A Compressive Sampling Architecture For Wideband Communication

by

CHANDRA PRAKASH
201021004

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY
to

DHIRUBHAI AMBANI INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY

January, 2020

Declaration

I hereby declare that

i) the thesis comprises of my original work towards the degree of Doctor of Philosophy at Dhirubhai Ambani Institute of Information and Communication Technology and has not been submitted elsewhere for a degree.

ii) due acknowledgment has been made in the text to all the used reference materials.

Chandra Prakash

Certificate

This is to certify that the thesis work entitled A COMPRESSIVE SAMPLING ARCHITECTURE FOR WIDEBAND COMMUNICATION has been carried out by CHANDRA PRAKASH for the degree of Doctor of Philosophy at *Dhirubhai Ambani Institute of Information and Communication Technology* under my supervision.

Prof Yash Vasavada
Thesis Supervisor

Acknowledgments

I would like to express my sincere gratitude to my research supervisor Prof. Yash Vasavada for his constant guidance and encouragement. His continuous pursuit for excellence has been the utmost inspiration towards shaping up my goals and targets. He has been a constant source of inspiration and is a true teacher for my life.

I wish to express my gratitude to my parents and family members who have always encouraged and motivated me to realize my goal. A special mention to my father in law Shri D G Palnitkar who has always shared my vision of "Keep Learning". Special thanks to my wife Shilpa, son Krish, and little daughter Shreya for their unconditional love and support.

I would like to acknowledge the support received from Shri Abhinav Anand and friends for their technical review and suggestions about my work.

I would also like to express my sincere thanks to Shri D K Das, Director-Space Applications Centre, ISRO, Ahmedabad and Dr. K S Dasgupta, Director-DAIICT, Gandhinagar for providing me conducive environment and opportunity to work towards my research goals.

Contents

Abstract vii

List of Principal Symbols and Acronyms xi

List of Tables xii

List of Figures xiii

1 Introduction 1

1.1 Problem Statement 3

1.2 Theoretical Background of Compressive Sampling 4

1.2.1 Properties of a Sensing Matrix 5

1.2.2 Recovery from Compressed Measurements 8

1.2.3 Signal Measurement Models in CS 11

1.2.4 Realisable CS Architectures 13

1.3 Research Contribution 13

1.4 Overview of the Dissertation 14

2 A Review of CS Architectures and Spectrum Sensing Techniques 16

2.1 Hardware Architectures for Compressive Sampling 16

2.1.1 Multicoset Sampling 17

2.1.2 Random Demodulator 19

2.1.3 Modulated Wideband Converter 21

2.2 Spectrum Sensing Techniques 23

2.2.1 Auto Correlation based Detection 25

2.2.2 Euclidean Distance based Detection 26

2.2.3 Wavelet based Detection 26
2.2.4 Matched Filter based Detection 27
2.2.5 Energy Estimator based Detection 27

3 A Detailed Review of MWC Architecture 29
3.1 Signal Reconstruction in MWC 30
3.2 Modified MWC Architecture 31
3.3 MWC Receiver for Jamming Detection and Filtering 34
3.4 Noise foldover and its characterization in CS 38

4 Proposed CS Architecture and System Model 42
4.1 Signal Model 42
4.2 Proposed CS Architecture 43
4.2.1 Design of Analog Mixing Waveforms 46
4.2.2 Model of Discrete-Time Measurements 48
4.2.3 Sensing Matrix Construction 53
4.3 Theoretical Bounds on the Size of Measurement Matrix 53
4.3.1 Derivation of Lower Bounds: Coding Theory Approach . . 54
4.3.2 Derivation of Lower Bounds : An Information Theoretic Approach 56
4.3.3 Type of Applications and their Lower Bounds 57
4.4 Algorithms for Blind Support Detection 59
4.4.1 Algorithm A: Based on Gallager LDPC Decoding Technique 59
4.4.2 Algorithm B : Based on Sudocodes 62
4.4.3 Performance comparison 62
4.5 Signal Reconstruction 63
4.6 Noise Performance of Proposed Architecture 65

5 Belief Propagation based Sparsity Detection Algorithms 71
5.1 Proposed Sparsity Detection Algorithms 73
5.1.1 Message Passing Algorithm with Independent Probability Estimate (IPE) 73

5.1.2 Message Passing Algorithm with Joint Probability Estimate (JPE) . . . 77
5.2 Simulation based Performance Verification . . . 81
5.2.1 Characterization of Measurement Matrix . . . 81
5.2.2 Simulation Results for IPE and JPE Algorithms . . . 82
5.3 Convergence Analysis using Density Evolution and Extrinsic Information Transfer (EXIT) Chart . . . 91
5.4 Computational Complexity of BP Algorithms . . . 94
5.5 Extension of Sparsity Detection Algorithms . . . 95
5.5.1 Algorithm for Discrete Non-binary Input . . . 95
5.5.2 Deterministic Algorithm for Noiseless Case . . . 99

6 Low Density Integer Constrained (LDIC) Codes 103
6.1. The Concept of LDIC Codes . . . 103
6.1.1 Source Encoding . . . 104
6.1.2 Source Decoding . . . 109

7 Summary and Conclusion 111
7.1 Work Summary and Conclusion . . . 111
7.2 Future Work . . . 114

References 115

Appendix A CS Based Scalable Video Codec (SVC) 128

Abstract

This dissertation proposes a novel Compressive Sampling (CS) scheme for Sub-Nyquist Spectrum Sensing (SNSS) of spectrally sparse wideband signals. A novelty of our proposed SNSS scheme resides in the analog front-end. We show that it can be modeled as a sparse binary-valued measurement matrix. This has allowed us to bring to bear the proven advantages of the Low Density Parity Check (LDPC) matrices in improving the performance of the existing SNSS methods. Specifically, we show that the number of parallel SNSS channels required for a robust CS sparsity detection in our proposal is reduced compared to the existing SNSS methods. We provide new analytic (information-theoretic) lower bounds on this number and show that the LDPC-based measurement matrix is closer to this bound compared to the alternatives.

The existing algorithms (such as those based on Matching Pursuit or Basis Pursuit) for CS sparsity detection are not optimal for our proposed architecture given the unique (sparse binary-valued) aspect of the measurement matrix. We develop two new Belief Propagation (BP) algorithms - an Independent Probability Estimates (IPE) algorithm and a Joint Probability Estimates (JPE) algorithm - to solve the sparsity detection problem. The performance of these algorithms is evaluated using Monte-Carlo simulations as well as semi-analytic approaches based on Density Evolution and EXIT (Extrinsic Information Transfer) methods. We show that the proposed algorithms outperform several existing algorithms (including the well-known Orthogonal Matching Pursuit (OMP) algorithm).

Another contribution of our work is in mitigating the problem of noise enhancement (during Zero-Forcing based signal reconstruction) that affects several existing SNSS schemes (such as the Modulated Wideband Converter (MWC)). We

provide analytical proofs showing this benefit and confirm the analytical results by simulation.

Finally, we demonstrate the signal reconstruction in the proposed CS receiver through simulation. The Bit Error Rate (BER) performance of a QPSK system with the proposed CS receiver is simulated and the performance improvement over the MWC is demonstrated. As an extension of the developed algorithms, a framework of joint compression and denoising application is envisioned and presented with theoretical analysis.

List of Principal Symbols and Acronyms

List of Principal Symbols

$\boldsymbol{A}$	Sensing matrix
$\boldsymbol{x}$	Discrete time signal
$\boldsymbol{y}$	Measurement vector
diag $\{.\}$	Diagonal matrix
$\frac{L}{N}$	Rate or Compression ratio of sensing matrix
σ	Variance
$\sigma(A)$	Spark of matrix A
$\mathbf{R}$	Correlation matrix
$A^{\dagger}$	Pseudo-inverse of matrix A
A^{T}	Transpose of matrix A
C_m	Set of indices of variable nodes connected to m^{th} check node
F	Fourier matrix
f_{NYQ}	Nyquist sampling frequency
$H(.)$	Entropy function
i_c	Iteration count
k	Sparsity

m Number of channels in sensing architecture

$p_l(t)$ Analog mixing waveform in l^{th} branch

Q Covariance matrix

$q_{n\to m}$ Message sent by n^{th} variable node b_n to the check node c_m

$u_{m\to n}$ Message sent by the check node c_m to the variable node b_n

V_n Set of indices of check nodes connected to n^{th} variable node

w_c Number of ones in each column of sensing matrix

w_r Number of ones in each row of sensing matrix

List of Acronyms

ADC Analog to Digital Converter

AMP Approximate Message Passing Algorithm

AWGN Additive White Gaussian Noise

BER Bit Error Rate

BP Belief Propagation

BSC Binary Symmetric Channel

CN Check Node

CR Cognitive Radio

CS Compresive Sensing

CTF Continuous to Finite

DDS Direct Digital Synthesis

DTFT Discrete Time Fourier Transform

EAMP Enhanced Approximate Message Passing Algorithm

EXIT Extrinsic Information Transfer

FDD Frequency Difference Detection

FFT Fast Fourier Transform

GF Galois Field

IMV Infinite Measurement Vector

IPE Independent Probability Estimate

JPE Joint Probability Estimate

LDIC Low Density Integer Constrained

LDPC Low Density Parity Check

LLR Log-Likelihood Ratio

LP Linear Programming

MMV Multiple Measurement Vector

MP Message Passing

MSE Mean Square Error

MWC Modulated Wideband Converter

NSP Null Space Property

OMP Orthogonal Matching Pursuit

PFD Power Flux Density

PN Pseudo Random

PSK Phase Shift Keying

RD Random Demodulator

RIP Restricted Isometry Property

SMV Single Measurement Vector

SNR Signal to Noise Ratio

SNSS Sub-Nyquist Sampling Scheme

SVC Scaleable Video Codec

TICA Time Interleaved Converter Array

VN Variable Node

List of Tables

1.1 Complexity and minimum measurement requirement comparison of some reconstruction algorithms 11

5.1 Average number of sparsity detection errors for IPE and JPE for increasing iteration count i_c when $L = 3840$ and $N = 5120$ 85

5.2 Computational Cost per Bit of IPE, JPE and OMP 95

5.3 Set of bit sequence for $w_r = 3$. 97

A.1 Performance comparison of CS based SVC with standard H.264 model 132

List of Figures

2.1 Block diagram of multicoset sampling. 17
2.2 Block diagram of random demodulator. 19
2.3 Block diagram of modulated wideband converter. 22

3.1 Block diagram of modified MWC architecture. 31
3.2 MSE comparison for MWC and modified MWC. 32
3.3 MSE comparison for different mixing sequence in MWC. 33
3.4 Block diagram of overall system. 34
3.5 Block diagram of CS receiver for jamming detection and filtering. . 35
3.6 Estimated power spectrum of input with jamming signal. 37
3.7 Power spectrum of receiver output signal after filtering. 37
3.8 MSE vs band number plot of MWC architecture. 40

4.1 Model of spectrally sparse multiband signal (maximum subband bandwidth = B and total bandwidth = W). 43
4.2 Hardware architecture of the proposed CS framework. 44
4.3 Signal processing steps for signal reconstruction of proposed CS framework. 45
4.4 An approach for direct digital synthesis of the proposed analog mixing signal. 47
4.5 Analog mixing signal generation by a comb generator. 47
4.6 Pictorial representation of effect of low density measurement matrix ($w_r = 3, w_c = 2$ in this illustration). 49
4.7 State diagram of a binary symmetric channel. 55
4.8 Theoretical lower bound for different rates. 55

4.9 Algorithm performance comparison (size of A vs sparsity). 63
4.10 Recovery performance of proposed architecture in frequency domain with CW input signal. 64
4.11 Recovery performance of proposed architecture in frequency domain with QPSK input signal. 64
4.12 Simulation setup for noise performance evaluation of CS. 67
4.13 Simulated wideband signal (noiseless case) with $N = 36$ narrow bands and four active bands. 68
4.14 QPSK bit error probability: proposed approach versus MWC. . . . 68
4.15 Noise amplification at different subbands; MWC versus proposed approach. 69

5.1 Tanner graph model of the CS spectrum sensing architecture. . . . 72
5.2 Connectivity graph for N variable nodes connected to a check node. 78
5.3 Possible states of variable node vectors for a given N and S_m. . . . 78
5.4 Coherence measure for matrix A (Gallager method). 81
5.5 Coherence measure for matrix A (without cycles of length 4 and 6). 82
5.6 Convergence of IPE algorithm for different p (rate $= 0.75$ and $N = 5120$. 83
5.7 Performance comparison of IPE algorithm. 83
5.8 Convergence of the estimated probabilities of the variable nodes to 0 or 1 for IPE and JPE ($L/N = 0.5, N = 4958$). 84
5.9 A simulation based comparison of sparsity detection algorithms. . 85
5.10 Threshold p^* for rate $L/N = 0.5$ with different values of N. 86
5.11 IPE Convergence with different parametric variations. 87
5.12 IPE convergence for different values of w_r ($w_c = 3$, $p_1 = 0.1$). 87
5.13 Convergence of IPE algorithm in presence of measurement noise (σ_2 is varied and $\sigma_1 = 0$, $p_1 = 0.2$, $N = 1280$ and rate $L/N = 0.75$). 88
5.14 Convergence of JPE algorithm in presence of measurement noise (σ_2 is varied and $\sigma_1 = 0$, $p_1 = 0.2$, $N = 1280$ and rate $L/N = 0.75$). 89
5.15 Success rate for different σ_1 ($\sigma_2 = 0$, $p = 0.04$ and $N \approx 1280$). 89

5.16 Success rate for different σ_2 and p ($\sigma_1 = 0.02$, $L = 960$ and $N = 1280$ and rate $L/N = 0.75$). 90
5.17 Success rate for varying p at different rates L/N ($\sigma_1 = 0.02, \sigma_2 = 0.04$ and $N \approx 1280$). 90
5.18 Success rate for varying σ_i, $i = 1, 2$ at different values of p and L/N. 91
5.19 Statistical evaluation of convergence of IPE algorithm by density evolution method. 92
5.20 Statistical evaluation of convergence of IPE algorithm by EXIT chart method ($p = 0.1; rate = 1/2.$) 93
5.21 Statistical evaluation of convergence of IPE algorithm by EXIT chart method ($p = 0.15; rate = 1/2.$) 93
5.22 Convergence of the proposed IPE algorithm (Heat Map). 94
5.23 Simulation result for sparsity detection with non-binary discrete inputs. 99
5.24 Performance comparison of deterministic algorithm with JPE. . . . 101

6.1 Evaluation of the individual terms of (6.7). 109
6.2 Efficiency η. 109

A.1 Block Diagram of CS based SVC 129
A.2 Processing steps of Scalable Video Encoder 129
A.3 3D-DWT Process at SVC Encoder 130
A.4 Conversion from MMV to SMV at Encoder 130
A.5 Reconstructed frame from video at different resolutions 131

CHAPTER 1

Introduction

With wireless communication becoming omnipresent, there is an increasing need for cheaper, smarter and more energy-efficient wireless devices. However, there is also an increasing need for higher amount of information processing and transfer, which results in greater energy consumption by the signal processing engine of these devices. Furthermore, the required sampling rate often exceeds the capability of the commercially-available analog to digital converters (ADC). Thus, there are two objectives that stand in contrast: design wireless devices (i) that have high energy efficiency and (ii) that can generate, process, and transfer information at high rate. Satisfying these two objectives simultaneously poses a design challenge.

