.85 ± 1.67	9.77 ± 1.18	99.74	99.64
7	9.93	13.63	12.34 ± 4.09	15.69 ± 3.73	53.55	71.85
8	10.38	15.58	12.46 ± 4.52	17.41 ± 4.11	51.21	76.21
9	15.36	16.53	10.79 ± 1.67	10.83 ± 1.39	96.56	97.08
10	6.75	10.33	11.00 ± 3.55	12.21 ± 2.65	78.01	85.31
11	8.59	12.16	8.78 ± 2.08	12.00 ± 2.37	65.71	81.78
12	9.40	14.56	8.42 ± 2.46	8.39 ± 2.46	99.61	99.56
13	7.70	12.42	11.31 ± 4.73	15.56 ± 4.09	47.70	68.75
14	11.58	14.49	11.16 ± 3.22	12.06 ± 2.43	87.74	92.77
15	2.79	10.66	10.41 ± 5.15	14.88 ± 4.96	51.84	73.57
16	4.15	9.63	13.29 ± 5.00	16.50 ± 4.18	57.60	74.94
17	4.79	8.66	6.13 ± 2.97	9.70 ± 3.15	41.22	59.03
18	15.49	16.79	10.77 ± 1.68	10.87 ± 1.32	96.78	97.51
19	4.65	12.20	12.61 ± 5.99	15.23 ± 4.27	59.86	72.97
20	2.67	7.90	6.74 ± 3.77	8.52 ± 3.16	63.69	73.65

Table 5.2: ECG CS System Model Performance at C=1.5 and $\pm 100 \mu V$ Fixed Thresholding

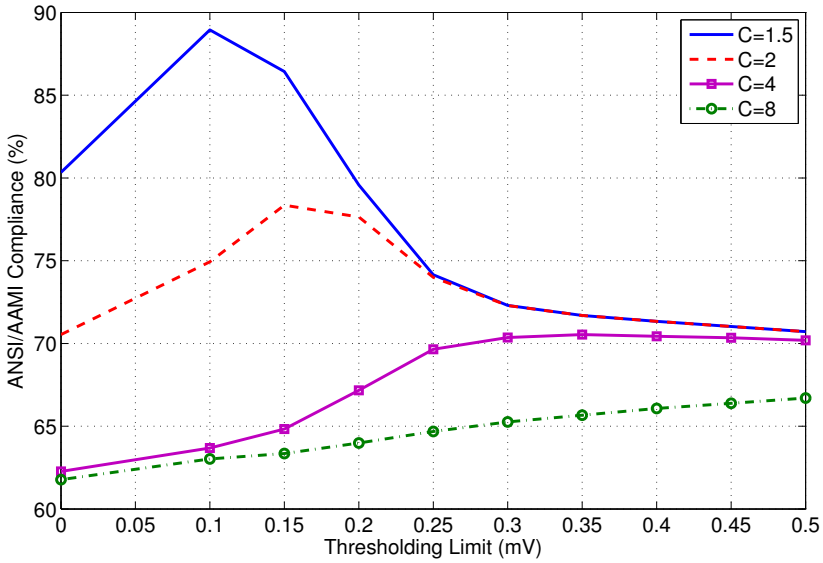


Figure 5.6: Effect of Thresholding on ANSI/AAMI Standards Compliance

improved. The ANSI/AAMI compliance for the level-3 Haar wavelet domain showed little change from that without thresholding. This is largely due to the fact that the thresholding is performed in the time domain and has little effect on level-3 Haar wavelet sparsity. Thresholding can be performed in the wavelet domain but would require wavelet basis conversion in the CS encoder (whereas now the wavelet analysis is confined to only the decoder) and would increase the hardware complexity of the analog CS encoder. Thus, further thresholding techniques are left to future work. The ANSI/AAMI compliance for the system with $\pm 100\mu V$ absolute thresholding and C=1.5 over twenty patients was 72.82% and 83.14% for the time and level-3 Haar wavelets, respectively.

As thresholding increases, sparsity increases and thus compressed sensing performance. Unfortunately, information of the ECG signal is permanently lost through thresholding. Figure 5.6 shows the ANSI/AAMI compliance as a function of thresholding limit for Patient #1 at several different compression ratios for the time domain sparsifying basis. For C=1.5X, as thresholding increases past $100\mu V$ the ANSI/AAMI compliance decreases as expected.

The overall takeaway about the HALT system is that it can achieve ANSI/AAMI standards

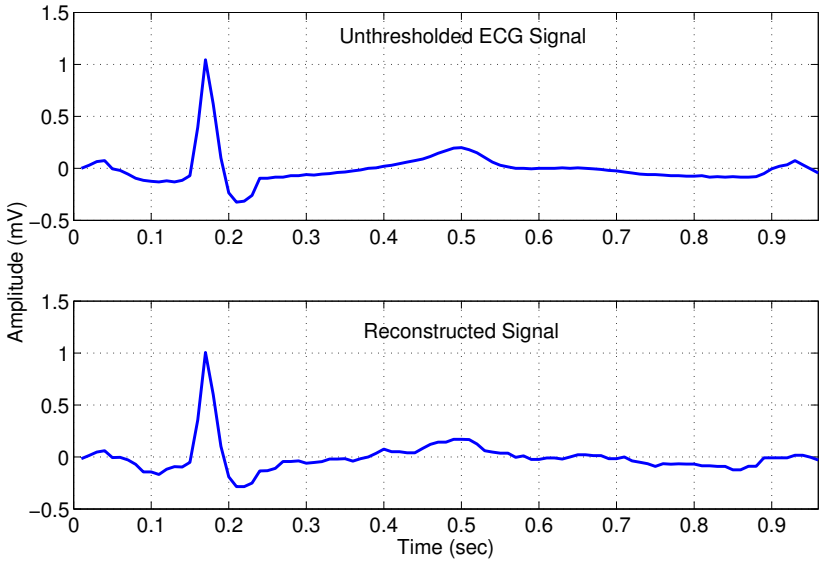


Figure 5.7: Sample ECG Signal CS Compression without Thresholding

compliance for a reasonable amount of the time. This can be achieved with little or no thresholding. Figure 5.7 shows a sample unthresholded ECG signal that underwent CS compression at C=1.5X. Reconstruction was achieved via basis pursuit convex optimization and the use of level-3 Haar wavelet sparsifying domain. The advantage is that signal is left intact, and doesn't experience degradation from thresholding. Performance is largely dependent on the patient's signal morphology (i.e. the length of time between major signal activity relative to the time of major signal activity, signal amplitude, etc.). However, it is very important to recognize that the achievable compression is very low, and thus draws into question the motivation for such a system.

As such, the remainder of the work will not focus on ANSI/AAMI compliance but rather the SNR between the thresholded signal and the reconstructed signal. In doing so, the performance for much higher compression ratios is observed. Section 5.2.2 will step through the justification of the thresholding from a diagnostic quality perspective and demonstrate the impact of thresholding on CS compression capability.

5.2.2 High Compression and Thresholding System Performance

The alternative system is the High Compression and Thresholding (HCT) system. The HCT system more rigorously thresholds a signal to ensure high signal sparsity. In doing so, the CS decoder is able to reconstruct the thresholded signal with very high accuracy. Due to this predictable nature, the HCT system is the most popular method of compressed sensing and will thus be the system explored for the remainder of this work. As previously discussed (Section 5.1.4), this work only explores the absolute thresholding technique.

The drawback to this system is the thresholding itself. The thresholding hardware adds extra complexity. However, the extra hardware is minimal in comparison to the CS encoder, and will thus not be considered in this analysis. More importantly, the thresholding degrades the signal integrity. By thresholding a signal before compression, signal information is permanently lost. However, Figure 5.5 also shows that a thresholded signal can still provide valuable diagnostic information.

The HCT system does not strive for perfect signal integrity but rather maximizes diagnostic information and the compression ratio. As such, the focus on ANSI/AAMI compliance in the HCT system is not appropriate. The HCT performance will instead be measured by the SNR between the thresholded signal and the reconstructed signal. To consider the effectiveness of the HCT system, observe the thresholded ECG signals of Figure 5.5. An ECG signal thresholded to 93% still contains information pertaining to the patient's QRS complex. According to the ideal system performance given in Figure 5.4, compression ratios of 2X and 4X are capable of 58.5 and 28.2 dB SNR between the thresholded and reconstructed signals, respectively. If an even higher sparsity signal is considered the performance continues to improves. For example, if the ECG signal sparsity were 97%, the ideal CS design yields 63.0 and 56.0 dB SNR for the 2X and 4X compression ratios, respectively.

5.3 Analog to Digital Converter Non-idealities

The first key component considered is the ADC. This CS architecture contains one ADC per row (total of M ADC's). Each ADC digitizes one compressed signal point per frame. This section will look closely at the consequences of various ADC errors (quantization, offset, gain and nonlinearity) to determine their overall effect on the analog CS system.

5.3.1 Quantization Noise

It is appropriate to first model the expected behavior of the ADC. An ADC converts a signal into a signal with a predefined resolution. The resolution of the ADC depends on the operating range ($[v_{min}, v_{max}]$) and the number of bits (N):

$$v_{lsb} = \frac{v_{max} - v_{min}}{2^N} \tag{5.3}$$

In doing so, the ADC's output signal ($v_{out,dig}$) can be thought of as the input signal (v_{in}) superimposed with a quantization noise (v_q) signal:

$$v_{out,dig} = v_{in} + v_q \tag{5.4}$$

The quantization noise is simply the difference between the output signal defined by the bit resolution and the input signal. Ideally, the quantization is limited by the voltage of least significant bit (v_{lsb}):

$$-\frac{1}{2}v_{lsb} \leq v_q < \frac{1}{2}v_{lsb} \tag{5.5}$$

Now, we can use this basic understanding of ADC's to estimate it's performance. The accuracy of the ADC is measured in terms of signal-to-noise ratio (SNR):

$$SNR = 20log\left(\frac{v_{in,rms}}{v_{q,rms}}\right) \tag{5.6}$$

The quantization noise is typically derived for the general case using a stochastic approach. It is assumed that the quantization noise is uniformly distributed between $\pm\frac{1}{2}v_{lsb}$. Thus, the probability distribution function of the quantization error, $f_q(x)$, using this assumption is defined as:

$$f_q(x) = \begin{array}{ll} \frac{1}{v_{lsb}} & -\frac{v_{lsb}}{2} \leq x \leq \frac{v_{lsb}}{2} \\ 0 & \text{otherwise} \end{array} \tag{5.7}$$

Finally, the RMS of the quantization error is derived as:

$$v_{q,rms} = \sqrt{\int_{-\infty}^{\infty} x^2 f_q(x)\,\mathrm{d}x} = \sqrt{\frac{1}{v_{lsb}}\int_{-\frac{v_{lsb}}{2}}^{\frac{v_{lsb}}{2}} x^2\,\mathrm{d}x} = \frac{v_{lsb}}{\sqrt{12}} \tag{5.8}$$