The communication spectrum is a limited resource. Development of multiple radio standards and their static spectrum allocation leads to under-utilisation of allocated spectrum. Real-time wideband spectrum sensing will enable allocation of un-utilised spectrum to secondary users on need basis, and thus improve its utilisation. Nyquist sampling based real-time wideband spectrum sensing is difficult to realise using the available commercial ADCs due to their limited input analog bandwidth and sampling rate. Additionally, it also needs a large amount of hardware.

Numerous efforts have been made to develop an efficient sampling scheme which results in minimal sampling rate and reconstruction of original signal from those samples. The sampling schemes utilise different characteristics of input analog signals such as spectral spread (bandlimited or bandpass signals) to achieve optimum sampling rate. Different sampling techniques have been studied in de-

tail, starting with classical Shannon-Nyquist sampling [1–3], Papoulis expansion of sampling theorem [4], under-sampling or bandpass sampling [5], multichannel sampling [6], and non-periodic sampling [7,8].

Sub-Nyquist signal sampling, which offers a wideband signal with maximum frequency of f_h to be sampled at a rate less than the Nyquist rate $f_{\mathrm{NYQ}} = 2f_h$, offers computational and storage advantages. However, it is not always possible to achieve the optimum sampling rate due to the spectral occupancy pattern of the input signal [5]. A practical approach to implement high rate sampling is to distribute the task to a bank of M parallel samplers [4]. For example, in Time Interleaved Converter Arrays (TICA) [9], there are M parallel branches and sampling is performed at a rate $f_s = f_h/M$ in each branch. Incoming signal is imparted a time-shift (delay) of $\tau_m = m/f_{\mathrm{NYQ}}$ at m^{th} branch, where $0 \leq m \leq M-1$. While TICA is a Nyquist rate method (aggregated rate over M branches equals f_{NYQ}), each individual branch operates at a rate lesser than the Nyquist rate. While efforts are on to use TICA for wideband communications [10], there are some issues (e.g. timing mismatch) [11] that need to be addressed.

Compressive Sampling or Compressive Sensing (CS) is a sub-Nyquist sampling scheme which exploits the sparsity of input signal [12] to represent the high dimensional sparse wideband signal with a few measurements and techniques to recover signal from those measurements. In compressive sampling, a high-dimensional sparse signal is under-sampled by linearly projecting the sparse signal onto a few basis vectors that forms a measurement matrix. This matrix has a lesser number of rows compared to the number of columns and it represents an under-determined system of linear equations. In CS, this matrix system is typically inverted using convex optimisation or a greedy pursuit approach.

Under-utilisation of allocated spectrum results in sparsity of spectrum occupancy in time and frequency domain, which motivates the use of compressive sampling schemes for wideband cognitive radio communications [13,14]. Therefore, as the needs of real-time spectrum sensing and wideband communication are growing, work has begun on different possible CS system architectures and signal processing techniques for energy-efficient wideband communication. This

work is aimed at developing efficient CS architecture and algorithms for wideband sensing.

1.1 Problem Statement

The CS system implementation has resulted in hardware architectures such as Multicoset sampling [15], Random demodulator [16], and Modulated Wideband Converter (MWC) [17]. In Multicoset sampling, the analog signal is sampled using a bank of low-rate ADCs where a random delay (subsample shift) is introduced in each branch. Its drawbacks are that it requires the resolution of this subsample delay to be on the order of bandwidth W, and that the finite bandwidth of practically available ADCs restricts the maximum frequency of the signal that can be accommodated. The second approach, the Random demodulator is more sensitive to analog component tolerances, jitters, and non-linearity of mixers. MWC overcomes these limitations to a large extent but the problem of noise foldover due to deliberate aliasing introduced in each of the parallel branches [17] remains a concern. By design, the communication performance of signals towards the band edges gets penalised in MWC.

This work investigates the existing CS architecture and proposes an improved CS framework that offers uniform treatment to all input bands and mitigates noise enhancement problems of the present CS implementations such as MWC.

Sparse spectrum occupancy over wideband results in inefficient spectrum utilisation. Recently, spectrum sensing techniques have inspired applications in Cognitive Radio (CR) communications [18] to improve spectrum utilisation. Different CS techniques have been proposed for spectrum sensing [19–21]. The major challenge in it is to perform blind spectrum sensing or sparsity detection using minimal hardware and signal processing resources.

The second proposal of this work is to develop algorithms for blind sparsity detection under low SNR condition and their theoretical as well as simulation based performance analysis. Finally, we build an improved CS framework for wideband communication applications.

1.2 Theoretical Background of Compressive Sampling

Most of the real world signals often have a sparse representation in some domain. Transform coding exploits the sparsity of signals to achieve compression. CS has emerged as a framework for simultaneous sensing and compression of finite-dimensional vectors, which relies on linear dimensionality reduction. It enables potentially large reduction in sampling rate and computational cost for signals that have a sparse or compressible description [12].

A real-valued, finite-length, discrete-time signal x ($x \epsilon \mathbf{R}^N$), can be represented in terms of basis vectors as

$$x = \sum_{i=1}^{N} s_i \psi_i \tag{1.1}$$

where, $\mathbf{S}$ is the $N \times 1$ column vector of weighting coefficients $s_i =< x.\psi_i >$, and $\mathbf{S}$ is an equivalent representation of the signal x in Ψ domain. Mathematically, the sampled signal x is said to be k-sparse (or compressible) if it can be well approximated by $k \ll N$ (N being number of Nyquist samples) coefficients under linear transform as given in (1.2).

$$x = \Psi \mathbf{S} \tag{1.2}$$

Here, Ψ is the sparsifying basis, and $\mathbf{S}$, the transform coefficient vector that has at most k (significant) non-zero entries. According to CS theory, such a signal can be acquired through the following random linear projections as shown in (1.3) as

$$\mathbf{y} = \Phi \mathbf{x} + \mathbf{e} \tag{1.3}$$

where, $\mathbf{y}$ is the sampled vector with $M << N$ data points, Φ represents an $M \times N$ measurement or sensing matrix and $\mathbf{e}$, the measurement noise. CS theory suggests that it is possible to recover signal $\mathbf{x}$ faithfully from $\mathbf{y}$ using $M = O(k\, log(N))$ measurements, provided the input signal $\mathbf{x}$ is sparse and sensing matrix is designed appropriately [22]. It is important to note here that as per CS theory the sensing matrix remains the same for all class of signals.

The CS framework acquires the compressible signal representation directly. It does not follow the intermediate stages of acquiring N samples and then trans-

forming them using a sparsifying basis to achieve compression. Considering a general linear measurement process that computes $M \ll N$ inner products between $\mathbf{x}$ and a collection of vectors a_j^M as in $y_j =< x, a_j >$, and arranging the measurements y_j in a $M \times 1$ vector $\mathbf{y}$ and the measurement vectors a_j^T as rows in an $M \times N$ matrix $\mathbf{A}$, $\mathbf{y}$ can be written as

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{1.4}$$

where, $\mathbf{A} = \phi\psi$ is an $M \times N$ matrix called measurement matrix. The measurement process need not be adaptive, meaning that ϕ is fixed and does not depend on signal $\mathbf{x}$. In this framework, the problem consists of (a) designing a stable measurement matrix ϕ (matrix $\mathbf{A}$) such that the salient information in any k-sparse or compressible signal is not lost due to dimensionality reduction from $\mathbf{x}\epsilon\mathcal{R}^N$ to $\mathbf{y}\epsilon\mathcal{R}^M$ and (b) development of a reconstruction algorithm to recover $\mathbf{x}$ using only $M(M << N)$ measurements $\mathbf{y}$ and given sensing matrix $\mathbf{A}$.

1.2.1 Properties of a Sensing Matrix

The main design criteria for the sensing matrix $\mathbf{A}\,\epsilon\,\mathcal{R}^{M\times N}$ are to ensure the unique identification of a sparse signal $\mathbf{x}$ in the transformed or measurement domain, i.e. for any two sparse vectors $\mathbf{x_1}$ and $\mathbf{x_2}$ with the same sparsity (k) being sensed. Using matrix A, it should be possible to reconstruct the input vectors uniquely using their respective measurement vectors $\mathbf{y_1}$ and $\mathbf{y_2}$ under the framework of $\mathbf{y} = \mathbf{A}\mathbf{x}$. Here, matrix $\mathbf{A}$ reduces dimensionality and $\mathbf{x}$ is assumed as a finite length window of Nyquist rate samples. However, the issue of directly acquiring compressive measurement without first sampling at Nyquist rate is ignored.

The major CS challenge is the design of the sensing matrix $\mathbf{A}$ and to decide on the minimum number of measurements needed for signal recovery. The design of suitable signal processing techniques to recover $\mathbf{x}$ from incomplete or compressed measurements $\mathbf{y}$ is also needed.

Properties of Sensing Matrix* A *for Stable Reconstruction The property of a stable sensing matrix for unique reconstruction guarantee has been studied in the litera-

ture using various properties or metric such as Null Space Property (NSP), Spark, Restricted Isometry Property (RIP), Coherence, and Mutual Coherence [12].

Null Space Property (NSP): The null space of a matrix $\mathbf{A}$ is defined as $\mathcal{N}(\mathbf{A})$, where,

$$\mathcal{N}(\mathbf{A}) = \{z : \mathbf{A}z = 0\} \tag{1.5}$$

NSP suggests that for given sensing matrix $\mathbf{A}$, all $\mathbf{x} \in \sum_k$ ($\sum_k$ represents the set of k sparse vectors) can be recovered uniquely from measurements $\mathbf{y}$, provided, $\mathcal{N}(\mathbf{A})$ does not contain any vectors in $\sum_{2k}$.

Spark: The spark of a matrix $\mathbf{A}$ is the smallest number of columns of $\mathbf{A}$ that are linearly dependent. It is denoted by $\sigma(\mathbf{A})$.

For any vector $y \in R^M$, there exists at most one signal $\mathbf{x} \in \sum_k$, such that $\mathbf{y} = \mathbf{A}\mathbf{x}$ if any only if $\sigma(\mathbf{A}) \geq 2k$. Therefore, for unique recovery of $\mathbf{x}$ from measurement $\mathbf{y}$ and given $\mathbf{A}$, we must have minimum measurement sample size equal to $M \geq 2k$.

While NSP or Spark based conditions are necessary and sufficient to establish guarantees for sparse signal recovery, they do not account for noise. Candes and Tao [23] [24] had introduced the Restricted Isometry Property (RIP) which provides the error bounds on recovery for a given sensing matrix with noisy measurement.

Restricted Isometry Property (RIP): A matrix $\mathbf{A}$ is said to satisfy the RIP of order k if there exists a positive constant $\delta_k \in (0, 1)$ such that

$$(1 - \delta_k)\|x\|_2^2 \leq \|\mathbf{A}x\|_2^2 \leq (1 + \delta_k)\|x\|_2^2 \tag{1.6}$$

for all $x \in \sum_k$.

According to CS theory, the sensing matrix $\mathbf{A}$ that satisfies RIP of order $2k$ approximately preserves the distance between any pair of k-sparse vectors and guarantees stable recovery. Matrices which satisfy RIP, also satisfy NSP [12].

While RIP provides tighter bounds of recovery errors, it is often difficult to compute for large sensing matrix. In many cases, it is easy and preferable to use the other properties of $\mathbf{A}$ that provide more concrete recovery guarantee. Coherence is one such property which is used extensively in CS matrix evaluation.

Coherence: The coherence of a matrix $\mathbf{A}$ (denoted as $\mu(\mathbf{A})$) is defined as the largest absolute inner product between any two columns a_i and a_j of $\mathbf{A}$. This is expressed as:

$$\mu(\mathbf{A}) = \max_{1 \leq i < j \leq N} \frac{|\langle a_i, a_j \rangle|}{\|a_i\|_2 \|a_j\|_2} \tag{1.7}$$

For any matrix $\mathbf{A}$ of size M x N (with $M \leq N; N \geq 2$), whose columns are normalised (i.e.$\|a_i\| = 1$ for all i), the coherence of $\mathbf{A}$ satisfies the limit given by

$$\sqrt{\frac{N-M}{M(N-1)}} \leq \mu(\mathbf{A}) \leq 1 \tag{1.8}$$

In CS framework, coherence is used as a measure of stability for sensing matrices. For a sensing matrix $\mathbf{A}$, if the sparsity k satisfies (1.9),

$$k < \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{A})}\right) \tag{1.9}$$

then for each measurement vector $\mathbf{y} \in R^M$ there exists at most one signal $\mathbf{x} \in \sum_k$ such that $\mathbf{y} = \mathbf{A}\mathbf{x}$.

RIP constant and coherence: If $\mathbf{A}$ has unit-norm columns and coherence μ (= $\mu(\mathbf{A})$), then matrix $\mathbf{A}$ always satisfies RIP of order k with $\delta_k \leq (k-1)\mu$. This relationship outlines the importance of coherence as a measure for quality of sensing matrices for signal recovery in compressive sensing framework. For sparse binary sensing matrices, coherence is used as a measure for CS recovery.

Mutual Coherence: When the sensing matrix is a combination of two matrices (a sparsifying basis and another random matrix), mutual coherence is used as a measure to determine the recovery guarantees. Mutual coherence is defined as the largest absolute inner product between columns of two constituent matrices of sensing matrix. Let, the sensing matrix $\mathbf{A} = [\Phi, \Psi]$ be the concatenation of two orthogonal bases Φ and Ψ, where both are unitary and of size N x N, then mutual coherence $\mu(\Phi, \Psi)$ satisfies the bounds given by,

$$\frac{1}{\sqrt{N}} \leq \mu(\Phi, \Psi) \leq 1 \tag{1.10}$$

For any given sensing matrix $\mathbf{A} = [\Phi, \Psi]$, if the sparsity of input signal k and mutual coherence satisfy the relationship given by (1.11),

$$k < \frac{1}{\mu(\Phi, \Psi)} = \frac{1}{\mu(\mathbf{A})} \tag{1.11}$$

then for each measurement vector $\mathbf{y} \in R^M$ there exists at most one signal $\mathbf{x} \in \sum_k$ such that $\mathbf{y} = \mathbf{A}\mathbf{x}$, which provides the CS recovery guarantee for a given input signal through measure of mutual coherence.

1.2.2 Recovery from Compressed Measurements

In the compressive sensing setup, for a given k sparse signal $\mathbf{x} \in R^N$ ($k \ll N$), the compressed measurement $\mathbf{y}$ and sensing matrix $\mathbf{A}$ are related as given in (1.12).

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{1.12}$$

Here, $\mathbf{A} \in R^{MxN}$ and $\mathbf{y} \in R^M$. In case of CS, $M \ll N$, this system of linear equation results in an under-determined system. The sparsity assumption helps in the recovery of $\mathbf{x}$ from $\mathbf{y}$ for the given sensing matrix $\mathbf{A}$. For reconstruction, the natural attempt is to solve this optimization problem by finding

$$\widehat{x} = \underset{y = \mathbf{A}x}{\arg\, min} \; \|x\|_0 \tag{1.13}$$

This turns out to be an NP-Hard problem. However, Donoho [22] suggested that for a given matrix $\mathbf{A}$ which satisfies the condition of RIP, the generally NP-Hard (l_0) problem becomes equivalent to its convex relaxation problem as given by

$$\widehat{x} = \underset{y = \mathbf{A}x}{\arg\, min} \; \|x\|_1 \tag{1.14}$$

where, $M \approx Klog(N) \ll N$. In the most practical cases, when $\mathbf{x}$ is not exactly sparse and corrupted with noise, we consider the mathematical program for re-

construction given by

$$min\|x\|_1 \quad \text{subject to} \quad \|(\mathbf{A}x - y)\|_2 \ll \epsilon \tag{1.15}$$

Basis Pursuit (BP): Basis Pursuit is the most popular reconstruction algorithm in CS which can reconstruct a sparse signal with a high accuracy by solving a convex optimisation problem through linear programming (LP) [25]. The l_1 norm minimisation

$$min\|x\|_1 \quad \text{subject to} \quad \mathbf{A}x = y \tag{1.16}$$

solution is unique and is the closest estimate to x_0. The major advantages of Basis Pursuit (and its related methods) are high performance and robustness, whereas its weakness is high computational complexity.

Matching Pursuit (MP): Matching pursuit is an iterative greedy algorithm which takes advantage of greedy search and vector projection to reconstruct a sparse signal [26]. It reconstructs the k-sparse signal by iteratively constructing a support set S of the signal. At each iteration, matching pursuit optimises the approximation by selecting one column (called an atom) which has the maximum correlation (the inner product with the largest absolute value) with the residue $\mathbf{R}$ from the measurement matrix $\mathbf{A}$ (also called dictionary D). Then, matching pursuit updates the support set by appending selected atoms till termination. The drawbacks of matching pursuit are its slow convergence rate and poor sparse reconstruction performance than Basis Pursuit.

Orthogonal Matching Pursuit (OMP): Orthogonal Matching Pursuit, an improved version of Matching Pursuit, addresses the drawbacks of matching pursuit by projecting the signal orthogonal to the set of all selected atoms [27]. The principle of OMP is the same as that of matching pursuit, but its major difference is that it never chooses an atom selected in the previous iterations since the residual $\mathbf{R}$ is orthogonal to the already chosen atoms.

If a sensing matrix $\mathbf{A}$ satisfies the RIP of order $(k+1)$ with $\delta_{k+1} < \frac{1}{3\sqrt{k}}$ and given $\mathbf{y} = \mathbf{A}\mathbf{x}$, OMP can recover the k-sparse signal exactly in k iterations. Since, we expect the columns of sensing matrix $\mathbf{A}$ to be approximately orthogonal, then

$\mathbf{A} * \mathbf{Ax}$ is locally a good approximation of $\mathbf{x}$. Therefore, the largest coordinate of the observation vector $\mathbf{A} * \mathbf{Ax}$ corresponds to a non-zero entry of $\mathbf{x}$. Thus, one coordinate for the support of $\mathbf{x}$ is estimated. Its subtraction from the observation vector $\mathbf{y}$ and repeating the process k times yields the entire support of the signal $\mathbf{x}$. OMP is quite fast, both in theory and in practice, but its reconstruction guarantees are not as strong as that of BP.

Many variants of OMP such as Stagewise OMP (StOMP), Regularised OMP (ROMP), and Compressive Sampling Matching Pursuit (CoSAMP)) are applied [28, Chapter 8] for signal reconstruction.

Approximate Message Passing Algorithm (AMP): AMP algorithm proposed by Donoho, Maleki, and Montanari [29] is a low complexity algorithm with reconstruction guarantee similar to Basis Pursuit. The AMP method has been derived using the statistical concept of "state evolution", and considerably accelerates the convergence rate in special CS-decoding applications.

The standard AMP algorithm iteratively updates the estimates of the unknown input signal, with $\mathbf{x^t}$ being the estimate and $\mathbf{z^t}$ being the residue at the t^{th} iteration. Based on the iteration count and the measurement vector (y), AMP estimates the sparse signal $\hat{x}$ which is almost similar to x, starting with the initial guess $x^0 = 0$ and $z^0 = y$. The first order AMP algorithm proceeds iteratively according to the following equations [30]:

$$x^{t+1} = \eta_t(\phi^* z^t + x^t) \tag{1.17}$$

$$z^{t+1} = y - \phi x^t + z^{t-1}(\eta_t^{'}(\phi^* z^{t-1} + x^{t-1})) \tag{1.18}$$

Here, $\eta_t(.)$ is a scalar threshold function. An improved version of AMP called Enhanced Approximate Message Passing (EAMP) algorithm performs exceedingly well for many deterministic and highly structured matrices compared to the AMP algorithm.

Compressive video sampling using AMP has been proposed in the literature [31]. During the performance evaluation of different reconstruction algorithms, we conceptualised and simulated a novel design of CS based Scalable Video Codec

(SVC) which uses Bernoulli sensing matrices to encode high frequency sparse 3-D wavelet coefficient and decodes them successfully at decoder using EAMP algorithm. While this technique offers a better compression ratio without compromising on quality (PSNR), the reconstruction is robust and fast compared to AMP. The details of CS based Scalable Video Codec application is included in Appendix-A [32–34].

In addition to these popular reconstruction algorithms, other known algorithms available in the literature are LASSO and Basis Pursuit Denoising (BPDN) based on ℓ_1 penalized Least Squares [28, Chapter 9], Bayesian Belief Propagation [35], iterative thresholding [29], and CS Linear Programming Decoding [36].

A detailed study on sparse signal recovery guarantee for different reconstruction algorithms is presented in [12]. The table 1.1 presents the complexity of the some of known reconstruction algorithms where, N represents length of signal, M is the minimum number of measurements required for reconstruction and k is sparsity.

Algorithm	**Complexity**	**Minimum Measurement**
Basis Pursuit	$O(N^3)$	$O(k\ logN)$
OMP	$O(kMN)$	$O(k\ logN)$
StOMP	$O(N\ logN)$	$O(N\ logN)$
Belief Propagation	$O(N\ log^2N)$	$O(k\ logN)$

Table 1.1: Complexity and minimum measurement requirement comparison of some reconstruction algorithms

A detailed review on complexity and memory requirements of sparse recovery algorithms is also described in [37].

1.2.3 Signal Measurement Models in CS

Compressive sampling framework is applicable to all classes of signals. Depending upon the application setup, their measurement and signal recovery algorithms can be optimised for better performance. The measurement setup and signal recovery in CS setup are categorised into three main segments namely Single Mea-

surement Vector (SMV), Multiple Measurement Vector (MMV), and Infinite Measurement Vector (IMV) [12].

Single Measurement Vector (SMV): According to CS theory, it is possible to uniquely recover the k sparse input signal vector $\mathbf{x}$ of size $N \times 1$ from measurement vector $\mathbf{y}$ of size $M \times 1$ (where, $M > 2k$) obtained using sensing matrix $\mathbf{A}$ of size $M \times N$ (here, $M \leq N$). A setup, wherein the input signal vector $\mathbf{x}$ is recovered directly using single measurement vector $\mathbf{y}$ and knowledge of sensing matrix $\mathbf{A}$, is called the Single Measurement Vector (SMV). Recovery in such setups are performed in two stages, support recovery using algorithms like OMP and reconstruction of input signal using measurement $\mathbf{y}$ [12].

Multiple Measurement Vector (MMV): There are real-life applications where CS theory is applicable in distributed acquisition of multiple correlated signals. The cases wherein multiple signals are sparse and exhibit the same indices for non-zero locations (or jointly sparse) are referred to as MMV setup. In an MMV setting, L measurements of $\mathbf{y_i}$ $(= \mathbf{Ax_i})$ are given, where vectors $\mathbf{x_i}; 1 < i < L$ are jointly sparse. By making multiple measurements and stacking these vectors into the columns of a matrix $\mathbf{X}$, there will be at most k non-zero rows in $\mathbf{X}$ which occur on a common location set. In such MMV cases, $\mathbf{X}$ is said to be row-sparse. It is different from SMV as we use multiple measurement vectors for recovery of input signal. Rather than trying to recover for each $\mathbf{x_i}$ separately, we try to jointly recover the set of vectors using their common sparse support.

In MMV setup, the recovery approach uses convex optimisation and greedy methods similar to SMV setup. However, specialised algorithms such as ReMBo were also introduced which convert the MMV problem into SMV problem and then utilise SMV recovery techniques for better performance [12].

Infinite Measurement Vector (IMV): A CS measurement is called an IMV setup, if the measurement set λ is continuous in time or it consists of infinitely large number of samples.

$$\mathbf{y}(\lambda) = \mathbf{Ax}(\lambda), \quad \lambda \in \Omega \tag{1.19}$$

An IMV system comes up naturally whenever analog signals are considered. IMV problems are converted into equivalent MMV problems and then signal recovery

is performed. In this approach, the number of unknowns is large and this increases the computation load. A specialised method such as continuous to finite (CTF) [38] is proposed to convert the IMV problem into an MMV problem for signal recovery.

1.2.4 Realisable CS Architectures

The hardware realisation of the CS framework results in an equivalent sensing matrix, which should meet the signal reconstruction guarantees similar to RIP. Very few deterministic construction methods are available in the literature [39], and the hardware implementation of CS sensing matrix has evolved in different architectures. Some popular architectures include Multicoset sampling [15], Random demodulator [16], Modulated Wideband Converter (MWC) [17], and variants of MWC. The MWC architecture offers benefits such as hardware realisation using available low sampling rate devices, analog mixers, and low pass filters. While MWC does not depend on limited analog bandwidth of available ADCs, it has several other constraints. The detailed review of available CS hardware realisations are presented in Chapter 2.

1.3 Research Contribution

The major research contributions of this dissertation are:

1. A regular sparse binary LDPC matrix based realisable novel CS hardware architecture for frequency domain sparse multi-band wideband communication system is proposed. CS implementation in the form of analog front-end and use of complex sinusoid as mixing sequence has ensured uniform performance of all input bands and mitigates noise enhancement present in CS implementations such as in MWC [40,41].

2. Two new Belief Propagation (BP) algorithms for blind sparsity detection have been developed. The theoretical lower bound on the size of the measurement matrix for a given sparsity level is derived. The performance of

developed algorithm is closer to theoretical lower bound than any other similar detection techniques [40, 41].

3. Semi-analytical convergence analysis using Density Evolution and EXIT chart methods has been performed for the developed sparsity detection algorithms. Also simulation based performance verification has been done to establish that the proposed architecture together with developed sparsity detection algorithms offer minimum one fourth hardware savings [41].

4. Noise performance analysis of PSK communication system within the proposed CS architecture and simulation based performance verification has demonstrated that the proposed CS system offers better noise mitigation than the published MWC framework [41].

5. Theoretical analysis and framework to use BP algorithms for compression and denoising application are also presented. Also, a deterministic algorithm is developed for lossless compression [42].

6. During the dissertation work, proposal for CS-based applications such as (i) receiver for interference detection and filtering [43–46](ii) Scalable Video Codec implementation [33, 34] were envisioned, simulated, and presented. Presently we are working on its hardware implementation.

1.4 Overview of the Dissertation

This dissertation work is organised in seven chapters. The first introduces the theoretical background of compressive sampling, the problem statement, and highlights major research contributions. The second chapter includes a comprehensive review of different CS architectures and signal processing techniques for spectrum sensing. The third chapter provides a detailed review of MWC architecture and variants, their performance and limitations. The fourth and fifth chapters cover the novel contributions made in the development of CS architecture and sparsity detection algorithms. They include derivation of theoretical lower bound

and convergence analysis for developed algorithms together with noise performance analysis of the proposed CS system for wideband communication. The sixth chapter proposes the concept of using the sparsity detection algorithms for compression and denoising applications. The last seventh chapter summarises the research work and presents the future work.

CHAPTER 2

A Review of CS Architectures and Spectrum Sensing Techniques

CS is one of the sub-Nyquist signal acquisition technique which performs signal acquisition and compression simultaneously. The performance of CS relies on two fundamental requirements: the first is representation of the signal using sparse basis and the second is the design of measurement matrix [47] [48]. The measurement matrices and their hardware realisations are key to CS recovery performances for any application. The efficient implementation of CS measurement matrix using available hardware technologies has been challenging, especially for wideband communication.

The first section of this chapter presents an overview of few popular CS realisations and their equivalent measurement matrices representation. It also highlights advantages and shortcomings of those architectures. The second section of this chapter presents the needs of spectrum sensing and provides a brief review of signal presence detection techniques being used for spectrum sensing.