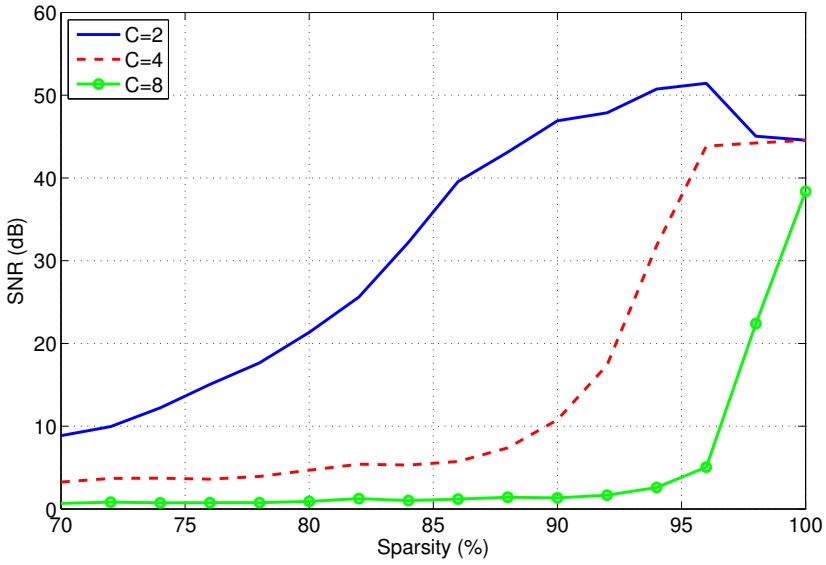


Figure 5.8: Analog CS System with Ideal SAR ADC and Fixed Pre-Amplification ECG Sparsity vs. SNR

Returning to Eq 5.6, the SNR is calculated from the RMS of the quantization error and the input signal. For example, if the input signal is a sine wave ranging the full scale of the ADC range, the resulting performance of the ADC is defined as follows:

$$SNR = 20log\left(\frac{v_{max} - v_{min}/2\sqrt{2}}{v_{lsb}/\sqrt{12}}\right) = 6.02N + 1.78\text{dB} \tag{5.9}$$

The first noise model of Section 5.1.5 is based on Equation 5.9 and provides a good understanding of the mechanics/performance of the analog CS encoder, but this work pushed the model another step closer to reality. The ADC noise model was replaced with an ideal 10-bit successive approximation ADC. An identical experiment was conducted as that for Figure 5.4. Pre-amplification was constant for all patients. Figure 5.8 shows the SNR of the reconstructed signal with respect to the input signal sparsity. The results are initially alarming! The maximum achievable SNR (at very high signal sparsity and low compression ratios) was only ≈ 45dB. This is a substantial difference from the initially anticipated 60 dB.

An investigation was conducted to understand the SNR performance of the ADC. The previous experiment showed that the SNR of the reconstructed signal was comparable to that of the compressed signal. Table 5.3 shows the SNR of the compressed signal for each of the twenty patients for C=2 only. All signals whose sparsity was not sufficient to achieve reconstruction were omitted. The table presents both the SNR of the patient's signal as a whole and the mean and standard deviation SNR of the individual compressed signal frames. The mean of the overall compressed signal SNR among all the patients was 38.96 dB. These results are consistent with the reconstructed signal SNR.

In order to achieve better CS performance, it is imperative to improve the ADC performance. Recall from Equation 5.6 that the SNR performance of the ADC is dependent on two factors: the RMS of the input signal voltage and the RMS of the quantization error voltage. Table 5.3 shows the compressed signal's RMS voltage and quantization error RMS voltage for the 20 patients at C=2X compression. This helps illuminate the various reasons for inadequate SNR. According to Equation 5.8, the quantization error for the ADC modeled here (input dynamic range of 600 mV and 10 bits), should be approximately $0.17V_{rms}$. Patient #9, 11, 14, 17 and 18 all have the expected quantization error voltage and an overall SNR greater than 48 dB. Patient #2, 6 and 12 all show the expected quantization error voltage but their overall SNR values were less than 48 dB. The difference between these patients and those with better SNR is the magnitude of the input compressed signal. The patients whose SNR was inferior had a far smaller y_{rms} value. Clearly, these patients would benefit from a higher gain in the CS Pre-processing block, thus increasing their signal power. Patient #1, 3, 4, 5, 7, 8, 10, 13, 15, 16, 19 and 20 all showed much higher than expected quantization noise and consequently degraded SNR. The reason for the high quantization error is that the signal power for these patients was much higher and the compressed signal often exceeded the ADC's input dynamic range, leading to saturation. These patients would clearly benefit from a smaller gain in the CS Pre-processing block.

The results presented in Table 5.3 lead to the conclusion that a CS encoder for ECG applications needs a variable gain amplifier in the CS Pre-processing block. A "one gain setting suits all" approach will not suffice. The optimal gain is dependent on both the patient and the compression ratio. The dependence of the SNR on compression ratio is substantially smaller than that of the patient and this works assumes that the effect of the compression ratio is negligible. Figure 5.9 shows

Patient	y_{rms}(V)	$y_{q,rms}$(V)	Overall SNR (dB)	SNR ($\sigma \pm \mu$,dB)
1	71.26	0.29	47.74	52.10 ± 2.85
2	22.33	0.17	42.51	43.85 ± 7.40
3	127.75	3.23	31.95	52.13 ± 9.86
4	96.53	0.51	45.54	54.75 ± 3.83
5	131.14	6.92	25.55	43.63 ± 14.59
6	30.33	0.17	45.05	45.04 ± 2.40
7	156.65	9.39	24.44	43.18 ± 14.35
8	154.96	6.70	27.29	44.49 ± 14.72
9	50.45	0.17	49.46	49.28 ± 1.77
10	77.95	0.57	42.77	52.77 ± 2.76
11	77.47	0.18	52.89	53.03 ± 1.62
12	21.83	0.17	42.40	43.06 ± 3.584
13	157.51	9.66	24.25	22.39 ± 23.02
14	68.38	0.18	51.65	51.64 ± 2.81
15	128.51	3.67	30.89	33.37 ± 25.32
16	162.65	13.44	21.66	41.04 ± 16.15
17	73.76	0.17	52.74	50.64 ± 5.76
18	51.27	0.17	49.62	49.47 ± 1.78
19	133.41	5.20	28.19	47.45 ± 13.25
20	44.15	0.33	42.60	47.91 ± 2.52

Table 5.3: ECG CS System Model Compressed Signal Voltage and Quantization Error Voltage for Fixed Gain

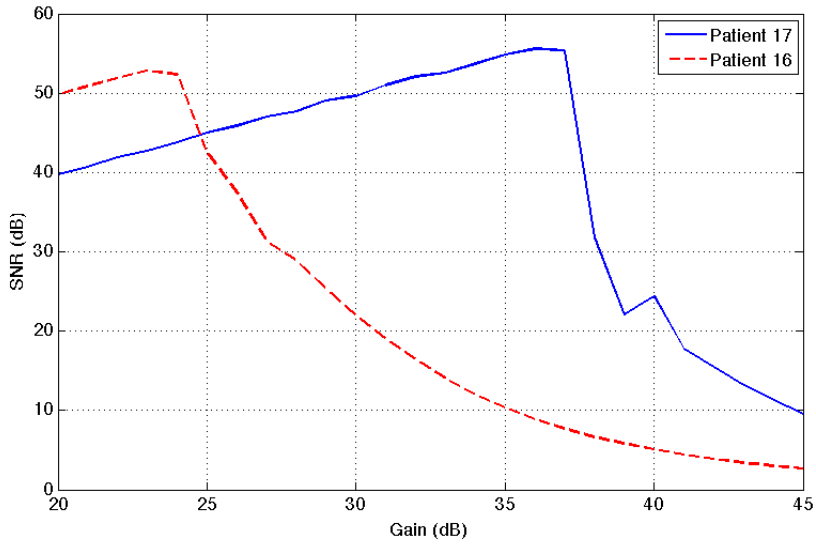


Figure 5.9: CS Pre-processing Gain vs. ECG Compressed Output SNR

the compressed signal SNR as a function of gain for Patients #16 and 17 at C=2. It illustrates the extreme difference in optimal gain settings between patients, as Patient #16's optimal gain is 22.5 dB and Patient #17's optimal gain is 32 dB. Furthermore, this figure shows the general trend for the maximum CS compressed signal SNR among all patients. Even under ideal conditions, the ADC performance for CS compression of ECG signals will not achieve that set out in Equation 5.9 (60 dB in this system). This is naturally due to the distribution function of the CS ECG compressed signal. Figure 5.3 shows a significant portion of the distribution lies near zero. Thus, the signal power ($v_{in,rms}$) is not on the same scale as a sine wave.

Table 5.4 shows the improved ADC performance for the system with optimal CS Pre-processing gain. The CS Pre-processing gain ranged from 22.5 dB to 39 dB. Quantization error RMS voltage was the expected 0.17 V for all patients. Most importantly, all patient's compressed signal overall SNR values were improved to more than 48 dB, with an average overall SNR of 52.41 dB. The same twenty hours of ECG data (spanning 20 patients) was sampled via the analog CS system. An ideal 10-bit SAR ADC was used for compression and the signals were calibrated per patient to utilize the optimal pre-processing gain. After compression, the ECG data was reconstructed using convex

Patient	Gain (dB)	y_{rms}(V)	$y_{q,rms}$(V)	Overall SNR (dB)	SNR ($\sigma \pm \mu$,dB)
1	29	62.64	0.17	51.12	51.02 ± 2.69
2	39	62.30	0.17	51.05	52.40 ± 5.50
3	26	78.16	0.17	53.11	52.22 ± 6.35
4	29	85.31	0.17	53.91	53.99 ± 3.29
5	26	81.68	0.17	53.77	50.26 ± 12.53
6	34	47.52	0.17	48.98	48.99 ± 2.43
7	24	77.32	0.17	53.27	52.88 ± 2.80
8	24.5	80.92	0.17	53.65	46.67 ± 17.59
9	33	70.30	0.17	52.36	52.18 ± 1.79
10	26.5	51.21	0.17	49.62	49.28 ± 2.45
11	30.5	79.36	0.17	53.43	26.14 ± 26.65
12	36.5	45.77	0.17	48.83	49.60 ± 4.25
13	24	76.85	0.17	53.14	24.33 ± 26.42
14	33.5	67.71	0.17	51.87	51.49 ± 2.93
15	27	88.27	0.17	54.39	29.01 ± 26.72
16	22.5	67.79	0.17	52.06	51.66 ± 3.03
17	32	93.24	0.17	54.87	50.48 ± 5.06
18	32.5	67.38	0.17	51.98	51.84 ± 1.72
19	26	82.76	0.17	53.77	53.39 ± 1.68
20	25.5	75.24	0.17	52.92	43.23 ± 2.35

Table 5.4: ECG CS System Model Compressed Signal Voltage and Quantization Error Voltage for Calibrated Gain

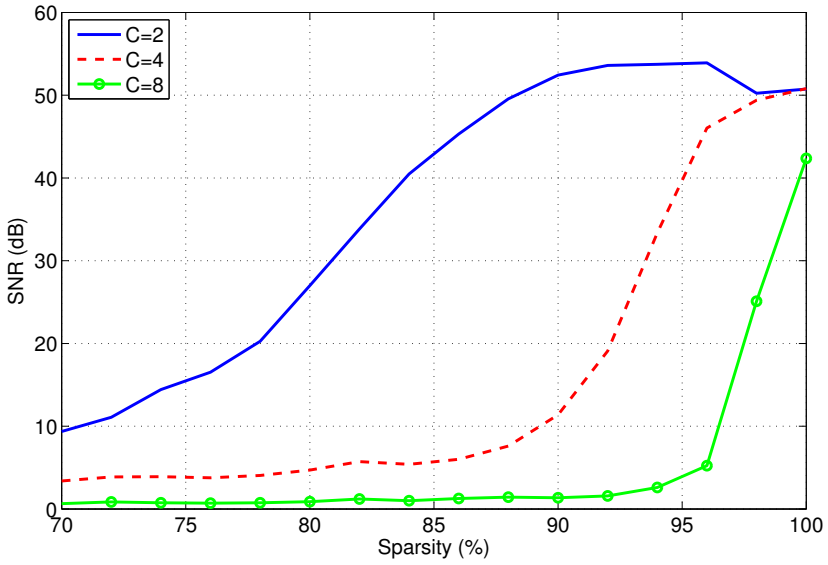


Figure 5.10: Analog CS System with Ideal SAR ADC and Calibrated Pre-Amplification ECG Sparsity vs. SNR

optimization with double precision. Figure 5.10 shows the results of ECG CS compression using the ideal calibrated analog CS system. The maximum achievable reconstructed signal SNR (at very high signal sparsity and low compression ratios) is now $\approx$ 52dB. If the target SNR is 42 dB, the C=2X compression is achievable for signal sparsity $> 86\%$, the C=4X compression is achievable for signal sparsity $> 95\%$, and C=8X is not possible.

5.3.2 ADC Offset

The next major source of error to be considered is the ADC offset. The effects of the ADC offset can be seen in Figure 5.11. The transfer curve of the ADC is simply shifted by the amount of the offset voltage, resulting in a potentially incorrect digital output. ADC offset voltage is often measured in terms of LSBs.

The ADC offset has a direct impact on the SNR of the compressed signal, which will translate to a degradation of SNR of the reconstructed signal. So, if the ADC contributed only quantization

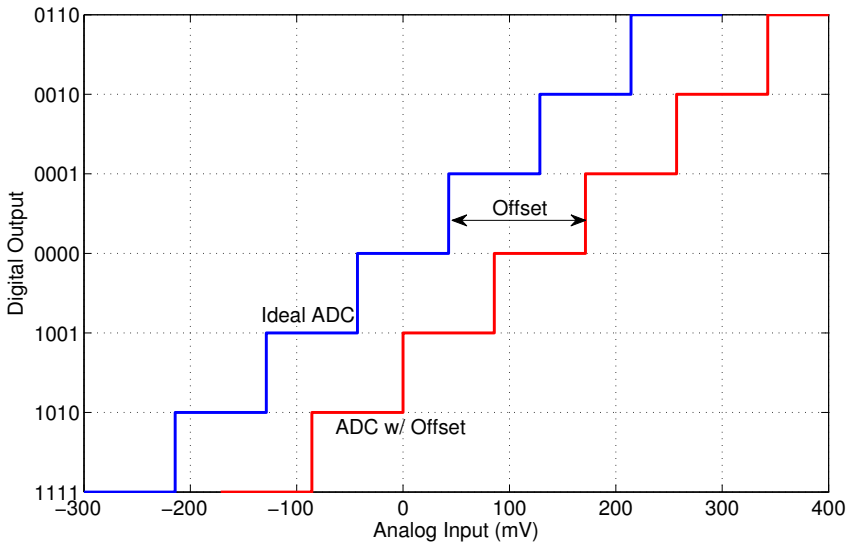


Figure 5.11: ADC Transfer Curve with Offset

noise and offset error to the compressed signal, the resulting SNR would be:

$$SNR = 20log\left(\frac{v_{in,rms}}{v_{q,rms} + v_{o,rms}}\right) \tag{5.10}$$

If the CS analog encoder time-interleaved the compressed signal through a single ADC, then the RMS offset voltage would be approximately constant (assuming no major drift in offset voltage over time). The analog CS encoder architecture presented in this work, however, consists of a single ADC per row (or an ADC per compressed signal value in a frame). Therefore, the offset voltage is modeled using a stochastic approach. It is assumed that the ADC offset is a random variable with normal distribution and zero mean. The probability distribution function of the offset voltage is therefore given by:

$$f_o(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-x^2/2\sigma^2} \tag{5.11}$$

Consequently, the RMS offset voltage is given by:

$$v_{o,rms} = \sqrt{\int_{-\infty}^{\infty} x^2 f_o(x)\,\mathrm{d}x} = \sqrt{\frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} x^2 e^{-x^2/2\sigma^2}\,\mathrm{d}x} \tag{5.12}$$

Figure 5.12 shows the effect of both the quantization noise and a normally randomly distributed ADC offset voltage on the SNR for Patient #1. Offset voltage is defined here as the standard deviation. First, the theoretical SNR based on Equations 5.10 and 5.12 are presented. The RMS values for the compressed signal, $v_{in,rms}$, and quantization error, $v_{q,rms}$, are taken directly from Table 5.4. As expected, the SNR of the compressed signal degrades according to Equation 5.10. Further, a CS experiment was performed on one hour of ECG data for Patient #1 using the analog CS model presented in Section 5.1 and added quantization and offset non-idealities. The SNR of the experimental compressed signal is consistent with the theoretical results. The SNR of the reconstructed signal for C=2 at 96% sparsity is also shown. The reconstructed signal SNR follows the same trend as the compressed signal SNR but with a slight positive vertical translation. The value of the translation is largely dependent on the sparsity level. The 96% sparsity level was chosen due to it's ability to retain most of the QRS complex in the ECG signal. The reconstructed signal at this sparsity level follows the same offset voltage degradation but consistently shows 3.12 dB better SNR performance.

The implications of Figure 5.12 are that offset has causes severe SNR degradation in the compressed signal and consequently the CS reconstructed signal. For example, an offset of 1 LSB in this system causes more than 10 dB loss in SNR performance. The source of the offset in a SAR ADC is offset in the comparator. Fortunately, incorporating offset-cancellation techniques is common [36].

5.3.3 ADC Gain Error

Another common source of ADC error is gain error. ADC gain error is a difference in slope between the ideal ADC transfer curve and the actual transfer curve. Assuming offset is zero and the ADC is linear, gain error is measured as the difference between the ADC output between the actual and ideal transfer curves. It is measured in terms of the number of LSBs the curves differ at the ADC curve's endpoints. Again, since the encoder consists of a single ADC per row, the ADC gain error is modeled stochastically. This work assumes the gain error is a random variable with normal distribution and zero mean.

Figure 5.13 shows the effect of both the quantization noise and a normally randomly distributed

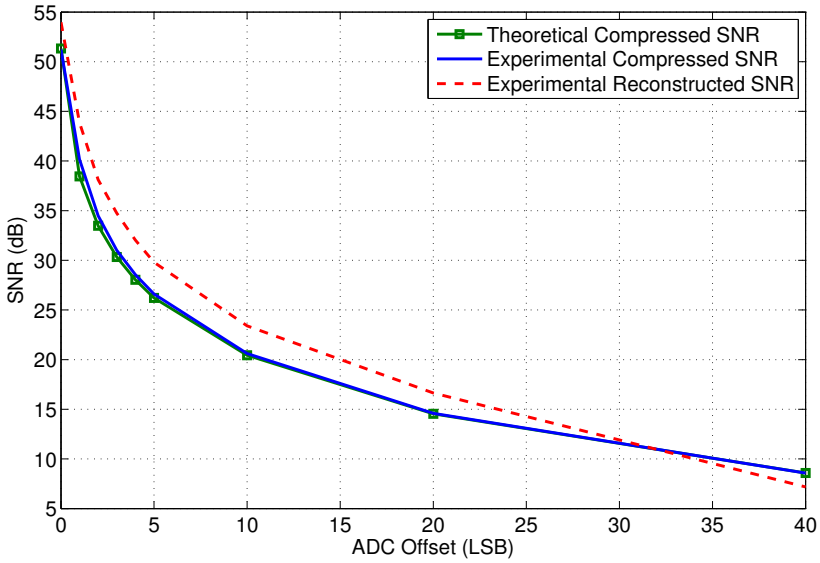


Figure 5.12: Analog CS System with ADC Offset vs. SNR

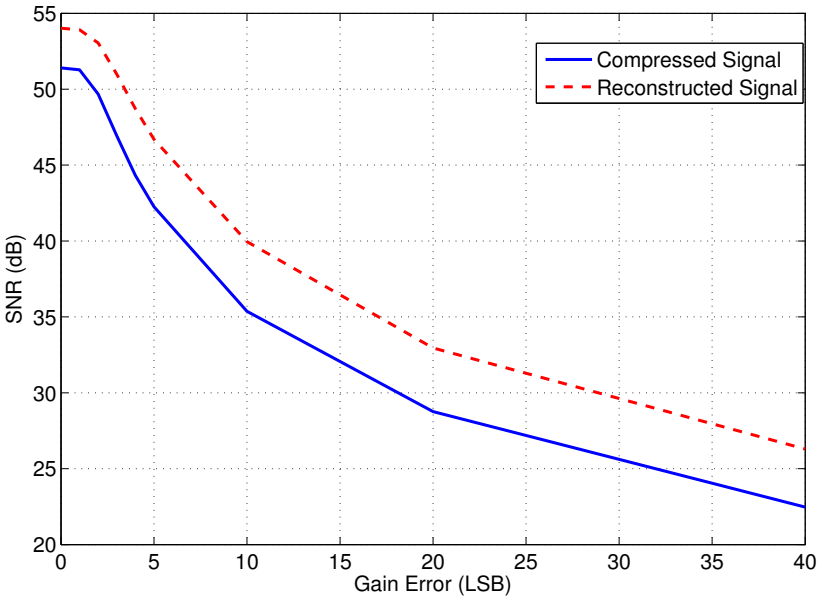


Figure 5.13: Analog CS System with ADC Gain Error vs. SNR

gain error on the SNR for Patient #1. Gain error is defined here as the standard deviation. As expected, the SNR of the compressed signal degrades with gain error. The SNR of the reconstructed signal for C=2 at 96% sparsity is also shown. The reconstructed signal SNR follows the same trend as the compressed signal SNR but with a slight positive vertical translation. The value of the translation is largely dependent on the sparsity level, being an average of 3.79 dB for the 96% sparsity level.

The gain error has a less severe impact than the offset error. For example, the SNR degrades less than 1dB for less than 2 LSB gain error. After 2dB gain error, the SNR drops off more, at a rate of approximately 1.64 dB/LSB until 10 LSB gain error. The gain error can also be easily calibrated and should be if the gain error is substantial [36].

5.3.4 ADC Nonlinearity

Probably the most challenging source of error present in the ADC is the nonlinearity. Nonlinearity measures the deviation of the ADC's transfer function from an ideal linear ADC transfer function. Nonlinearity is generally measured using two different parameters: integral nonlinearity (INL) and differential nonlinearity (DNL). INL measures the difference between the actual transfer function and the ideal transfer function (after the gain and offset error have been calibrated). While this figure is typically reported as the maximum value, it will also be shown in this work as a function of the digital word. INL is typically measured using two different methods: the endpoint line method and the best fit line method. The best fit line method (and the most commonly reported due to it's favorable figures) measures the difference between the actual ADC transfer curve and that curve's best fit line that minimizes the INL. The endpoint line method measures the difference between the actual ADC transfer curve and the straight line drawn through the ideal ADC transfer curve's endpoints. The ADC INL endpoint line method is formally defined as:

$$\text{INL}(B) = v_{out}[B] - v_{out}[0] - i\frac{v_{out}[i_{N-1}] - v_{out}[0]}{N-1}\left(\frac{1}{v_{lsb}}\right) \text{ LSB's} \tag{5.13}$$

DNL measures the separation of the adjacent levels measured at each step of the actual ADC transfer curve. Again, this value is often represented as the maximum seen by the system, but will also be shown here as a function of the digital word. ADC DNL is formally defined as:

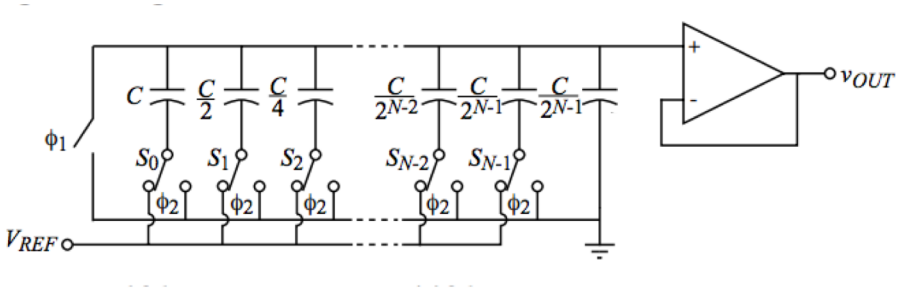


Figure 5.14: Binary-Weighted Charge Scaling DAC [2]

$$\text{DNL}(B) = \frac{v_{out}[B] - v_{out}[B-1]}{v_{lsb}} - 1 \text{ LSB's} \tag{5.14}$$

Unfortunately, ADC linearity is highly dependent on the architecture. The ADC modeled here is the SAR ADC, for which the linearity will originate from the DAC. For the capacitor-based designs considered here, the error will be largely due to deviation in the capacitors from their design value. In an effort to maintain the strictest accuracy, this work aims to study the effects of nonlinearity by modeling the error in the DAC's transfer curve resulting from capacitor tolerance error. The result is a model of absolute error, incorporating both nonlinearity and gain error.