2.1 Hardware Architectures for Compressive Sampling

CS is highly promising for wideband communication systems as it brings sampling and compression together to decrease the processing power and memory requirements. There are several techniques in literature for signal recovery from compressed samples [49] making the CS suitable candidate for various applications. Many hardware architectures for CS implementation have been proposed.

Here, We briefly discuss few popular CS hardware architectures.

2.1.1 Multicoset Sampling

Multicoset sampling proposed by Venkataramani and Bresler [50] is one of the first CS implementations, which reduces the sampling frequency of multi-band signals, whose frequency support is finite union of intervals. There are many reported implementations of Multicoset sampling framework. The most popular multicoset implementation is composed of several parallel branches, each with a time shift followed by a uniform sampler operating at a sampling rate lower than the Nyquist rate. Domìnguez-Jitnènez and N.González-Prelcic proposed *Synchronous Multirate Sampling* which uses uniform samplers operating at different rates to realize multicoset framework [51]. A subset of synchronous multirate sampling namely dual sampling architecture is presented for multicoset sampling by Moon et al. [52].

In multicoset sampling, the analog signal $x(t)$ is sampled at Nyquist rate and the Nyquist grid is divided into successive segments of L samples each. In each segment only p samples out of L are retained. Those p samples described by the set is called coset (C). In generic multicoset implementation, the input signal is fed into m parallel branches which are maintained at different time shifts. Few (at least one) samples from each branch are picked up as defined by cosets. The block diagram of multicoset sampler is shown in Fig. 2.1.

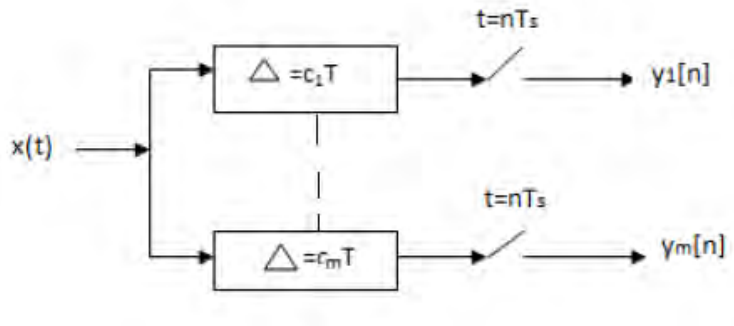


Figure 2.1: Block diagram of multicoset sampling.

The multicoset sampling is a selection of certain samples from the Nyquist grid. A constant set C of length p describes the indices of p samples that are

retained in each block while the rest are zeroed out. The set $C = \{c_i\}_{i=1}^{p}$ is referred to as the sampling pattern or coset where,

$$0 \leq c_1 \leq c_2 ... \leq c_p \leq L-1, \tag{2.1}$$

Here, we define the i^{th} sampling sequence for $1 \leq i \leq p$ as given by (2.2),

$$x_{c_i}[n] = \begin{cases} x(t = nT), & n = mL + c_i, for \quad m \in Z \\ 0, & otherwise \end{cases} \tag{2.2}$$

The sampling stage is implemented by p uniform sampling sequences with period $1/(LT)$, where the i^{th} sampling sequence is shifted by $c_i T$ from the origin. Therefore, a multicoset system is uniquely characterized by the parameters L, p, and the sampling pattern C.

The calculation of the Discrete-Time Fourier Transform (DTFT) of $x_{c_i}[n]$ as represented by $X_{c_i}(e^{j2\Pi fT})$ is shown as

$$X_{c_i}[e^{j2\pi fT}] = \frac{1}{LT} \sum_{r=0}^{L-1} exp(j\frac{2\pi}{L} c_i r) X(f + \frac{r}{LT}) \tag{2.3}$$

$$\forall f \in \mathcal{F}_0 = \big[0, \frac{1}{LT}\big), \quad 1 \leq i \leq p \tag{2.4}$$

In CS framework, the goal is to choose parameters L, p, and C such that $X(f)$ can be recovered. The above equation in a matrix form is represented as

$$\mathbf{y}(f) = \mathbf{A}\mathbf{x}(f), \quad \forall f \in \mathcal{F}_0 \tag{2.5}$$

where $\mathbf{y}(f)$ is a vector of length p whose i^{th} element is $X_{c_i}(e^{j2\pi fT})$, and the vector $\mathbf{x}(f)$ contains L unknowns for each f, given as follows:

$$x_i(f) = X(f + \frac{i}{LT}), \quad 0 \leq i \leq L-1, \quad f \in \mathcal{F}_0 \tag{2.6}$$

The matrix $\mathbf{A}$ depends on the parameters L, p, and the coset C. The measurement

matrix **A** is expressed as

$$\mathbf{A}_{ik} = \frac{1}{LT} exp(j\frac{2\Pi}{L} c_i k) \tag{2.7}$$

The average sampling rate of a multicoset sampling set is given by

$$\frac{1}{T_{AVG}} = \frac{p}{LT} \tag{2.8}$$

For $p < L$, the average sampling rate is lower than the Nyquist rate. In multicoset samplers, even if the input changes, parameter p and L remains fixed. Therefore, an adaptive multicoset sampler is described in [53], offers reconfiguration and better reconstruction.

Thus multicoset sampler is an efficient way of implementing CS framework but it needs the analog input bandwidth of the uniform sampler to be very high for wideband communication applications. Multicoset implementation also needs precise time shift and synchronization among parallel branches which makes it difficult to realize for higher bandwidth signals.

2.1.2 Random Demodulator

Random Demodulator (RD) is another proposed implementation of CS framework for acquisition of sparse band-limited multi-tone signal [54]. Block diagram of the random demodulator is shown in Fig. 2.2.

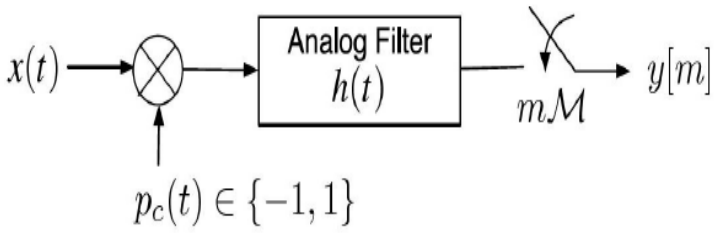


Figure 2.2: Block diagram of random demodulator.

In this implementation, the input signal $\mathbf{x}(\mathbf{t})$ is multiplied with periodic random piecewise constant chipping sequence between +1 and -1 at a Nyquist or higher rate, hence this architecture is called random demodulator. The demod-

ulation phase smears the tones across the entire spectrum due to the equivalent convolution in the frequency domain. The demodulator output is then passed through a low pass filter and sampled at a fraction of Nyquist rate. The low pass filter bandwidth and sampling rate depends on the maximum bandwidth of any band in multiband input signal.

The random demodulation results in aliasing of input bands to baseband by random amount resulting in unique signature of each tone present in the input multi-tone signal. For the mathematical analysis, the input signal $\mathbf{x}(\mathbf{t})$ can be expressed as

$$\mathbf{x} = \mathbf{FS} \tag{2.9}$$

where, **F** denotes DFT matrix and Fourier coefficients are denoted by **S**. The random periodic sequence can be expressed in Fourier domain by $\varepsilon_0, \varepsilon_1, ..., \varepsilon_{N-1}$. Therefore, the demodulation matrix can be represented as in (2.10),

$$\mathbf{D} = \begin{bmatrix} \varepsilon_0 & 0 & 0 \\ 0 & \varepsilon_1 & 0 \\ 0 & 0 & \varepsilon_{N-1} \end{bmatrix} \tag{2.10}$$

Demodulation matrix of RD is a $N \times N$ diagonal matrix which translates **x** to **Dx**. The demodulation is followed by low pass filter, which is realized as an integrator. It integrates the input over a duration before being digitized by low sampling rate ADC. The integrator can be expressed in matrix form (H) as given by (2.11).

$$\mathbf{H} = \begin{bmatrix} 1111 & & \\ & 1111 & \\ & & 1111 \end{bmatrix} \tag{2.11}$$

The above low pass filter matrix is given for a sub-sampling ratio of 4, which is a ratio of Nyquist sampling rate for input signal to low sampling rate of ADC in the RD architecture.

Mathematically, the RD architecture in matrix form is represented as **HD**. As this operates on input signal **x** (**x** has sparsity in frequency domain), the overall

sensing matrix (Φ) can be expressed as

$$\Phi = \mathbf{HDF} \tag{2.12}$$

The measurement matrix Φ satisfy the RIP property and the compressed output of RD can be expressed as $\mathbf{y} = \Phi\mathbf{x}$ [54, 55]. Using the measurement vector $\mathbf{y}$ and measurement matrix Φ, the input signal vector $\mathbf{x}$ can be recovered using proven CS recovery algorithms like Iteratively Re-weighted Least Squares method for the l_1-norm optimization [56]. The lowest required sampling rate (R) to achieve reliable reconstruction, while k is the number of frequency spikes and w is the Nyquist rate of the sparse signal, is derived empirically [55].

$$R \leq 1.7k \log(\frac{w}{k} + 1) \tag{2.13}$$

The RD sensing architecture is suitable for multi-tone signals [16]. Within RD framework, the signal recovery guarantees gets compromised due to non-ideal filter response, quantization, jitters, and mixer non-linearity, which is of major concern in RD implementation.

2.1.3 Modulated Wideband Converter

Modulated Wideband Converter (MWC) is an efficient and physically realizable sensing architecture for frequency domain sparse multiband signals as proposed by Mishali and Eldar [38]. The block diagram of MWC sensing architecture is shown in the Fig. 2.3. In MWC, the input multiband signal $x(t)$, which is composed of N bands of equal bandwidth B, enters into m channels simultaneously ($m \geq 4k$). In the i^{th} channel, $x(t)$ is multiplied by a random periodic mixing sequence $p_i(t)$, which is T_p periodic. The mixing of input signal by periodic waveform results in a linear combination of infinite f_p-shifted copies of $X(f)$ in frequency domain as expressed in (2.15). The mixing process aliases all the bands of input to baseband by a random amount determined by the DTFT coefficients of mixing sequence. The mixer output spectrum is truncated by a low-pass filter with a cut-off frequency of $1/2T_s$ and the filtered signal is sampled at rate $1/T_s$.

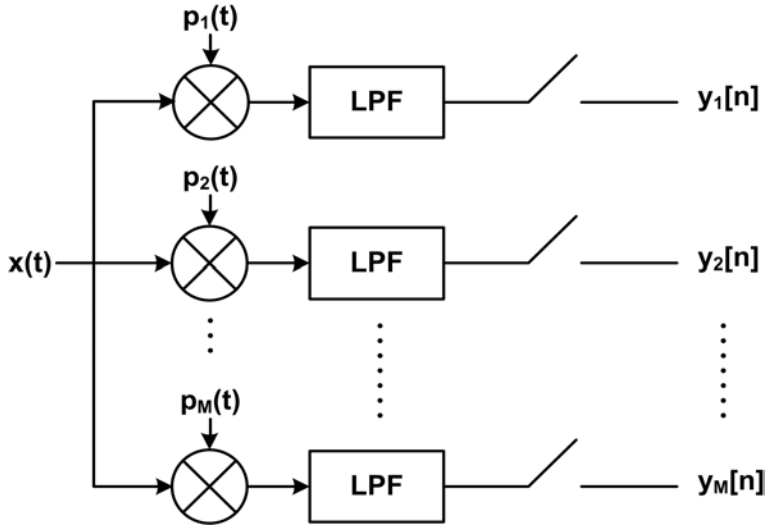


Figure 2.3: Block diagram of modulated wideband converter.

The reduced sampling rate required for each channel is set to B (or $> B$).

The Fourier transform of the output of mixer $\tilde{x_i}(t) = x(t)p_i(t)$ is expressed as:

$$\tilde{X}_i(f) = \int_{-\infty}^{\infty} \tilde{x_i}(t)e^{-j2\pi ft}dt \tag{2.14}$$

The low pass filtered output of i_{th} branch of MWC is expressed as [38]

$$Y_i(e^{j2\pi fT_s}) = \sum_{l=-L_0}^{L_0} c_{il}X_i(f - lf_p) \tag{2.15}$$

Here, f_p (=$\frac{1}{T_p}$; T_p is time period of mixing sequence) is the frequency of mixing sequence and L_0 is filter design parameter which equals the number of input bands contributing to aliasing.

In frequency domain, the MWC can be written in matrix form as [38]:

$$\mathbf{y}(f) = \mathbf{A}\mathbf{z}(f) \tag{2.16}$$

where, $\mathbf{y}(f)$ is a vector with i^{th} element as $y_i(f) = Y_i(e^{j2\pi fT_s})$ and,
$\mathbf{z}(f) = [z_1(f), z_2(f), \ldots\ldots\ldots, z_L(f)]^T$
$z_i(f) = X(f + (i - L_0 - 1)f_p); \quad 1 \leq i \leq L$;

Here, L is the length of mixing sequence given by $L = 2L_0 + 1$. L_0 is the smallest integer which contains all non-zero contribution of $X(f)$. Here, L_0 is

given by:

$$L_0 = \left\lceil \frac{f_{nyq} + f_s}{2f_p} \right\rceil - 1 \tag{2.17}$$

The measurement matrix for MWC is expressed in matrix form [38], as given by (2.18),

$$\mathbf{A} = \begin{bmatrix} a_{1,0} & \cdots & a_{1,M-1} \\ \vdots & \ddots & \vdots \\ a_{m,0} & \cdots & a_{m,M-1} \end{bmatrix} \begin{bmatrix} \vdots & \vdots & \vdots \\ F_{L_0} & F_0 & F_{-L_0} \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} d_{L_0} & & \\ & \ddots & \\ & & d_{-L_0} \end{bmatrix} \tag{2.18}$$

$$\mathbf{A} = \mathbf{S}\,\mathbf{F}\,\mathbf{D} \tag{2.19}$$

Wherein, matrix **F** represents the DFT matrix and **S** is the sign matrix with each row representing the piecewise constant value of mixing sequence corresponding to that branch (a_{il}). **D** is a diagonal matrix which represents the modulating coefficients for frequency domain response originating due to the shape of mixing waveform.