As an example, the absolute error is shown here for a binary-weighted charge scaling DAC (Figure 5.14) [2]. The derivation for the absolute error, ϵ is given in Appendix B. In summary, the absolute error of a generic binary-weighted charge scaling DAC at digital input B is given as:

$$\epsilon(B) = \sum_{i=0}^{N-1} b_i \frac{2^{N-i-1}\Delta C_i}{C} \text{ LSB's} \tag{5.15}$$

In this definition, b_i is 1 or 0 depending on the presence or absence of a bit, N-1 is the total number of bits, $\frac{\Delta C_i}{C}$ is the tolerance of capacitor C_i and i represents the position of the observed bit where $i = 0$ is the most significant bit. Despite the difference in error characteristics among their architectures, the charge scaling DAC's absolute error is always dependent on the capacitor tolerance.

First, this section observes the relationship between absolute error of a binary-weighted charge scaling DAC and capacitor tolerance. Capacitor tolerance is modeled as a normally distributed

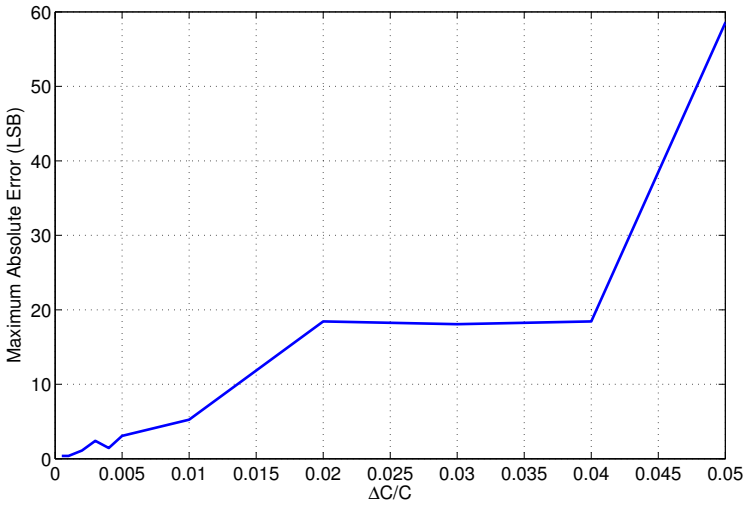


Figure 5.15: Binary-Weighted DAC Maximum Absolute Error vs. Capacitor Tolerance

random variable with zero mean and a standard deviation of $\sigma_{\frac{\Delta C}{C}}$ [2]. More than one million 10-bit binary-weighted SAR ADC's were modeled according to Equation 5.15 for varying values of $\sigma_{\frac{\Delta C}{C}}$. Figure 5.15 confirms that an increase in capacitor tolerance leads to an increase in DAC absolute error. For reasonable values of capacitor tolerance, the error metrics are also reasonable. For $\sigma_{\frac{\Delta C}{C}} < 0.01$, the average maximum absolute error is less than 5.5 LSB's. However, for $\sigma_{\frac{\Delta C}{C}} > 0.01$ maximum absolute error gets far worse, seeing average maximum absolute error of 58.55 LSB's at $\sigma_{\frac{\Delta C}{C}} = 0.05$.

Figure 5.16 plots the SNR of the compressed signal with respect to DAC absolute error. While there seems to be a relationship between absolute error and CS performance, the correlation is not as seemingly strong as the other parameters studied in this work. Because this analysis aimed to measure DAC nonlinearity in the most accurate manner possible (by modeling the DAC with capacitor tolerance errors), the DAC will also naturally incorporate gain error. This is appropriate for this analysis because this work assumes there will be no gain error calibration. However, this assumption has made it impossible to disambiguate the gain error and nonlinearity, causing the lack of a strong relationship between absolute error and SNR. Future work in this area is strongly

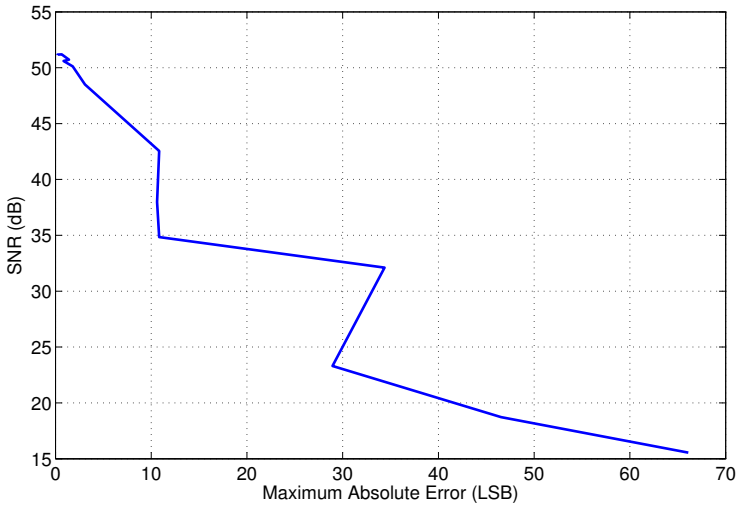


Figure 5.16: Binary-Weighted DAC Maximum Absolute Error vs. SNR

recommended to better determine the effects of nonlinearity to the system.

5.4 Analog CS Operator Non-idealities

The final key components of the analog CS encoder under consideration is the analog CS operator. The CS architecture contains one analog CS operator per row (total of M analog CS operators). The purpose of each analog CS operator is to simply multiply each of the N input signal points in the frame with their respective row of the measurement matrix and integrate. In doing so, each analog CS operator produces one of the M compressed signal points. The analog CS operator scrutinized in this design is a switch capacitor integrator, shown in Figure 5.2. This section explores the impact of non-idealities in the analog CS operator on the analog CS encoder performance.

5.4.1 Analog CS Operator Offset

The first non-ideality under consideration for the analog CS operator is the DC offset. The offset adds an unwanted DC component to the output compressed signal. The analog CS operator offset voltage will add error to the compressed signal in the same manner as the ADC offset voltage. So, if

the only error present in the system were the ADC's quantization noise and the CS analog operator, the SNR's behavior would be calculated by:

$$SNR = 20log\left(\frac{v_{in,rms}}{v_{q,rms} + v_{o,rms}}\right) \tag{5.16}$$

Again, the offset voltage is modeled using a stochastic approach. It is assumed that the analog CS operator offset is a random variable with normal distribution and zero mean. The probability distribution function of the offset voltage is therefore given by:

$$f_o(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-x^2/2\sigma^2} \tag{5.17}$$

Consequently, the RMS offset voltage is given by:

$$v_{o,rms} = \sqrt{\int_{-\infty}^{\infty} x^2 f_o(x)\,\mathrm{d}x} = \sqrt{\frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} x^2 e^{-x^2/2\sigma^2}\,\mathrm{d}x} \tag{5.18}$$

Figure 5.17 shows the effect of both the quantization noise and a randomly distributed analog CS operator offset voltage on the SNR for Patient #1. The offset voltage for each of the M CS analog operators was chosen from a normal random distribution with zero mean and the varying degrees of standard deviation. First, the theoretical SNR based on Equations 5.16 and 5.17 are presented. The RMS values for the compressed signal, $v_{in,rms}$, and quantization error, $v_{q,rms}$, are taken directly from Table 5.4. As expected, the SNR of the compressed signal degrades according to Equation 5.16. Further, a CS experiment was performed on one hour of ECG data for Patient #1 using the analog CS model presented in Section 5.1 and added quantization and offset non-idealities. The SNR of the experimental compressed signal is consistent with the theoretical results. The SNR of the reconstructed signal for C=2 at 96% sparsity is also shown. The reconstructed signal SNR follows the same trend as the compressed signal SNR but with a slight positive vertical translation. The value of the translation is largely dependent on the sparsity level. The reconstructed signal at this sparsity level shows an average of 3.95 dB better SNR performance for the first 5 mV of analog CS operator offset. The compressed signal SNR and reconstructed signal SNR eventually intersect at approximately 20 mV analog CS operator offset.

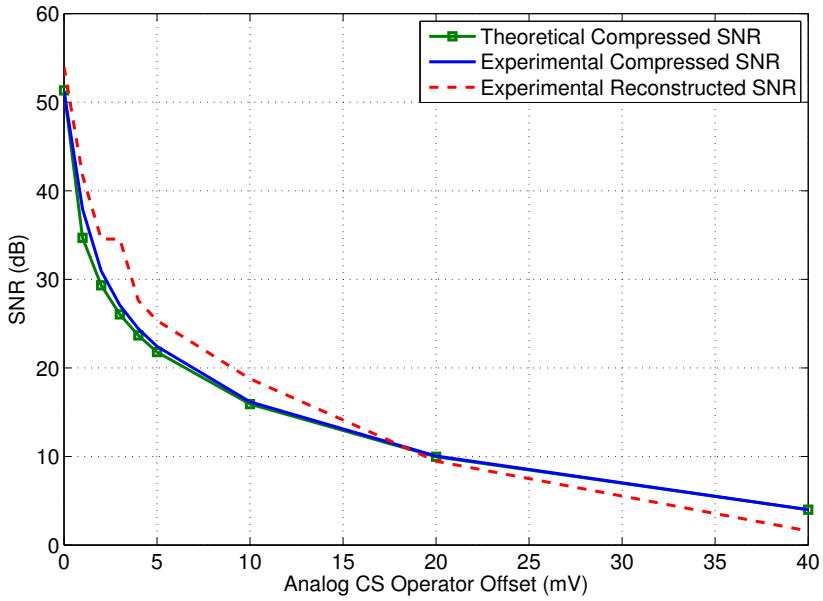


Figure 5.17: Analog CS System with Analog CS Operator Offset vs. SNR

5.4.2 Analog CS Operator Gain Error

Another potential issue is the analog CS operator gain error. For a digital to analog converter, the gain error is defined as the deviation from the slope of the DAC transfer curve assuming no offset or non-linearity error. Since the analog CS operator is essentially a one-bit (sign bit) MDAC and integrator, the gain error can be though of as an integrated MDAC gain error. The analog CS operator is characterized by the following difference equation:

$$v_{out}[n] = v_{out}[n-1] + \frac{C_1}{C_2}\frac{v_{diff}[n]}{2} \tag{5.19}$$

So, if $C_1 = 2C_2$ then the equation simplifies to:

$$v_{out}[n] = v_{out}[n-1] + \frac{C}{C}v_{diff} \tag{5.20}$$

Since the analog CS operator is one-bit, the ideal gain seen at the MDAC per iteration is simply $\frac{C}{C}$. Thus, the compressed output signal seen at each analog CS operator, y_i, is given by:

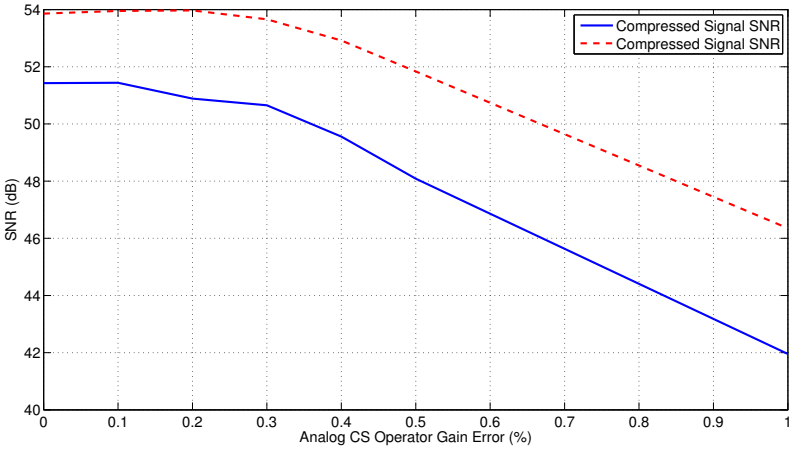


Figure 5.18: Analog CS System with Analog CS Operator Gain Error vs. SNR

$$y_i = \frac{C}{C}\sum_{j=1}^{N}\phi_{i,j}x_j = \sum_{j=1}^{N}\phi_{i,j}x_j \tag{5.21}$$

In this system, $\phi_{i,j} = \pm 1$. Unfortunately, the capacitors are not always perfectly matched, which leads to the analog CS operator gain error.

Figure 5.18 presents the analysis of the effects of analog CS operator gain error. The SNR performance is shown in terms of the standard deviation of the gain error. Both the SNR of the compressed signal and the reconstructed signal at 96% sparsity are shown. Again, the two curves follow the same trend. The main difference between the compressed signal SNR and the reconstructed signal SNR is a vertical shift in which the reconstructed signal consistently achieves better performance (dependent on signal sparsity). Gain error under 0.2% has minimal effect on performance. After the gain error exceeds 0.2%, the SNR steadily decreases, dropping approximately 7.5dB by 1% gain error. Fortunately, integrated circuit capacitors typically show matching within 0.1% [37]. Hence, the analog CS gain error should have negligible impact on this analog CS system design.

5.5 Analog CS Encoder System Design Conclusions

There are many design decisions regarding the analog CS encoder. This section assesses the impact of various typical issues on the analog CS encoder for ECG applications.

The first major discovery is the need for patient calibration. First, the frame size (N) should be calibrated per patient because it directly impacts the signal sparsity. Second, the pre-amplifier gain should be calibrated per patient for optimal results. In fact, pre-amplification calibration resulted in an overall 7dB performance improvement.

This section also shows that gain error in the analog CS encoder and ADC has minimal impact on system performance. Gain error in the ADC of 2 LSBs causes less than 1 dB SNR degradation. The gain error in the analog CS operator is dependent on the capacitor matching error. Capacitor matching error is usually within 0.1%, for which there is no impact on system performance.

Finally, this work shows that offset error in the analog CS encoder and ADC must be carefully considered. Offset error in the ADC of 1 LSB causes more than 10 dB SNR degradation. Offset error in the analog CS operator of 1 mV causes approximately 16 dB SNR degradation. To sum up, the results show that offset error has an extremely negative impact on analog CS encoder performance.

In general, this section shows that there are many failure modes for the analog CS encoder and tries to expose the most destructive errors to aid future designers. Designers should be very careful when considering error as some sources of error matter considerably more than others.

Chapter 6

CS DECODER AND RECONSTRUCTION ALGORITHMS

The purpose of the CS decoder of Figure 1.3(b) is to generate a reconstructed version, $\hat{\mathbf{x}}$, of the original bio-signal, $\mathbf{x}$, using the compressed signal, $\mathbf{y}$, the measurement matrix, $\mathbf{\Phi}$, and a sampling basis, $\mathbf{\Psi}$. As mentioned in Section 2.1.2, the CS signal recovery problem is an underdetermined system of linear equations. There are many possible solutions that satisfy Equation 2.3. However, only a few also satisfy the sparsity requirement for $\hat{\mathbf{x}}$. From among the possible sparse solutions, CS usually employs an optimization algorithm to select the sparsest solution as $\hat{\mathbf{x}}$, because it is known to exhibit the lowest error bound [38], [39]. This chapter guides the reconstruction algorithm selection [40]. Section 6.1 overviews the general approaches and theoretical performance limits of the different reconstruction algorithms. Section 6.2 presents experimental results that quantitatively compare the advantages and disadvantages of the algorithms for generic signals. Section 6.3 describes a case study on ECG signal reconstruction. Conclusions regarding the CS decoder for use in ECG applications are drawn in Section 6.4.

6.1 CS Reconstruction Algorithms Overview

Of the several CS recovery algorithms currently in use, most can be classified as variants of convex-optimization or greedy algorithms. Table 6.1 gives an overview of several state-of-the-art reconstruction algorithms including the general optimization objectives and the theoretical computation times. Initially convex optimization was chosen for its stability and high accuracy [6]. Convex optimization is the most widely accepted CS reconstruction and is thus the reconstruction method used in all prior experiments in this dissertation. Section 6.1.1 reviews convex optimization as it pertains to compressed sensing. Various greedy reconstruction algorithms have recently gained prominence by demonstrating a healthy trade-off of accuracy versus computational complexity. The popular CS greedy algorithms are presented in Section 6.1.2.

CS Reconstruction Algorithm	Algorithm Objective	Time Complexity
Basis Pursuit Convex Optimization [41]	Search the space for a solution with the minimum ℓ_1-norm	O($M^2N^1.5$)
Basis Pursuit Denoising (BPDN) Convex Optimization [42]	Search the space for a solution with minimum error and the minimum ℓ_1-norm together	O($M^2N^1.5$)
Orthogonal Matching Pursuit (OMP) [43]	Find the columns of $\mathbf{\Phi}$ with the maximum correlation to the residual	O(kMN)
Compressive Sampling Matching Pursuit (CoSaMP) [44]	Find the top 2K columns of $\mathbf{\Phi}$ with the maximum correlation to the residual	O($log(k)MN$)
Normalized Iterative Hard Thresholding (NIHT) [45]	Find the top k values of the sum of previous best guess and the signal proxy of the residual	O($log(k)MN$)

Table 6.1: Overview of CS Reconstruction Methods

6.1.1 CS Reconstruction Convex Optimization

The original CS reconstruction algorithm uses convex optimization wherein the ℓ_1-norm of the reconstructed signal, $\|\hat{\boldsymbol{\alpha}}\|_1 = \sum_i |\hat{\boldsymbol{\alpha}}_i|$, is an efficient measure of sparsity. The CS reconstruction process is described by the following ℓ_1-norm convex optimization:

$$\min \|\hat{\boldsymbol{\alpha}}\|_1 \text{ subject to } \mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\hat{\boldsymbol{\alpha}} \tag{6.1}$$

which can be cast as a linear programming problem. This approach is also known as Basis Pursuit (BP) [41].

In order for the signal reconstruction process to be robust in the presence of noise, the constraints of the optimization problem are relaxed:

$$\min \|\hat{\boldsymbol{\alpha}}\|_1 \text{ subject to } \|\mathbf{y} - \mathbf{\Phi}\mathbf{\Psi}\hat{\boldsymbol{\alpha}}\|_2 \leq \epsilon \tag{6.2}$$

The relaxed ℓ_1-norm convex optimization is known as Least Absolute Shrinkage and Selection

Operator (LASSO).

Finally, an alternative convex optimization approach to the basis pursuit and LASSO solutions is given as:

$$\min \frac{1}{2}\|\mathbf{y} - \mathbf{\Phi}\mathbf{\Psi}\hat{\boldsymbol{\alpha}}\|_2^2 + \lambda\|\hat{\boldsymbol{\alpha}}\|_1 \tag{6.3}$$

This solution is known as basis pursuit de-noising (BPDN) [42]. It finds a solution that simultaneously minimizes the noise level ($\mathbf{y} - \mathbf{\Phi}\mathbf{\Psi}\hat{\boldsymbol{\alpha}}$) and the sparsity. The parameter λ controls the balance between both optimization objectives.

The ℓ_1-norm convex optimization is the current standard approach to basis pursuit problems primarily due to its proven stability. The convex optimization problems are implemented in this work with the open-source software CVX [46]. CVX uses a primal-dual interior-point method to solve Equations 6.1, 6.2 and 6.3.

6.1.2 CS Reconstruction Greedy Algorithms

Unlike convex optimization, greedy algorithms solve the reconstruction problem in a less exact manner. They function by greedily optimizing a metric that predicts error minimization.

Matching pursuit, one class of CS reconstruction greedy algorithms, attempts to find the columns of the measurement matrix $\mathbf{\Phi}$ ($\mathbf{\Phi}$ is commonly referred to as an over-complete dictionary in this context) that have the most participation in the measurement $\mathbf{y}$. For each iteration, the column(s) of $\mathbf{\Phi}$ with the strongest correlation to the residual is added to the support vector and its contribution is subtracted from the residual. In orthogonal matching pursuit (OMP), only one column of $\mathbf{\Phi}$ (or atom) is added to the support vector per iteration [43], [47]. In CoSaMP (Compressed Sampling Matching Pursuit), a specialized adaption of matching pursuit for compressed sensing, the top 2k columns of $\mathbf{\Phi}$ are added to the support vector per iteration and later pruned [44].

Another popular reconstruction algorithm is the Normalized Iterative Hard Thresholding (NIHT) algorithm [45]. This method differs from the greedy algorithms described above by not directly searching for the columns of $\mathbf{\Phi}$ that reduce the residual error. Instead it operates by iteratively selecting solutions that both minimize the residual and maximize the difference between the solutions of the current and previous iterations.

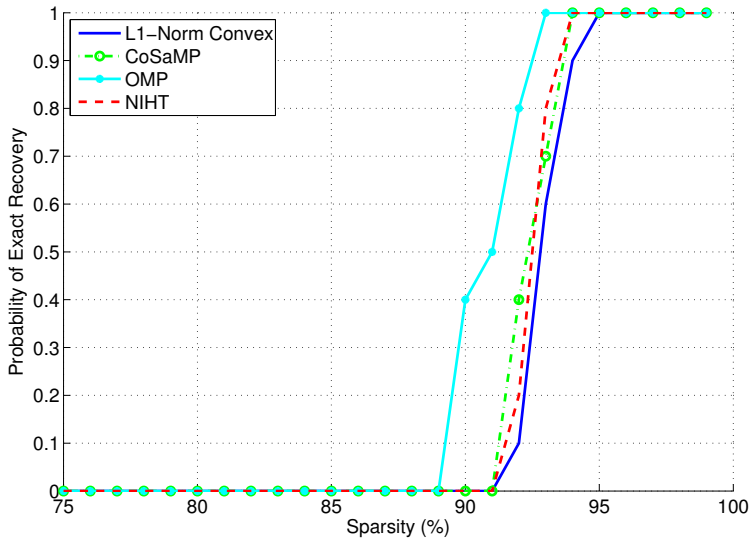


Figure 6.1: CS Reconstruction Computation Time Comparison

6.2 CS Reconstruction Algorithms Performance

The analysis of CS decoders begins with a general comparison of the selected CS reconstruction algorithms outlined in Section 6.1. In doing so, generic synthetic signals are analyzed in this section. 1024-point k-spike signals, $\mathbf{x_k}$, were synthesized by assigning randomly the distributed k elements to values from the i.i.d. Gaussian distribution with all other points set to zero. Sparsity levels were varied by adjusting k. Compression was performed using a 256×1024 measurement matrix $\mathbf{\Phi}$ populated with i.i.d. Gaussian random entries, achieving a C=4X data compression. Results were averaged over an ensemble of 100 k-spike sample signals for each sparsity level.

The most common performance metric among CS reconstruction literature is the reliability of signal recovery. In CS, it is the probability that the reconstruction algorithm will correctly recover the signal. This is of particular importance here because the reconstruction algorithms have a tendency to find local optima that do not necessarily represent the signal of interest. Figure 6.1 shows the probability of exact signal recovery, which increases with the sparsity of $\mathbf{x_k}$. All of the reconstruction curves show success above 88% sparsity. OMP produces the most reliable reconstruction

at low sparsity levels. Convex optimization, CoSaMP, and NIHT all exhibit moderate reliability.

6.3 CS Reconstruction ECG Bio-signal Case Study

The aim of this work is to assess the performance of the CS decoder on ECG bio-signals. This section takes a more realistic analysis approach by looking at metrics more appropriate to a bio-signal data acquisition system. More than 50 hours of ECG signals were collected from the Physiobank database [48]. Only ECG sparsity in time was observed, making the sparsifying matrix, $\mathbf{\Psi}$, the identity matrix. The ECG signal is windowed to 1024-point samples, and dynamically thresholded to control sparsity. To account for circuit noise, additive white Gaussian noise (AWGN) was added to the thresholded signal so that the compressed ECG signal achieved about 80 dB SNR. Compression was achieved using a 6-bit 256×1024 Gaussian $\mathbf{\Phi}$ matrix (i.e., C = 4). Reconstruction was performed in MATLAB.

Figure 6.2 compares the accuracy of several CS reconstruction methods for ECG signals. All reconstruction algorithms except BPDN meet the target SNR level (80 dB) at high sparsity levels. OMP gives higher accuracy at lower sparsity levels but its reliability is of concern because its performance suddenly dips at high sparsity levels. CoSaMP shows high accuracy at lower sparsity levels with better reliability. CoSaMP, NIHT and BP convex optimization all show moderate accuracy performance results, while BPDN convex optimization does not give promising accuracy results. However, future work including BPDN convex optimization should include a more thorough investigation of the optimum sparsity/error balance parameter (λ).

Computation time is also an important factor in a CS decoder because it has a direct impact on energy efficiency and real-time application feasibility. In fact, it is the primary objective for exploring greedy solutions. The greedy reconstruction algorithms were implemented exactly as outlined by their respective authors. Further, great care was taken to choose efficient (and more importantly, consistent) subroutines, especially the pseudoinverse calculation. The computation times of these algorithms, as reported by Matlab run on a 2.16 GHz Intel Core 2 Duo processor, are compared in Figure 6.3. The computation time is measured as the average computation time required to reconstruct one signal frame for a specified sparsity level. While the absolute computation time is dependent on the hardware utilized, this comparison allows for a qualitative comparison of the

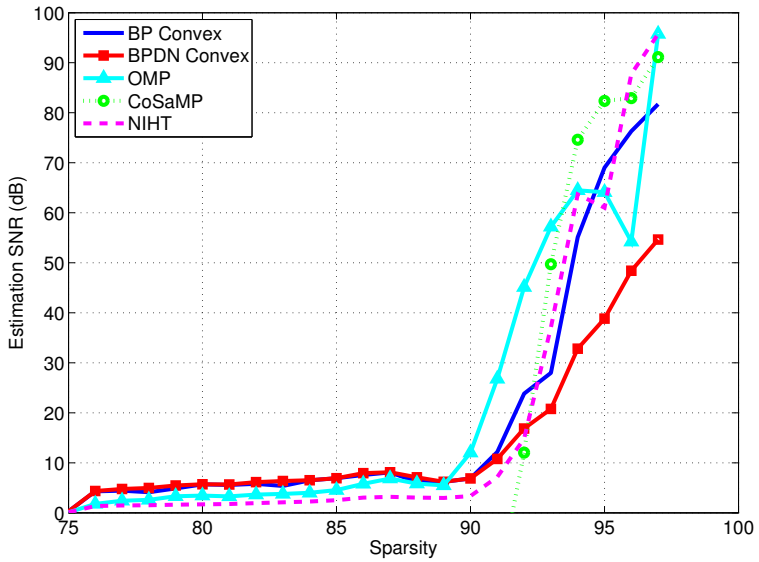


Figure 6.2: CS Reconstruction ECG Signal Accuracy Comparison

different reconstruction options. OMP requires the least computation time. BP convex optimization has higher computational complexity than most greedy solutions but is notably independent of sparsity. CoSaMP shows the largest computation time of the greedy solutions, but improves at higher sparsity levels ($> 90\%$) where the CS reconstruction algorithms show consistent recovery (Figure 6.2).

6.4 CS Decoder Conclusions

This chapter presented a detailed empirical comparative study of selected state-of-the-art compressed sensing reconstruction algorithms. Table 6.2 summarizes in qualitative terms the overall relative merits and demerits of these algorithms. A case study of CS operating on 80 dB SNR ECG bio-signals to show consistency with the empirical comparisons is presented.

If computation time is not of primary concern, CoSaMP and L1-norm convex are likely the best choice for most applications where accuracy is needed; further CoSaMP outperforms L1-norm convex in noise resilience. However, for systems where computational complexity is of concern,

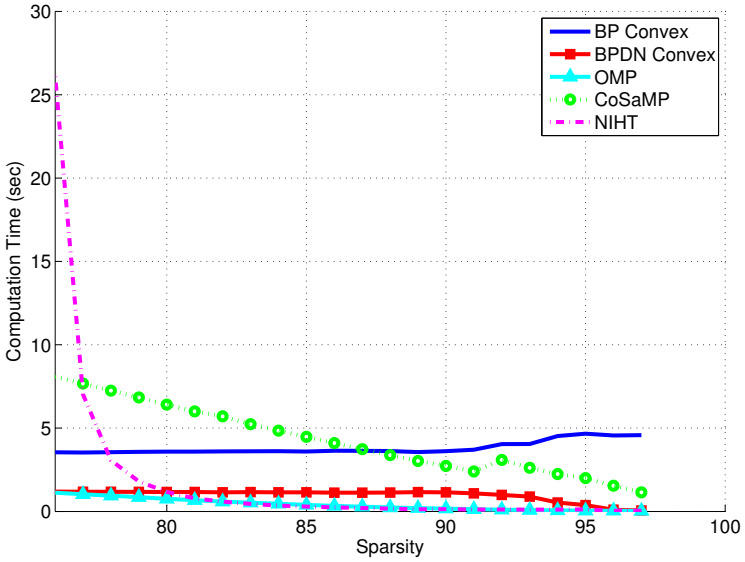


Figure 6.3: CS Reconstruction Computation Time Comparison

Reconstruction Algorithm	Reliability/Accuracy	Speed
BP Convex	Fair	Fair
BPDN Convex	Bad	Good
OMP	Good	Good
CoSaMP	Good	Fair
NIHT	Fair	Good

Table 6.2: ECG CS System Model Compressed Signal Voltage and Quantization Error Voltage for Fixed Gain

such as an ASIC implementation with low power consumption, OMP is preferable. Although no one best compressed sensing reconstruction algorithm can be assured for all applications, the proper system specifications would lead to optimum results based on the comparative study presented.

Chapter 7

CONCLUSIONS AND FUTURE WORK

A body area network is a network of wearable and wireless physiological sensors. BANs would have an enormous impact on more persistent and personalized healthcare and is consequently a growing field of research. One of the most important ways to improve BANs is through decreased wireless sensor power consumption. Compressed sensing is a compression scheme capable of significantly reducing a signal acquisition's data rate. Consequently, compressed sensing is an effective tool to reduce a wireless physiological sensor's power consumption. Since compressed sensing is still an emerging topic, only a handful of CS systems have been realized in hardware. These systems have shown promising and yet limited abilities. This dissertation provides the framework to more efficiently design CS systems targeting ECG applications (Figure 1.4). It accomplishes this task by addressing the following issues:

- **Sampling Basis** In order for compressed sensing to be successful, the target signal must be sparse. Fortunately, the sparsity restriction can apply in any reversible basis. This work addresses the design decisions regarding the CS sparsity requirement and the selection of an appropriate sampling basis for the ECG application. First, this work shows that a system without thresholding can achieve a C=1.5 compression ratio using the wavelet sampling basis. However, CS systems with thresholding are often considered due to their ability to achieve much higher compression ratios. This work shows that when an ECG signal is thresholded in its' sampling basis (for example, thresholded in the wavelet domain for a wavelet basis sampling domain), the wavelet domain exhibited much higher sparsity than the time domain. However, thresholding in the time domain is more practical for CS encoder hardware simplicity. So for thresholding exclusively in the time domain, there was little difference between the sampling bases but the time sampling basis performed the best.

- **Random Sensing Matrix Generation** Compressed sensing systems operate by multiplying the signal by a random sensing matrix. CS dictates that the sensing matrix be incoherent with the sampling basis matrix. The coherence between several probability distribution functions and the time and wavelet domains were computed. Interestingly, the Bernoulli distribution (a one-bit implementation) yielded the lowest coherence with both the time and wavelet domains. The one-bit sensing matrix simplifies the analog CS encoder hardware substantially. Pseudorandom number generators are efficient sensing matrix generators and this work is the first to explore their effectiveness. This work shows that there are two important metrics for sensing matrix generator design: (i) incoherence between the sensing matrix and sampling matrix and (ii) the sensing matrix mutual incoherence. For the analog CS encoder used in ECG applications, this work endorses the use of a Bernoulli sensing matrix developed by a simple Fibonacci or Galois LFSR.

- **Analog System Imperfections** Compressed sensing is a powerful mathematical concept. However, the analog encoder's performance in hardware is highly susceptible to system non-idealities. Analysis shows the effects of various analog CS attributes and imperfections. The first important discovery is that performance is greatly improved by per-patient calibration of frame size and pre-amplification. This work then shows the impact of various system errors typically present in the analog CS operator and ADC on analog CS encoder performance. Results show that gain error in the analog CS operator and ADC have minimal effect on the system. However, offset error in the analog CS operator and ADC have a significant negative impact of performance. Thus, this work advises CS encoder designers to be careful when considering system errors as some matter considerably more than others.

- **CS Reconstruction** Several CS signal reconstruction algorithms exist in CS literature. Convex optimization and greedy algorithms alike are viable solutions to recover CS compressed signals. A thorough comparison of the state-of-the-art CS reconstruction algorithms was given, comparing metrics such as reliability, accuracy and noise resilience and computation time. The conclusions of this study are that the reconstruction techniques are optimal for different design criteria. This dissertation promotes the selection of certain algorithms based on

the driving goal of the decoder (accuracy, computation time, etc.).