The sensing matrix **A** derived for MWC satisfy RIP guarantees. Therefore, MWC is a means of realizing the CS sensing matrix using available hardware components. MWC has m ($> 4k$) hardware branches operating with sampling rate B resulting in mB samples per second ($m \ll N$; $N = \frac{W}{B}$ = total number of input bands spread over W Hz). This results in significant hardware savings with respect to Nyquist processing. The proposed architecture can also work for input signal spread across two bands or falling on boundary of two consecutive bands with change in processing algorithms [57]. There are many MWC application frameworks reported. A detailed performance review of MWC architecture is covered in next chapter.

2.2 Spectrum Sensing Techniques

Usable spectrum is becoming scarce due to static spectrum allocation policies together with continuously increasing demand for broadcast, multimedia, and interactive services [58]. Spectrum sensing aims at detecting unused spectrum loca-

tions over the wideband such that it could be allocated to secondary users without causing interference to primary users [59].

Spectrum sensing techniques are classified into two main categories namely cooperative sensing and non-cooperative sensing. In non-cooperative sensing, the secondary users makes their own decision based on defined objectives and there is no communication among users. Cooperative sensing is primarily implemented in two ways, namely distributed sensing and centralised sensing. In case of distributed sensing architecture, the secondary users share their observations about spectrum usage (primarily uses narrowband sensing techniques) among themselves which helps secondaries to make their own decision. In centralised sensing architecture, all secondary users send their decisions to a centralised node for final decision making.

In today's cognitive communication world, real-time wideband sensing can greatly improve spectrum utilisation. Wideband spectrum sensing can be performed using Nyquist Sampling. However, challenges in wideband spectrum sensing with Nyquist sampling are requirement of high sampling rate, limited power and large amount of hardware [60]. Nyquist sampling based wideband spectrum sensing pushes contemporary ADCs to their performance limits, as the sampling rate must be at least twice the bandwidth of the signal. Since, in practise the spectrum occupancy remains sparse and Nyquist sampling at this rate may be inefficient as the signals of interest occupy limited bands within the wide multiband spectrum. Therefore, the important task is to do blind spectrum sensing at sub-Nyquist sampling rates.

Due to the sparse nature of spectrum occupancy, CS based spectrum sensing techniques are used in Cognitive Radio (CR) communications [18]. Different CS techniques have been implemented to improve the performance of wideband spectrum sensing [19–21]. There are ongoing efforts reported for terrestrial CR but still much research work has not been performed in the context of cognitive satellite communication [61]. MWC based CS receivers has significant advantages over other sub-Nyquist sampling systems for wideband applications [62,63]. However, the effectiveness of all spectrum sensing and cognitive communication schemes

lies on techniques for the signal presence detection under all channel conditions.

Spectrum Sensing or Signal Presence Detection Techniques : In literature, a number of spectrum sensing techniques have been proposed to identify the presence of the user signal transmission. These techniques are used by secondary users to identify the presence of primary users and to draw opportunistic spectrum utilization plan without causing interference to the primary users. The accuracy and latency of signal presence detection techniques has major impact on the overall system performance. Some of very popular techniques described in literature are summarized here.

2.2.1 Auto Correlation based Detection

This method exploit the features of auto correlation function of the received signal. The coefficients are calculated by:

$$R_{x,x}(\tau) = \int_{-\infty}^{\infty} x(t)x^*(t-\tau)dt$$

where τ is the lag in time and $x^*(t)$ is the complex conjugate of the signal $x(t)$. The sensing decision is based on the knowledge of the statistical distribution of the auto correlation function in presence of noise [64]. Since white noise is uncorrelated the auto correlation function of white noise results into peak at $\tau = 0$ and zero values elsewhere. However, the same auto correlation function has decreasing values with increasing τ and the difference between the auto correlation function value at $\tau = 0$ and $\tau = 1$ is used to decide the signal presence or absence. Typically a small difference indicates signal presence while large difference indicates absence of signal.

Arianandа and Leus have successfully used the auto correlation of compressive domain measurement samples to calculate the power spectral density of input signal within MWC framework [65]. This compressive wideband power spectrum estimation technique needs reduced processing power.

2.2.2 Euclidean Distance based Detection

Euclidean distance based sensing technique is described in [66]. It works by calculating the euclidean distance between the auto correlation of the received signal and with that of a reference signal or reference line. The euclidean distance D is calculated by :

$$D = \sqrt{\sum (R_{x,x}(\tau) - R)^2}$$

where, the reference line calculation is described in [67,68] and $R_{x,x}(\tau)$ is the auto correlation of the received signal. Euclidean distance is compared with a threshold to detect the presence or absence of signal. This distance based sensing is reported to be more efficient than the auto correlation methods of sensing.

2.2.3 Wavelet based Detection

Wavelet based sensing or edge detection is a continuous wavelet transform in which signal is decomposed into coefficients with the help of a basis [69]. For a given signal $s(t)$, the continuous wavelet function $\psi(t)$ is given by :

$$g(u,v) =< x(t), \psi_{u,v} >= \int_{-\infty}^{\infty} x(t)\psi^*_{u,v}(t)dt$$

where, u is the scaling parameter, v is the translating parameter, and $\psi_{u,v}(t)$ is the basis. This transform maps one dimensional signal into two dimensional coefficients $g(u,v)$. This can help in frequency-time analysis with frequency corresponding to parameter v, and the time instant corresponding to parameter u. The wavelet based sensing is done by calculating the power spectral density using computation of continuous wavelet transform. The local maxima of the power spectral density is compared to the threshold to decide about the spectral occupancy.

Wavelet based sensing easily distinguishes between the signal and the noise while deciding about the spectrum occupancy. It is used for cases where sensing over a wideband is required for multi channel communication. While, it is very effective for wideband multi channel use cases, it needs large processing time.

2.2.4 Matched Filter based Detection

Matched filter based detector works on the principle of maximizing the SNR at the output of the detector. This method is best suited, where the receiver has prior knowledge about the input signal. The prior knowledge is provided to secondary user in form of pilot signal or PN sequence. The matched filter performs convolution of the received signal with the already known pilot signal at the receiver. The matched filter based detection is prone to errors due to frequency mismatch and phase noise. To improve the performance of match filter based detection due to frequency errors, a hybrid match filter technique has been proposed [70].

2.2.5 Energy Estimator based Detection

Energy estimator uses the sampled output of the ADC to calculate the energy of the received signal which is then compared against a threshold to declare the presence or absence of a signal. Energy can be estimated by calculating the square of the magnitude of the FFT bins and comparing them against the threshold. The other method of energy estimation is by calculating the square of the sampled output of ADC averaged over fixed number of samples. The output of the detector can be given by:

$$E_d = \sum_{n=0}^{N} y^2(n)$$

The energy of the received signal is compared with that of the noise level for estimating the presence of signal. The energy based sensing is easier to implement but poses major challenge in case of time varying noisy conditions. The threshold for energy detector is an important parameter and poor estimate of threshold results into significant performance degradation. In order to minimize the sensing error due to estimation error of the noise power, a dynamic estimation of the noise power is recommended in [71].

All the above techniques of spectrum sensing are well defined for Nyquist based signal processing. In CS, the measurement samples are compressed. Therefore, these techniques can not be applied directly to CS measurements. An intermedi-

ate signal processing step is required for conditioning of the measurement samples. One such algorithm called Back-DSP for the popular MWC framework is proposed in [57] to condition the compressed measurement samples before Nyquist processing. Alternatively, there are specialized signal processing algorithms proposed for a given architecture as described in [65].

In this work, along with other design changes, it is proposed to use energy based signal presence detection technique to convert the MWC architecture to discretized CS framework [41] as explained in Chapter 4.

Summary: A brief review on some popular CS architectures and signal presence detection techniques are included in this chapter. The MWC architecture is found to be a realizable CS framework for spectrum sensing using available analog hardware components.

The next chapter provides a detailed analysis on MWC implementation with its advantages and disadvantages for spectrum sensing and wideband applications. It also covers the modified version of MWC and its performance analysis.

CHAPTER 3

A Detailed Review of MWC Architecture

This chapter discusses the MWC architecture for compressive sampling. MWC overcomes the shortcomings of CS architectures such as multicoset sampling, which suffers with the limitation of input analog bandwidth. MWC is different from multicoset sampling in the process of aliasing. In multicoset, aliasing is due to under-sampling whereas in MWC, aliasing is due to modulation of the input signal with random periodic sequence before sampling operation. It also uses multiple parallel modulating branch to achieve wideband operation suitable for analog multiband signals. It has several potential advantages for CS hardware realisations such as (i) analog mixers are proven technology (ii) sign alternating mixing sequences can be implemented by standard digital circuit (iii) sampling is synchronized in all channels without need of additional time shifts and (iv) reduced sampling rate requirement with respect to Nyquist for wideband applications.

In this chapter, a detailed analysis of MWC framework starting with signal reconstruction, noise performance analysis, modified MWC architecture and their advantages are provided. An application of interference detection and filtering using MWC is also included. The problem of noise enhancement and non-uniform treatment of input bands in MWC are explained with simulation results, which is the prime motivation for the development of the new CS architecture.

3.1 Signal Reconstruction in MWC

The MWC hardware architecture is shown in Fig. 2.3. The MWC input is an analog signal. The sampled output of MWC is a vector of size $m \times 1$, where, m is number of parallel branches. Since, the input to MWC is an analog signal, the signal reconstruction setup becomes an Infinite Measurement Vector (IMV) setup. First, the IMV measurement setup is converted to MMV using Continuous to Finite (CTF) transformation proposed by Mishali et.al., in their works [17, 38] and [72]. In CTF, we compute frame $\mathbf{Q}$ from time domain measurements as given by

$$\mathbf{Q} = \int_{f \epsilon F_s} \mathbf{y}(f)\mathbf{y}^H(f)df = \sum_{n=-\infty}^{\infty} \mathbf{y}[n]\mathbf{y}^T[n] \tag{3.1}$$

where, $\mathbf{y}[n] = [\mathbf{y}_1[n], \mathbf{y}_2[n], ..., \mathbf{y}_m[n]]^T$ is the vector of samples at time instances nT_s. Then, any matrix $\mathbf{V}$, for which $\mathbf{Q} = \mathbf{V}\mathbf{V}^H$ is a frame of $\mathbf{y}(F_s)$ and its column span is equal to that of $\mathbf{y}(f)$. It is also shown that vector $\mathbf{U}$ obtained by solving $\mathbf{V} = \mathbf{A}\mathbf{U}$ exhibit same support as that of the input signal [73]. The decomposition of Q to obtain V is based on eigen decomposition.

Therefore, finding support in MWC is a MMV problem. Using the frame of samples in time domain, we compute the active support set (S) of input signal using standard CS recovery methods. The knowledge of active support is then used to determine the updated sensing matrix $\mathbf{A_S}$, which is made up of columns of $\mathbf{A}$ corresponding to active support set S. The product of pseudo inverse of updated sensing matrix ($\mathbf{A_S}$) and measurement sample vector $\mathbf{y}[n]$ provides the reconstructed input signal.

The above reconstruction method is used in MWC framework and all its variants for signal recovery. The reconstructed signal with input is therefore used for MSE calculations for architecture performance analysis.

By architectural design, MWC has been robust and realizable sensing architecture for wideband communication. However, *one of the major challenges faced by MWC is design of m mixing PN sequences of length L*. In many applications, the value of L is small and the number of required parallel branches, m ($\geq 4k$; where, k is

sparsity), is high. This is a contrasting requirement and often difficult to achieve. Next, we describe the modified MWC architecture, which needs only one mixing sequence in place of m different mixing sequences without having a performance penalty.

3.2 Modified MWC Architecture

To address the challenge of designing m different mixing sequences of length L in MWC, a new modified MWC framework with single mixing sequence has been proposed [74]. The modified MWC has only one mixer and a filter bank with m sub-filters and m low rate ADCs as shown in Fig. 3.1.

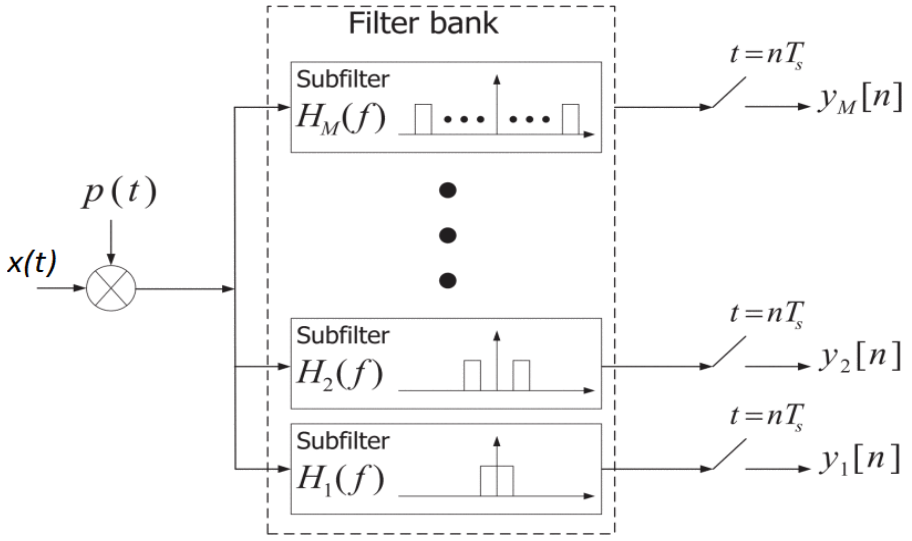


Figure 3.1: Block diagram of modified MWC architecture.

The modified architecture uses a filter bank with pass-band of each branch shifted by B, where B is the maximum bandwidth of any single channel of multi-band input signal. Each parallel branch filters a unique aliased version of input bands. This modified architecture has implementation advantages. Due to the use of only one mixing sequence (PN sequence generator) and single mixer, the complexity of analog circuit design and the effort for synchronization between m branches are significantly reduced. Moreover, it is observed that the equivalent measurement matrix of the proposed CS implementation follows a Toeplitz structure [74]. As a result, the matrix computations also gets simplified. Since, the

requirement is getting reduced to one mixing sequence, the use of the best mixing sequence is possible. The proposed system is reported to provide reconstruction guarantee similar to MWC.