Potential future work is plentiful. The following lists composes the most important future work points:

- **Thresholding Techniques** This work very demonstrates the effectiveness of the wavelets domain as a sparsifying basis for ECG signals. However, this work further stated that this was only really beneficial for HALT systems. This work employed an absolute thresholding technique (for hardware simplicity), for which there was not a tremendous signal sparsity benefit. Since signal sparsity is the key to success of CS systems, future work in this area includes a more rigorous search for a better thresholding technique and it's hardware implementation.

- **Analog CS System Imperfections: ADC Nonlinearity** This work modeled the absolute accuracy of a binary-weighted charge scaling DAC (used inside a SAR ADC) in terms of capacitor tolerance. Unfortunately, absolute accuracy is dependent on both gain error and nonlinearity. Future work in this area includes further analysis to determine the effects of nonlinearity itself. Further, it would be interesting to see the impact of absolute error and nonlinearity for other ADC architectures, such as a C-2C SAR ADC architecture.

- **Decoder Hardware** This work compared and contrasted several CS reconstruction algorithms and deduced there are optimal choices based on the primary objective of the decoder. This work assumes since the decoder presides at a personal base station that hardware complexity is of less concern. However, a true optimization of decoder solutions would take this into account. Future work in this area includes a study on the hardware complexity of the decoder solutions.

BIBLIOGRAPHY

[1] D. Gangopadhyay, "Compressed sensing analog-to-digital converters for biomedical applications," Ph.D. dissertation, University of Washington, 2011.

[2] P. Allen and D. Holberg, *CMOS Analog Circuit Design*, ser. The Oxford Series in Electrical and Computer Engineering. Oxford University Press, USA, 2002.

[3] "World health statistics 2011," World Health Organization, Tech. Rep., 2011.

[4] C. O. E. Jovanov, A. Milenkovic and P. de Groen, "A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation," *J. of NeuroEngineering and Rehabilitation*, vol. 2, pp. 1–10, 2005.

[5] J. Pandey and B. Otis, "A sub-100 w mics/ism band transmitter based on injection-locking and frequency multiplication," *Solid-State Circuits, IEEE Journal of*, vol. 46, no. 5, pp. 1049 –1058, may 2011.

[6] E. Candes and M. Wakin, "An introduction to compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21 –30, march 2008.

[7] A. Dixon, E. Allstot, D. Gangopadhyay, and D. Allstot, "Compressed sensing system considerations for ecg and emg wireless biosensors," *Biomedical Circuits and Systems, IEEE Transactions on*, vol. 6, no. 2, pp. 156 –166, april 2012.

[8] D. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289 –1306, april 2006.

[9] R. Yazicioglu, S. Kim, T. Torfs, H. Kim, and C. Van Hoof, "A 30 w analog signal processor asic for portable biopotential signal monitoring," *Solid-State Circuits, IEEE Journal of*, vol. 46, no. 1, pp. 209 –223, jan. 2011.

[10] R. Rieger and J. Taylor, "An adaptive sampling system for sensor nodes in body area networks," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 17, no. 2, pp. 183 –189, april 2009.

[11] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. Baraniuk, "Single-pixel imaging via compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 83 –91, march 2008.

[12] J. Laska, S. Kirolos, M. Duarte, T. Ragheb, R. Baraniuk, and Y. Massoud, "Theory and implementation of an analog-to-information converter using random demodulation," in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, may 2007, pp. 1959 –1962.

[13] T. Ragheb, J. Laska, H. Nejati, S. Kirolos, R. Baraniuk, and Y. Massoud, "A prototype hardware for random demodulation based compressive analog-to-digital conversion," in *Circuits and Systems, 2008. MWSCAS 2008. 51st Midwest Symposium on*, aug. 2008, pp. 37 –40.

[14] M. Mishali, Y. Eldar, O. Dounaevsky, and E. Shoshan, "Xampling: Analog to digital at sub-nyquist rates," *Circuits, Devices Systems, IET*, vol. 5, no. 1, pp. 8 –20, january 2011.

[15] M. Mishali and Y. Eldar, "From theory to practice: Sub-nyquist sampling of sparse wideband analog signals," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 375 –391, april 2010.

[16] M. Lexa, M. Davies, and J. Thompson, "Reconciling compressive sampling systems for spectrally sparse continuous-time signals," *Signal Processing, IEEE Transactions on*, vol. 60, no. 1, pp. 155 –171, jan. 2012.

[17] F. Chen, A. Chandrakasan, and V. Stojanović and, "A signal-agnostic compressed sensing acquisition system for wireless and implantable sensors," in *Custom Integrated Circuits Conference (CICC), 2010 IEEE*, sept. 2010, pp. 1 –4.

[18] F. Chen, A. Chandrakasan, and V. Stojanovic, "Design and analysis of a hardware-efficient compressed sensing architecture for data compression in wireless sensors," *Solid-State Circuits, IEEE Journal of*, vol. 47, no. 3, pp. 744 –756, march 2012.

[19] S. Jalaleddine, C. Hutchens, R. Strattan, and W. Coberly, "Ecg data compression techniques-a unified approach," *Biomedical Engineering, IEEE Transactions on*, vol. 37, no. 4, pp. 329 –343, april 1990.

[20] *EC13: Cardiac monitors, heart rate meters and alarms*, Association for the Advancement of Medical Instrumentation, 2002.

[21] *EC11: Diagnostice Cardiographic Electronic Devices*, Association for the Advancement of Medical Instrumentation, 1991.

[22] C. Li, C. Zheng, and C. Tai, "Detection of ecg characteristic points using wavelet transforms," *Biomedical Engineering, IEEE Transactions on*, vol. 42, no. 1, pp. 21–28, 1995.

[23] J. Sahambi, S. Tandon, and R. Bhatt, "Using wavelet transforms for ecg characterization. an on-line digital signal processing system," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 16, no. 1, pp. 77–83, 1997.

[24] M. Kania, M. Fereniec, and R. Maniewski, "Wavelet denoising for multi-lead high resolution ecg signals," *Measurement Science Review*, vol. 7, pp. 30–33, 2007.

[25] R. Benzid, F. Marir, A. Boussaad, M. Benyoucef, and D. Arar, "Fixed percentage of wavelet coefficients to be zeroed for ecg compression," *Electronics Letters*, vol. 39, no. 11, pp. 830–831, 2003.

[26] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 11, no. 7, pp. 674–693, 1989.

[27] R. Edwards and C. Cauwenberghs, "A vlsi implementation of the continuous wavelet transform," in *Circuits and Systems, 1996. ISCAS '96., Connecting the World., 1996 IEEE International Symposium on*, vol. 4, may 1996, pp. 368 –371 vol.4.

[28] C.-T. Huang, P.-C. Tseng, and L.-G. Chen, "Analysis and vlsi architecture for 1-d and 2-d discrete wavelet transform," *Signal Processing, IEEE Transactions on*, vol. 53, no. 4, pp. 1575 – 1586, april 2005.

[29] R. Edwards and M. Godfrey, "An analog wavelet transform chip," in *Neural Networks, 1993., IEEE International Conference on*, 1993, pp. 1247 –1251 vol.3.

[30] E. Allstot, A. Chen, A. Dixon, D. Gangopadhyay, and D. Allstot, "Compressive sampling of ecg bio-signals: Quantization noise and sparsity considerations," in *Biomedical Circuits and Systems Conference (BioCAS), 2010 IEEE*, Nov. 2010, pp. 41 –44.

[31] A. Salman, E. Allstot, A. Chen, A. Dixon, D. Gangopadhyay, and D. Allstot, "Compressive sampling of emg bio-signals," in *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on*, May 2011, pp. 2095 –2098.

[32] E. Allstot, A. Chen, A. Dixon, D. Gangopadhyay, H. Mitsuda, and D. Allstot, "Compressed sensing of ecg bio-signals using one-bit measurement matrices," in *New Circuits and Systems Conference (NEWCAS), 2011 IEEE 9th International*, june 2011, pp. 213 –216.

[33] P. Alfke, "Efficient shift registers, lfsr counters and long pseudo-random sequence generators," Xilinx, Tech. Rep., July 1996.

[34] D. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *Information Theory, IEEE Transactions on*, vol. 47, no. 7, pp. 2845 –2862, nov 2001.

[35] E. Candes and J. Romberg, "Sparsity and incoherence in compressive sampling," 2006.

[36] M. Integrated. (2002, July) The abcs of adcs: Understanding how adc errors affect system performance. [Online]. Available: http://www.maximintegrated.com/app-notes/index.mvp/id/748

[37] D. A. Johns and K. Martin, *Analog Integrated Circuit Design*. John Wiley and Sons, Inc, 1997.

[38] E. Candes, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, pp. 1207–1223, August 2006.

[39] E. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *Information Theory, IEEE Transactions on*, vol. 52, no. 12, pp. 5406 –5425, dec. 2006.

[40] A. Dixon, E. Allstot, A. Chen, D. Gangopadhyay, and D. Allstot, "Compressed sensing reconstruction: Comparative study with applications to ecg bio-signals," in *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on*, may 2011, pp. 805 –808.

[41] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Computing*, vol. 20, pp. 33–61, January 1999.

[42] W. Lu and N. Vaswani, "Modified basis pursuit denoiing (modified-bpdn) for noisy compressive sensing with partially known support," *IEEE Int. Conf. on Acoustics Speech and Signal Processing*, pp. 3926–3929, 2010.

[43] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *Information Theory, IEEE Transactions on*, vol. 53, no. 12, pp. 4655 –4666, dec. 2007.

[44] D. Needell and J. A. Tropp, "Cosamp: iterative signal recovery from incomplete and inaccurate samples," *Commun. ACM*, vol. 53, no. 12, pp. 93–100, Dec. 2010. [Online]. Available: http://doi.acm.org/10.1145/1859204.1859229

[45] T. Blumensath and M. Davies, "Normalized iterative hard thresholding: Guaranteed stability and performance," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 298 –309, april 2010.

[46] M. Grant. (2010, August) Cvx: Matlab software for disciplined convex programming. [Online]. Available: http://cvxr.com/cvx

[47] T. Blumensath and M. Davies, "On the difference between orthogonal matching pursuit and orthogonal least squares," University of Edinburgh, Tech. Rep., 2007.

[48] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic

signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000 (June 13), circulation Electronic Pages: http://circ.ahajournals.org/cgi/content/full/101/23/e215 PMID:1085218; doi: 10.1161/01.CIR.101.23.e215.

Appendix A

ANALOG CS ROW OPERATOR OPERATION

The analog CS row operator has three modes of operation: @reset = 1, @multiply = 1 and @sum =1

@reset = 1: This mode is enabled at the beginning of each new CS matrix row operation frame. It resets values to prepare to calculate the next $\mathbf{y}$ value.

To perform the CS operation, the encoder to go through N subsequent @multiply = 1 and @sum = 1 cycles. The analog CS row operator in Figure A.1 produces one of the M total $\mathbf{y}$ outputs after the N cycles. The operations performed during this cycle is described by the following operation.

@multiply = 1: The multiplication mode conducts the multiplication of the current $\mathbf{x}$ value and the corresponding $\phi[n]$. Figure A.2 provides a simplified view of the analog CS operator in this mode. It is assumed that $\phi[n] = 1$ here.

The sum of the charges at this phase is as follows

$$\sum Q(multiply) = C_1(V_p[n] - V_{CM}[n]) + C_2(V_{OUT}[n-1] - V_{CM}[n]) \tag{A.1}$$

@sum = 1: The sum mode simply adds the result of the multiplication mode to the previous sum. Figure A.3 provides a simplified view of the analog CS operator in this mode.