The modified sensing matrix A for a typical case of $m = 3$ and $L_0 = 3$ (where, $L = 2L_0 + 1$; L is the length of mixing sequence), takes the form as given by (3.2),

$$\mathbf{A} = \begin{bmatrix} c_{-3} & c_{-2} & c_{-1} & c_0 & c_1 & c_2 & c_3 \\ 0 & c_{-3} & c_{-2} & c_{-1} & c_0 & c_1 & c_2 \\ 0 & 0 & c_{-3} & c_{-2} & c_{-1} & c_0 & c_1 \end{bmatrix} \tag{3.2}$$

where the matrix coefficients c_i are obtained by performing the Fourier transform of the α_k (piece-wise constant values of periodic mixing sequence), i.e.

$$c_i = \frac{1}{M} \sum_{k=0}^{M-1} \alpha_k e^{j\frac{2\Pi}{M}ik}; \quad -L_0 \leq i \leq L_0 \tag{3.3}$$

Therefore, the storage requirement is reduced to $\frac{1}{2M}$ of that in MWC. A simulation based recovery performance comparison of MWC and the modified MWC is carried out by calculating MSE. A multiband signal with three occupied bands containing a modulated signal of bandwidth B is used for MSE calculation. The MSE comparison is shown in Fig. 3.2.

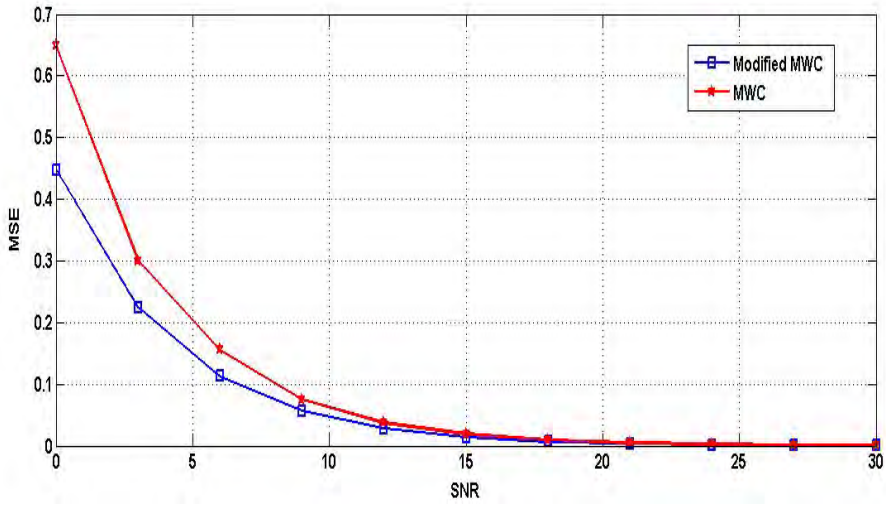


Figure 3.2: MSE comparison for MWC and modified MWC.

The modified MWC architecture with single mixing sequence offers similar reconstruction as compared to standard MWC. This architecture needs only one mixing sequence which is easy to construct. Alternatively, to address the need and construction of multiple mixing sequences, *another variant of MWC architecture is also proposed, which uses deterministic construction methods of mixing sequences [75].*

A simulation based performance verification is carried out for MWC receiver with different types of mixing sequences. A performance comparison of MWC at different SNR with different types of mixing sequences are shown in Fig. 3.3. From the simulation, it is seen that Legendre Sequence performs better at low SNR conditions. In high SNR conditions all the sequences have similar reconstruction performance.

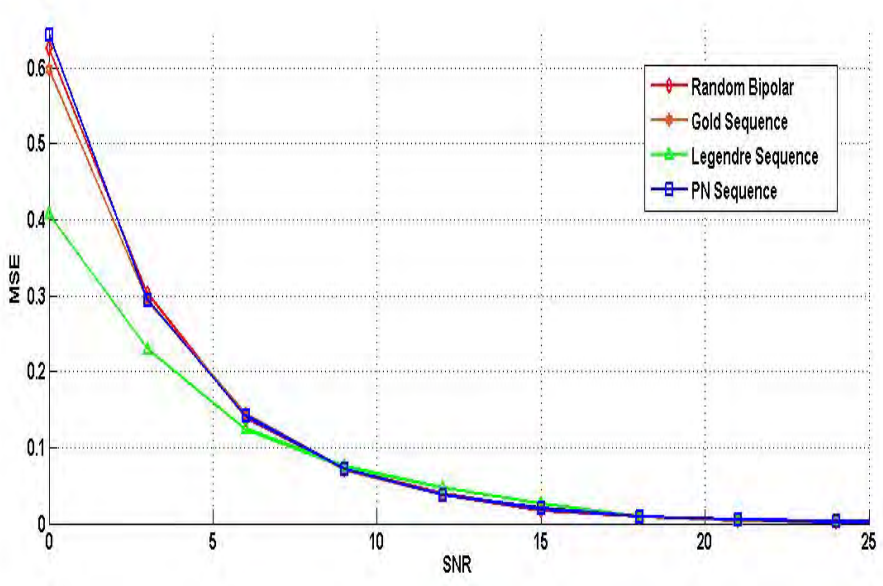


Figure 3.3: MSE comparison for different mixing sequence in MWC.

In next section, a MWC based CS receiver for interference detection and filtering is proposed. The proposed CS framework provides significant processing advantage for such applications.

3.3 MWC Receiver for Jamming Detection and Filtering

There are instances of unauthorized spectrum access and transponder jamming by injecting narrowband interferer [76, 77]. Fixed spectrum allocation policies often leads to sparsity of channel occupancy in time and frequency domain. This frequency domain sparsity of channels makes CS based receiver suitable candidate for implementation. MWC based CS receiver is proposed to address the problem of interference detection and filtering in satellite transponder.

Application Setup: This application assumes that a satellite transponder is channelized and very few channels are active at any given point of time leading to sparsity in frequency domain. In each channel, there are one or more valid users with similar waveform for communication. The objective is to identify the presence of jamming signal (high power tone-jammer) or interfere in real-time and notch it out before the output stage of transponder, without affecting the other communications from valid users.

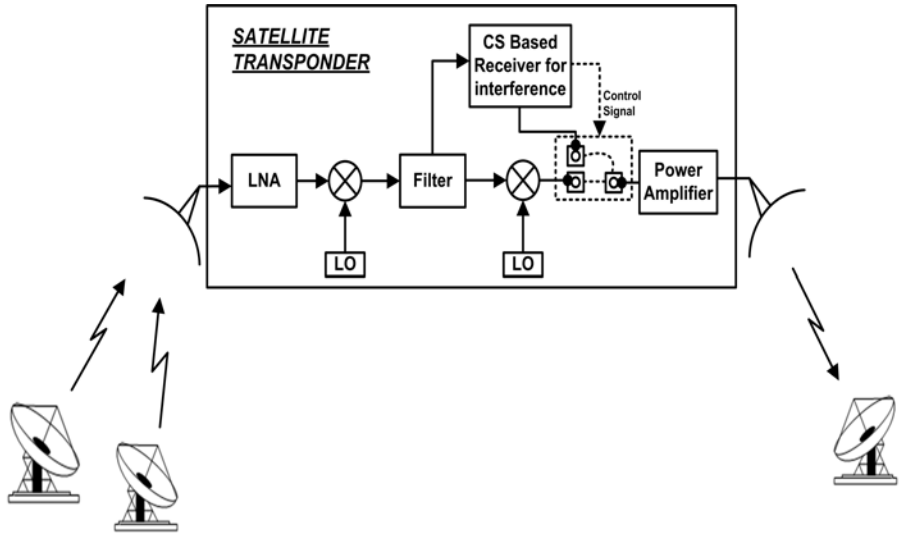


Figure 3.4: Block diagram of overall system.

System Description: The block diagram of proposed receiver is shown in Fig. 3.4. In the proposal, a MWC based CS receiver is used for real-time monitoring of presence of interfere in satellite payload. The block diagram of MWC based

CS receiver is shown in Fig. 3.5. During operation, when the interfere presence is detected by CS receiver, it switches the input of Power Amplifier (PA) to the filtered output coming from the CS receiver. The signal processing in receiver is done in a way that no other ongoing communication in that transponder gets affected.

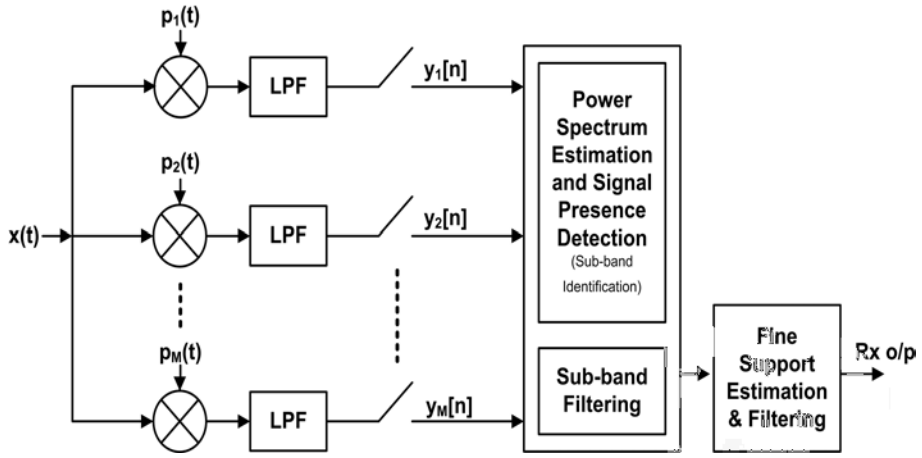


Figure 3.5: Block diagram of CS receiver for jamming detection and filtering.

Signal Processing steps of CS Receiver:

1. The receiver estimates the input signal power spectrum using compressed output samples of MWC and a computationally light algorithm [65]. According to theory, the Fourier transform of auto-correlation function of a signal $x[n]$ gives its power spectral density.

$$p_x[w] = \sum_{-\infty}^{\infty} r_x[n] e^{-jnw}, \quad 0 \leq w \leq 2\pi \tag{3.4}$$

where, $r_x[n]$ is the auto correlation function of the samples $x[n]$ given by $r_x[n] = E[x[m]x[m-n]]$. The cross correlation between output samples of MWC (i.e. between $y_i[k]$ and $y_j[k]$) along with deterministic cross correlation between $c_i[n]$ and $c_j[n]$ (coefficients of mixing sequences), can be used to determine the auto correlation function $r_x[n]$ of input signal. The relationship between $\mathbf{r}_x$ and $\mathbf{r}_y$ is expressed as [65]

$$\mathbf{r}_y = \mathbf{R}_c \mathbf{r}_x \tag{3.5}$$

Solving (3.5), using Least Square method to obtain $r_x[n]$ and in-turn $p_x[w]$. Thus, the input power spectrum is estimated using compressed domain samples. Alternatively, the power spectrum can also be estimated by applying conventional pwelch method to MWC output samples after necessary conditioning to bring the samples in Nyquist domain [57].

2. The power spectrum of input signal is compared against a threshold based on spectrum allocation plan of service provider and regulations [78], to identify the presence of an interfere (Jamming signal).

3. Once the presence of interfere is identified, the corresponding sub-band or channel is filtered [43].

4. The samples of filtered sub-band are then conditioned suitably for Nyquist processing [57] and Frequency Difference Detection techniques (Analog Doubling Automatic Frequency Detector is used in this case) is applied to find accurate location of interference within that band or channel. Here, assumption is that any channel can have multiple narrowband communication carriers. The FDD technique utilizes the centre frequency of the band or channel as initial coarse estimate and resolves it further to estimates the fine support of the jamming signal. This is possible as amplitude of the jamming signal is high compared to that of other carriers. A discrete version of Analog Doubling Automatic Frequency Detector (AD-AFC) is used for faster convergence and better tracking accuracy [45].

5. The interferer is then filtered out using notch of fixed width.

***Simulation and Result**:* The MWC receiver is simulated with m=24 parallel branches, considering maximum of six (6) active bands at any point of time. The multiband input signal x[n] with three (3) active channels, one with tone-jammer, are simulated. The total signal bandwidth of 5 GHz with 100 channels (maximum individual channel bandwidth of B = 50 MHz) is considered. A user defined PFD threshold is used for jamming signal presence detection. After interfere location identification, a notch filter with fixed Q-factor (= 10) is used for filtering of the jamming signal.

The estimated power spectrum of the input signal obtained by MWC based receiver is shown in Fig. 3.6. The receiver output spectrum after interference (jamming signal) detection and filtering is shown in Fig. 3.7. The output shows that other ongoing communication within the transponder remains unaffected while jamming signal is filtered successfully.

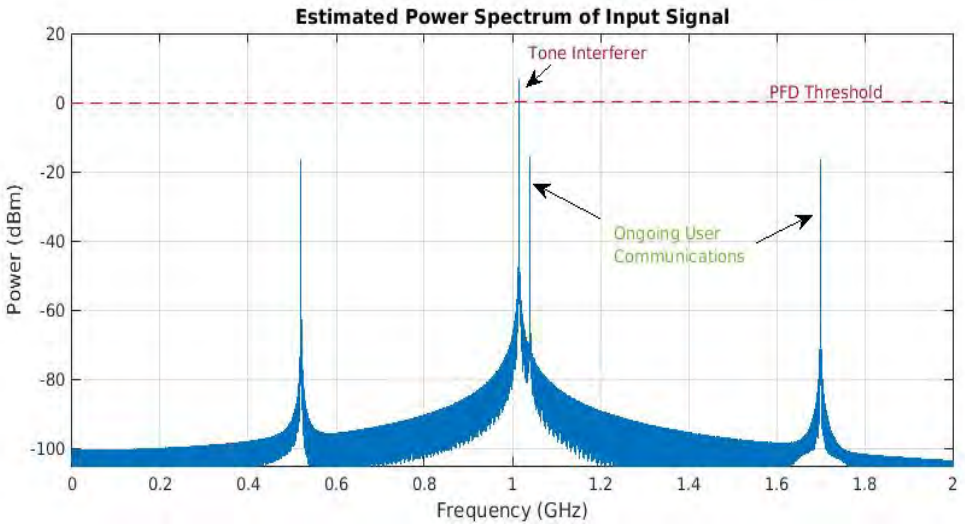


Figure 3.6: Estimated power spectrum of input with jamming signal.

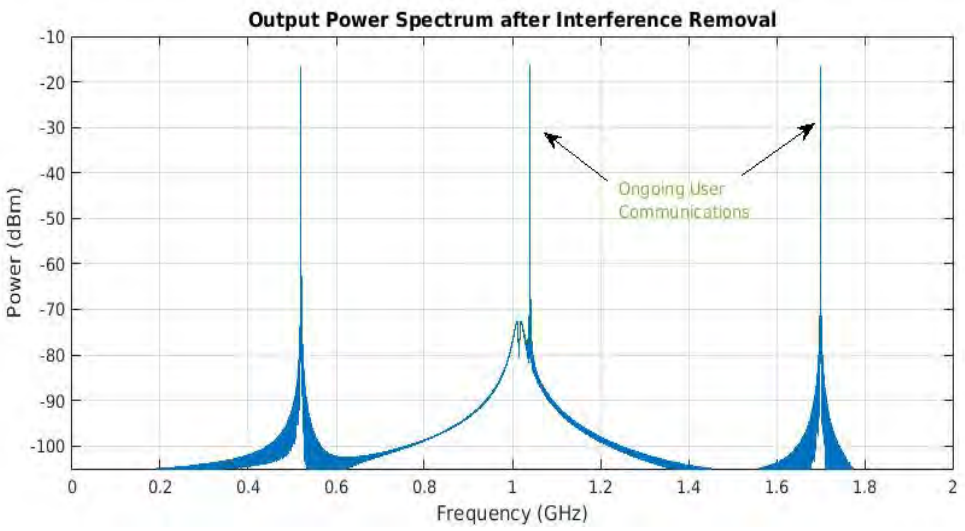


Figure 3.7: Power spectrum of receiver output signal after filtering.

The above simulation work uses less than fifteen (15) percent Nyquist samples for blind interference detection and filtering task. This offers significant reduction in processing load making CS based receiver suitable for wideband signal processing.

The noise performance analysis of MWC system together with generic CS framework is provided in next section.

3.4 Noise foldover and its characterization in CS

The CS receivers encounters two types of noise in the system, the measurement noise **w** due to sensing hardware and the channel noise (**z**) added with the signal **x** ($\in R^N$) at the input to the sensing device. The CS receiver is represented by sensing matrix **A** ($\in R^{MxN}$; $M \ll N$). The sensing system therefore, can be expressed as given by (3.6).

$$\mathbf{y} = \mathbf{A}(\mathbf{x} + \mathbf{z}) + \mathbf{w} \tag{3.6}$$

where, **z** represents the channel noise vector ($\in R^N$) and assumed to be additive white Gaussian noise with zero mean. The measurement noise **w**, due to sensing hardware is also assumed to be white noise with zero mean [79]. Assume, that noise vector **w** has covariance $\sigma^2\mathbf{I}$ and **z** has covariance $\sigma_0^2\mathbf{I}$, which is independent of **w**. Then (3.6) in equivalent form is written as

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{v} \tag{3.7}$$

where, **v** is defined by $\mathbf{v} = \mathbf{w} + \mathbf{A}\mathbf{z}$. Under the assumption of white noise, the effective noise vector **v** has covariance **Q**, which is given by

$$\mathbf{Q} = \sigma^2\mathbf{I} + \sigma_0^2\mathbf{A}\mathbf{A}^\mathbf{T} \tag{3.8}$$

This shows that the noise vector **v** expressed in (3.7), no longer remains white. In case if $\mathbf{A}\mathbf{A}^\mathbf{T}$ becomes proportional to identity matrix **I**, then only **v** can still be considered as white noise. Assume **A** as a concatenation of r orthonormal bases, i.e. $\mathbf{A} = \left[A^{(1)}.....A^{(r)}\right]$. Where, $r = \frac{N}{M}$ = compression ratio. Here, each $A^{(k)}$ is an $N \times N$ orthogonal matrix.

$$\mathbf{A}\mathbf{A}^\mathbf{T} = [\mathbf{A}(\mathbf{1})\mathbf{A}(\mathbf{1})^\mathbf{T} + .. + \mathbf{A}(\mathbf{r})\mathbf{A}(\mathbf{r})^\mathbf{T}] = r\mathbf{I} = \frac{N}{M}\mathbf{I} \tag{3.9}$$

In this case the variance of noise vector $\mathbf{v}$ becomes $\mathbf{Q} = \gamma\mathbf{I}$ and $\gamma = \sigma^2 + \frac{N}{M}\sigma_0^2$ [80]. The noise variance has increased by a factor of $\frac{N}{M}$. This is called noise fold-over. However, in this case the noise becomes white and all recovery method works well. Noise whitening is also achieved by multiplying the linear system by $\mathbf{Q_1}^{\frac{-1}{2}}$, where, $\mathbf{Q_1} = \frac{\mathbf{Q}}{\gamma}$. This whitening results into an equivalent system, given by

$$\mathbf{y} = \mathbf{B}\mathbf{x} + \mathbf{u} \tag{3.10}$$

where,

$$\mathbf{B} = \mathbf{Q_1}^{\frac{-1}{2}}\mathbf{A};\ \ \mathbf{u} = \mathbf{Q_1}^{\frac{-1}{2}}\mathbf{v}$$

Now, the noise vector $\mathbf{u}$ is white, similar to the case of $\mathbf{A}\mathbf{A}^\mathbf{T}$ being proportional to identity. However, the main difference is that whitening has changed the measurement matrix from $\mathbf{A}$ to $\mathbf{B}$. It has been shown that new matrix $\mathbf{B}$ also satisfies RIP property. If $\mathbf{A}$ satisfies the RIP of order k with constants $0 \leq a_1 \leq b_1$, then matrix $\mathbf{B}$ also satisfies the RIP of order k with constants $a_1(1-\alpha_1)$ and $b_1(1+\alpha_1)$ [81]. Here, $\alpha_1 = \frac{\alpha}{1-\alpha}$, α is the measure of quality of approximation of $\mathbf{A}$ as concatenation of orthonormal bases [81]. Noise whitening is used in many CS applications.

The above analysis shows that CS framework has inherent noise foldover. It is a challenging problem in CS based receivers for wideband communication applications. In MWC architecture, the periodic random mixing sequence results into noise enhancement [82]. This is primarily due to the non-zero DTFT coefficients of mixing sequences in each branch. As noted in earlier section that the diagonal matrix $\mathbf{D}$ in the sensing matrix contains DTFT coefficient of periodic mixing sequence which determines the amount of aliasing in MWC. This aliasing can be controlled, if the contribution of these DTFT coefficients could be limited to few bins. Within MWC architecture, pulse shaping of mixing sequence with design of different pulse shaping functions [46] were carried out to study the effect of shaping on recovery under different noise conditions.

Another disadvantage of MWC is that the shape of analog periodic mixing sequence, which results in the sampled response of each branch weighted by different coefficients in the spectral down-conversion process. The rectangular

pulses shape of the analog mixing signal generates $\sin(x)/x$ profile (waveform) in the frequency domain (this is captured by diagonal matrix **D** of the sensing matrix([17]). The null of this frequency-domain pulse shape coincides with the edge of the wide band W that is sensed. Effectively, the rectangular pulse shape causes a low-pass filtering effect, thereby penalizing the higher frequency subbands in the signal reconstruction.

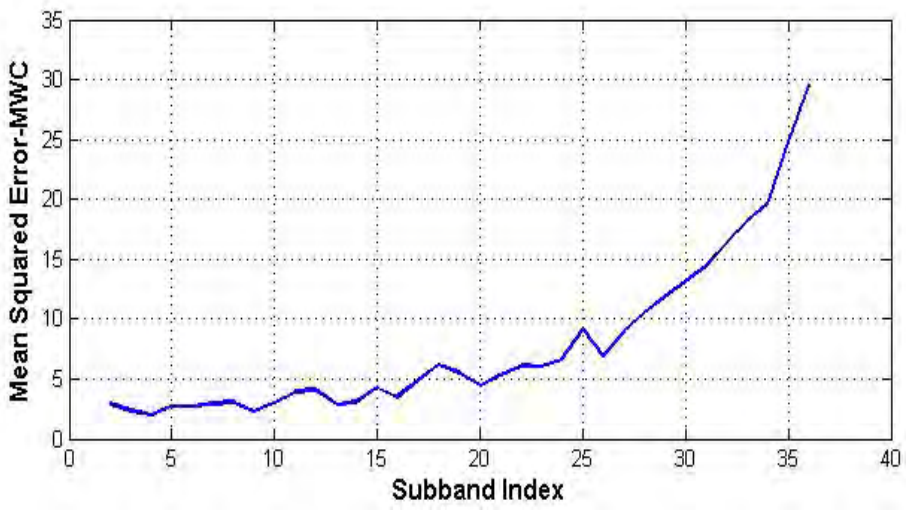


Figure 3.8: MSE vs band number plot of MWC architecture.

The effect of rectangular mixing sequence has been simulated by calculating the MSE of the input and reconstructed waveform after placing them in different sub-bands. The plot of MSE vs Band number for MWC is shown in Fig. 3.8. The MSE increases towards the band edge. The simulation was performed with 36 bands. It may be noted that the pseudoinverse of the MWC measurement matrix **A** greatly amplifies the noise at the high-frequency subbands in attempting to equalize the sinc shaped weighting of the coefficients.

Summary: In this chapter the performance of MWC and its variants is described. MWC architecture leads to real sensing matrix resulting into large noise fold-over problem. MWC also suffers from the non uniform performance of input subbands.

Recently, researchers started use of Binary matrix for CS systems. These matrix have deterministic construction methods and offer definite recovery guarantee [83–85]. It is well established that the binary LDPC matrix under l_1 minimization

performs well for CS implementation [86].

In next chapter, a regular binary LDPC matrix based novel CS architecture (motivated by MWC framework) is proposed, which is suitable for implementation using available ADCs. In the proposed architecture, the use of w_r (equals to number of ones in a row of LDPC matrix) complex sinusoid is proposed instead of the rectangular mixing sequences of MWC. This has ensured the uniform treatment of all bands and their DTFT coefficients are limited to w_r bins, thereby, reducing noise foldover.

CHAPTER 4

Proposed CS Architecture and System Model

In this chapter, we propose a CS architecture motivated by MWC, with a novel approach towards reducing the noise foldover and to achieve fairness to all input bands unlike in MWC. This has been achieved by modifying the architecture to provide discrete measurements, thereby, allowing signal processing techniques such as message passing algorithms to be used for sparsity detection resulting in robust signal recovery.

The CS architecture is an analog front-end implementation of sparse binary regular LDPC matrix. This chapter describes the signal model and architecture details, followed by the theoretical bound on size of the measurement matrix (equivalent hardware requirement) for given sparsity of input signal. It also describes the two message passing algorithms motivated by Sudocode and Gallager (hard decision LDPC decoding), for sparsity detection followed by noise performance analysis of the proposed architecture for wideband communication.

4.1 Signal Model

The proposed signal model considers the input to the sampling device as a complex-valued continuous-time multiband signal, $x(t) = s(t) + n(t)$, bandlimited to $W = 1/T_W$. The Fourier Transform of $x(t)$ is nonzero over a frequency range of $\left[-1/2T_W, +1/2T_W\right]$. Signal $x(t)$ comprises of a variable number (k) of narrowband signals, each of bandwidth $B \ll W$. It is assumed that $x(t)$ is spectrally sparse. Denoting the ratio W/B as N (i.e, N is the number of subband locations within W in which the narrowband signals can reside), the spectrum sparsity assumption

Figure 4.1: Model of spectrally sparse multiband signal (maximum subband bandwidth = B and total bandwidth = W).

translates to $k \ll N$. Here, p denotes the ratio k/N, and note that as $N \to \infty$, the ratio p approaches the probability that an element of vector $\mathbf{x}(f)$ is nonzero. A schematic spectral diagram that illustrates this spectrally sparse nature of $x(t)$ is shown in Fig. 4.1. The goal of the sampling mechanism in this scheme, as in [17,38,56,87], is to sample this wideband signal using a bank of analog mixers, analog filters, and ADCs (analog to digital converters) with a total of L branches, where each ADC operates only at a rate comparable to B. Furthermore, this saving in the required ADC sampling rate (from W to an aggregate of LB) is to be achieved without prior knowledge of the occupied spectral locations.

4.2 Proposed CS Architecture

There are several CS hardware architectural approaches proposed in the literature; specifically, random filtering [88], random convolution [89], random demodulator [56], multicoset sampling [38], and modulated wideband converter or MWC [17]. A unifying concept behind these various scheme is to allow different subbands to overlap with each other. The resulting aliasing is introduced in a controlled manner and it is removed by subsequent algorithmic processing.

The CS architecture proposed in this section is based on the use of a sparse binary sensing matrix $\mathbf{A}$ (this is unlike the earlier approaches [17,56] in which all $M \times L$ elements of matrix $\mathbf{A}$ are nonzero and non-binary). Similar to the construction of LDPC parity check matrix $\mathbf{H}$, our proposed sensing matrix has w_r number of ones randomly placed in each row and w_c number of ones in each column at random locations. The advantage of using a sparse matrix $\mathbf{A}$ is that it causes aliasing (overlapping in frequency domain) of only w_r slices of $X(f)$, thereby reducing the severity of noise fold-over phenomenon that affects the MWC. Another reason

for preferring a sparse matrix is that it holds Restricted Isometry Property (RIP) (and low mutual coherence), which allows easy recovery of sparse signals [90]. The implementation of **A** could be achieved as an analog front-end architecture

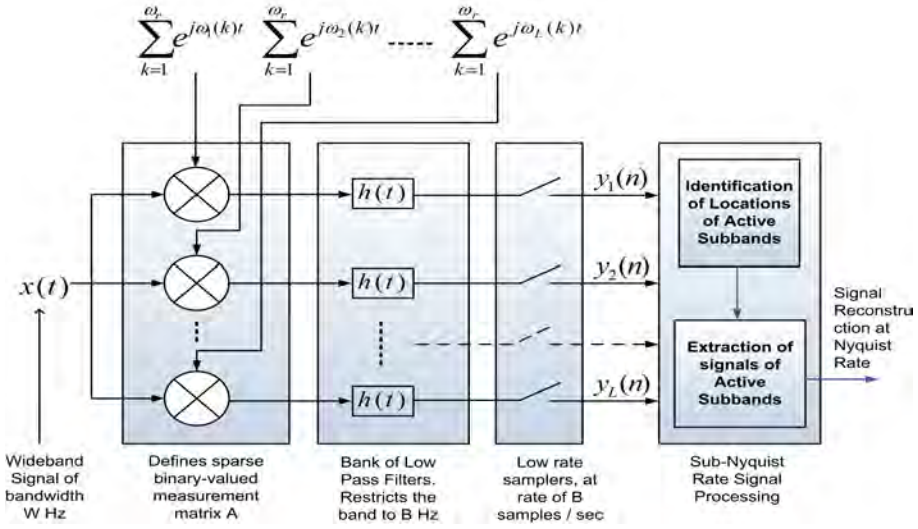


Figure 4.2: Hardware architecture of the proposed CS framework.

as shown in Fig. 4.2, which is similar to that in MWC [17]. Each branch has a front-end mixer stage, whose functionality is that of a group of w_r local oscillators which brings w_r different slices $x_\ell(f)$ of input signal to the baseband. The mixing signals are complex exponentials, $\sum_{r=1}^{w_r} \exp(j2\pi f_{LO}^{m,r} t)$, where $f_{LO}^{m,r}$ is r^{th} mixing frequency on m^{th} branch. The $f_{LO}^{m,r}$ of mixing frequencies are set as per the locations of ones in matrix **A**, and the indices of a total of w_r slices $x_\ell(f)$ that are down-converted in each of m^{th} branch. The construction of analog mixing sequence is explained at a later stage.