The sum of the charges at this phase is as follows

$$\sum Q(sum) = C_1(V_{CM}[n] - V_{CM}[n]) + C_2(V_{OUT}[n] - V_{CM}[n]) \tag{A.2}$$

Now, using the conservation of charges rule, the sum of the charges at the multiply phase equals the sum of the charges at the sum phase:

$$\sum Q(multiply) = \sum Q(sum) \tag{A.3}$$

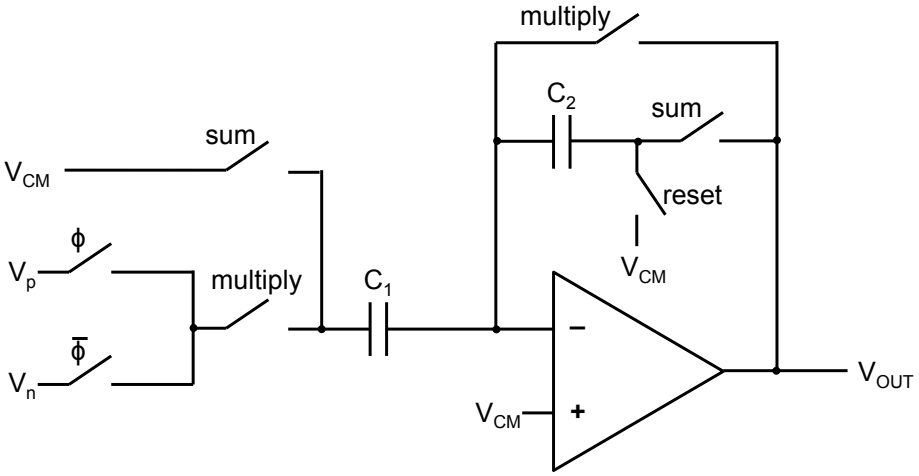


Figure A.1: One-Bit Analog CS Row Operator

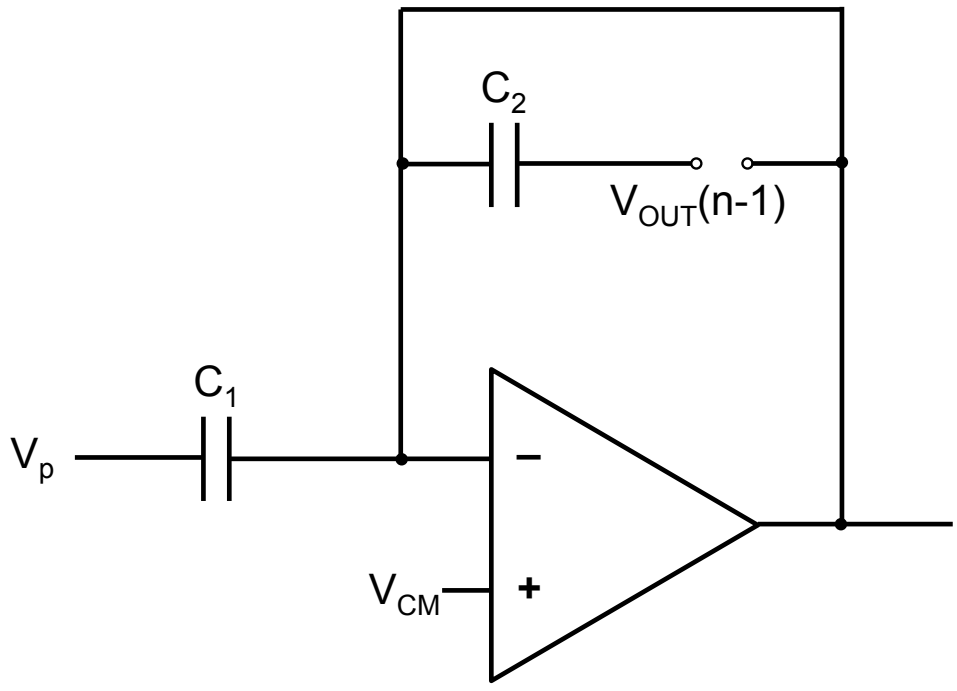


Figure A.2: One-Bit Analog CS Row Operator Multiplication Mode

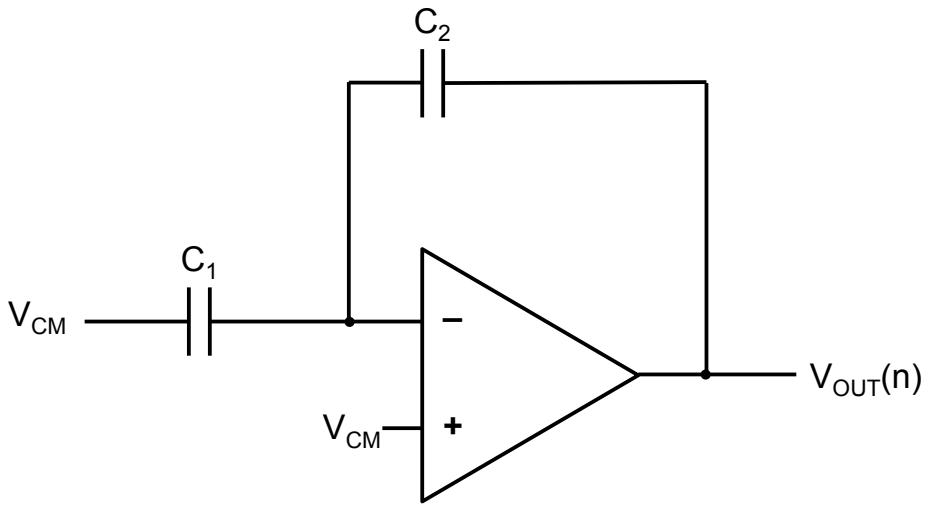


Figure A.3: One-Bit Analog CS Row Operator Multiplication Mode

$$C_1(V_p[n]-V_{CM}[n])+C_2(V_{OUT}[n-1]-V_{CM}[n]) = C_1(V_{CM}[n]-V_{CM}[n])+C_2(V_{OUT}[n]-V_{CM}[n]) \tag{A.4}$$

Rewriting Equation A.4 gives

$$V_{OUT}[n] - V_{OUT}[n-1] = \frac{C_1}{C_2}(V_p[n] - V_{CM}[n]) \tag{A.5}$$

By substituting $V_{CM} = \frac{V_p[n]-V_n[n]}{2}$, Equation A.5 is rewritten as

$$V_{OUT}[n] - V_{OUT}[n-1] = \frac{C_1}{C_2}\left(V_p[n] - \frac{V_p[n] - V_n[n]}{2}\right) \tag{A.6}$$

$$V_{OUT}[n] = V_{OUT}[n-1] + \frac{C_1}{C_2}\left(\frac{V_p[n] - V_n[n]}{2}\right) \tag{A.7}$$

By substituting $V_{diff}[n] = \frac{V_p[n]-V_n[n]}{2}$, the output of the analog CS row operator is expressed as

$$V_{OUT}[n] = V_{OUT}[n-1] + \frac{C_1}{C_2}V_{diff}[n] \tag{A.8}$$

Appendix B

BINARY-WEIGHTED CHARGE SCALING DAC ABSOLUTE ACCURACY

This Appendix shows the derivation of the absolute accuracy/error for the binary-weighted charge scaling DAC (Figure B.1). The derivation is done in close conjunction with that presented in [2].

In this architecture, the output voltage is defined as:

$$v_{OUT} = \frac{\sum C_{on}}{\sum C_{on} + \sum C_{off}} V_{REF} \tag{B.1}$$

where $\sum C_{on} = \sum_{i=0}^{N-1} b_i 2^{-i} C$ and $\sum C_{off} = \sum_{i=0}^{N-1} \bar{b_i} 2^{-i} C$. It is assumed throughout this analysis that $i = 0$ represents the most significant bit and $i = N - 1$ represents the least significant bit. Further, b_i is a binary value representing the i-th bit. So Equation B.1 is rewritten as:

$$v_{OUT} = \frac{\sum_{i=0}^{N-1} b_i 2^{-i} C}{\sum_{i=0}^{N-1} b_i 2^{-i} C + \sum_{i=0}^{N-1} \bar{b_i} 2^{-i} C} V_{REF} \tag{B.2}$$

Equation B.3 represents the ideal output voltage. In reality, the capacitors are not ideal and have an error ΔC. The actual output voltage can be rewritten as:

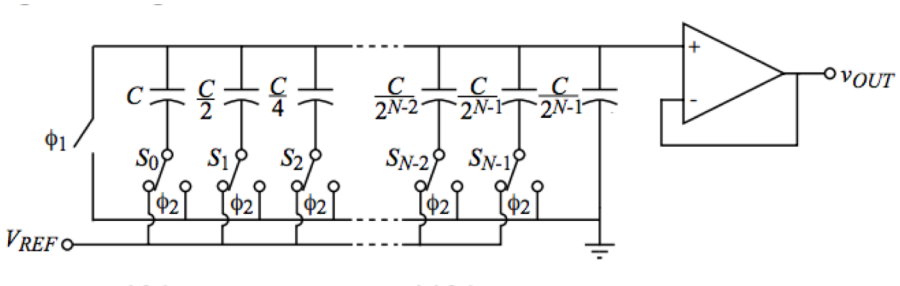


Figure B.1: Binary-Weighted Charge Scaling DAC [2]

$$v_{OUT} = \frac{\sum_{i=0}^{N-1} b_i 2^{-i} C + \Delta C_i}{\sum_{i=0}^{N-1} b_i 2^{-i} C + \Delta C_i + \sum_{i=0}^{N-1} \bar{b_i} 2^{-i} C + \Delta C_i} V_{REF} \tag{B.3}$$

The absolute error at digital input B, is the difference between the ideal and actual output voltages at that input:

$$\epsilon(B) = \frac{\sum_{i=0}^{N-1} b_i 2^{-i} \Delta C_i}{2C + \sum_{i=0}^{N-1} b_i 2^{-i} C + \Delta C_i + \sum_{i=0}^{N-1} \bar{b_i} 2^{-i} C + \Delta C_i} V_{REF} \tag{B.4}$$

Equation B.4 can be rewritten as

$$\epsilon = \frac{\sum_{i=0}^{N-1} b_i 2^{-i-1} \frac{\Delta C_i}{C}}{1 + \sum_{i=0}^{N-1} b_i 2^{-i-1} \frac{\Delta C_i}{C} + \sum_{i=0}^{N-1} \bar{b_i} 2^{-i-1} \frac{\Delta C_i}{C}} V_{REF} \tag{B.5}$$

Now the equation is written in terms of capacitor tolerance of each capacitor, $\frac{\Delta C_i}{C}$. If you assume that $\sum_{i=0}^{N-1} b_i 2^{-i-1} \frac{\Delta C_i}{C} + \sum_{i=0}^{N-1} \bar{b_i} 2^{-i-1} \frac{\Delta C_i}{C} << 1$, then Equation B.5 becomes:

$$\epsilon(B) = \sum_{i=0}^{N-1} b_i 2^{-i-1} \frac{\Delta C_i}{C} V_{REF} \tag{B.6}$$

Absolute error can be rewritten in terms of LSBs:

$$\epsilon(B) = \sum_{i=0}^{N-1} b_i 2^{-i-1} \frac{\Delta C_i}{C} V_{REF} \frac{2^N}{2^N} = \sum_{i=0}^{N-1} b_i 2^{N-i-1} \frac{\Delta C_i}{C} \tag{B.7}$$