As in MWC scheme, signal at the output of the mixer stage in each of M branches is passed through an analog low pass filter (LPF) with bandwidth B (with impulse response $h(t)$) as shown in Fig. 4.2. The complex output of filter is sampled at a rate of $1/B$. In our proposal for CS framework, this sampled stream is sent to an energy detector. Assuming an ideal energy estimator and uniform power spectral densities in all the k occupied bands, the normalized estimator output y_m is an integer in the range $0, 1, \ldots, w_r$. The y_m is equal to the number of nonzero signal bands that coincide with the frequencies of w_r complex exponential comprising the mixing signal in m^{th} branch. In practice, this ideal energy

estimation is not achieved and the normalized estimator has to be modeled as a continuous random variable with a mean that equals an integer from 0 to w_r and a variance that is determined by the duration of energy estimation.

The output of energy estimator is a $M \times 1$ vector which is given as input to the belief propagation based message passing (MP) algorithms like Sudocode MP [91] and Gallager MP (the one developed based on Gallager LDPC decoding technique). The algorithm does the sparsity detection, which inturn provides the location of active sub-bands of input signal. These active sub-band locations information is used to select the columns of sensing matrix for signal reconstruction. In the proposed CS framework, signal is reconstructed using the derived sensing matrix together with sampled output of LPF. The signal processing steps for sparsity detection and signal reconstruction are shown in Fig. 4.3.

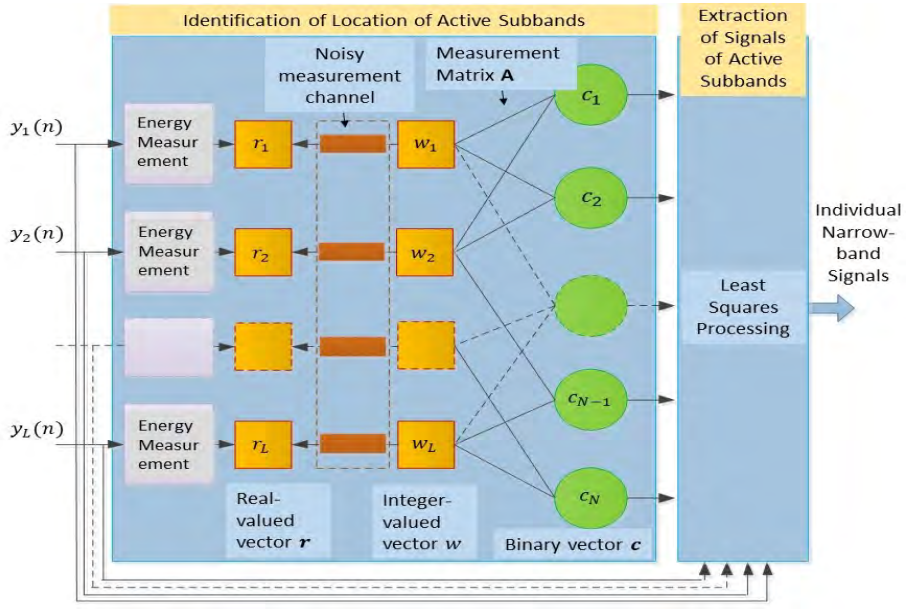


Figure 4.3: Signal processing steps for signal reconstruction of proposed CS framework.

4.2.1 Design of Analog Mixing Waveforms

The mixing waveform $p_\ell(t)$ used in ℓ^{th} branch (shown in Fig. 4.2) has a period of $T_B = 1/B = NT_W$. Its Discrete Fourier Series takes the following form:

$$p_\ell(t) = \sum_{k=0}^{w_r-1} \exp(2\pi m_\ell(k)Bt)$$
$$P_\ell(\omega) = \sum_{k=0}^{w_r-1} \delta(\omega - \omega_\ell), \qquad \text{where } \omega_\ell = 2\pi m_\ell(k)B$$

Thus, the mixing waveform, a combination of complex sinusoidal (continuous waveform or CW) signals at w_r different harmonics of B, acts as a combined w_r complex-valued local oscillators (LOs). The set of harmonic multipliers $\{f_\ell(k) = m_\ell(k)B\}, k = 0, \ldots, w_r - 1$ defines the spectral locations of $w_r \ll N$ LOs within 0 to W Hz on ℓ^{th} hardware branch ($1 \leq \ell \leq L$).

Direct Digital Synthesis

One of the methods for the generation of analog mixing waveforms $\{p_\ell(t)\}$ is the Direct Digital Synthesis (DDS).

A major concern with the DDS of these waveforms is a maximum frequency of W that needs to be synthesized. The solution that we propose is as follows. When the generated signal is in the discrete-time (D-T) domain, it is periodic in the frequency-domain, with a period (Hertz) equal to the sample rate F_S^ℓ used on ℓ^{th} branch. A D-T signal $\tilde{p}(t)$ generated at one frequency, $k \times B$, where k is an arbitrary integer, at a sample rate F_S has the following frequency-domain (DTFT) representation:

$$\tilde{P}(f) = \mathcal{F}(\tilde{p}(t)) = \sum_{u=-\infty}^{\infty} \delta\left(f - \left(k\left(\frac{B}{F_S}\right) - u\right)F_S\right)$$

In the conventional discrete-to-analog time converters, only the base or the primary period of this signal, within $0.5 \times [-F_S, F_S]$ is retained by passing the D-T signal through a low-pass filter. For our purpose, we enlarge the cutoff frequency of this filter to $v \times [0, F_S]$, where v is a positive integer, so that the filter output

encompasses multiple adjacent periods besides the base period. The aliases of the base frequency kB can be included in the analog filter bandwidth as shown in Fig. 4.4. We require these aliased frequencies to be multiples of B. Specifically, for some integers k, q, we require $kB + F_S = qB$.

The required sampling frequency corresponding to a chosen value of integer $q > k$ is determined as $F_S = (q - k)\, B$.

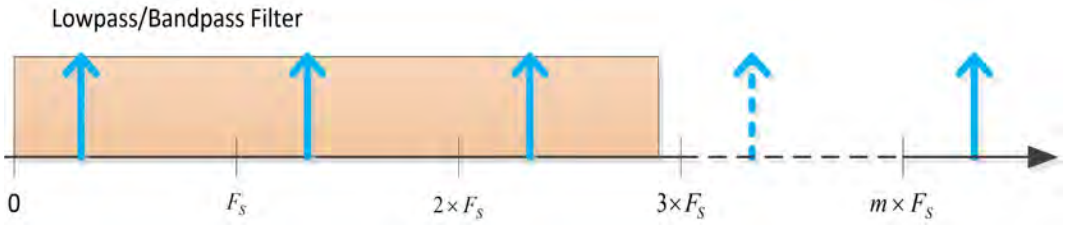


Figure 4.4: An approach for direct digital synthesis of the proposed analog mixing signal.

Analog Domain Synthesis

In this method, whose schematic diagram is provided in Fig. 4.5, a reference frequency of B Hertz is input to a comb generator [92–94] to produce a signal comprising of all the harmonics of B over a wide range of frequencies. This signal is fed to a bank of tunable bandpass filters centered at the desired harmonics of B (i.e., those in the set $\{f_\ell(k) = m_\ell(k)\}_{k=1}^{w_r}$) in the synthesis of $p_\ell(t)$ [95–97]. The output of these filters are given to a combiner to generate the desired mixing signal $p_\ell(t)$.

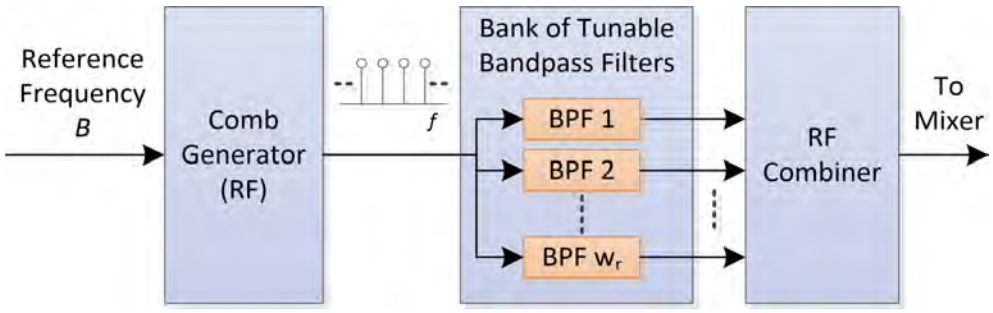


Figure 4.5: Analog mixing signal generation by a comb generator.

In the cases where the bandwidth B is very small, the realization of filter bank becomes challenging. For those cases, an alternative scheme of analog-domain

synthesis is to use bank of Phase Locked Loop (PLL) circuits fed with a common reference (B), which generates programmable multiples of B Hertz. The output of bank of PLLs are combined for the desired set of waveforms $\{p_\ell(t)\}$ for $1 \leq \ell \leq L$. As w_r is a small number, this option turns out to be easy and realisable.

4.2.2 Model of Discrete-Time Measurements

In this section, we provide a mathematical description of the effect of mixing the incoming wideband signal $x(t)$ with the bank of analog mixing waveforms $\{p_\ell(t)\}$.

The Fourier Transform of the analog product $\tilde{x}_\ell(t) = x(t)p_\ell(t)$ on ℓ^{th} hardware branch can be formulated as follows:

$$\begin{aligned}
\tilde{X}_\ell(f) &= \int_{-\infty}^{\infty} \tilde{x}_\ell(t) \exp(-j2\pi ft)\, dt \\
&= \int_{-\infty}^{\infty} \left(\sum_{k=0}^{w_r-1} \exp(2\pi f_\ell(k)t) \right) x(t) \exp(-j2\pi ft)\, dt \\
&= \sum_{k=0}^{w_r-1} \int_{-\infty}^{\infty} x(t) \exp(-j2\pi(f - f_\ell(k))t)\, dt \\
&= \sum_{k=0}^{w_r-1} X(f - f_\ell(k))
\end{aligned}$$

In a hypothetical noiseless scenario, the above sum, at a frequency f, contains at the most w_r nonzero terms, which occurs when all w_r band slices (subbands) $X(f - f_\ell)$ are nonzero [17].

The mixer output signal on each branch is passed through an ideal analog lowpass filter of bandwidth B and impulse response $h(t)$ as shown in Fig. 4.2. The filter output is converted to discrete time (D-T) at a sample duration of $T_B = 1/B$ seconds. The DTFT of the D-T signal $y_\ell(n)$ on ℓ^{th} branch is given as follows:

$$\begin{aligned}
Y_\ell(f) &= \sum_{n=-\infty}^{\infty} y_\ell(n) \exp(-j2\pi fnT_B) \\
&= \sum_{k=0}^{w_r-1} X(f - m_\ell(k)B), \quad |f| \leq B/2
\end{aligned} \tag{4.1}$$

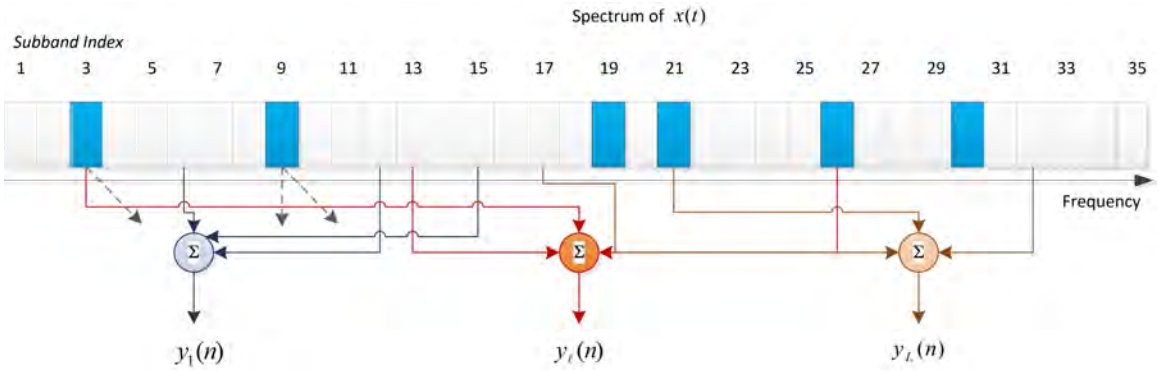


Figure 4.6: Pictorial representation of effect of low density measurement matrix ($w_r = 3, w_c = 2$ in this illustration).

Above equation can be written in a matrix form as follows:

$$\mathbf{y}(f) = \mathbf{A}\mathbf{x}(f) = \mathbf{A}\left(\mathbf{s}(f) + \mathbf{n}(f)\right) \tag{4.2}$$

Here, $\mathbf{y}(f)$ is a $L \times 1$ vector, and $\mathbf{s}(f)$ is an $N \times 1$ sparse vector, whose n^{th} element $S_n(f)$ is the n^{th} narrowband channel at the baseband (i.e., n^{th} subband of the original spectrum $S(f)$ of width B Hz, as shown in Fig. 4.1, shifted to the left by nB Hertz).

Each element of $\mathbf{y}(f)$ is a sum of multiple input bands of original signal. This is similar to the MWC scheme, with one important difference. In the MWC, this summation is a weighted sum, not limited to a fixed number w_r of subbands. The MWC weights are complex-valued coefficients of the Fourier Transform of the rectangular-shaped mixing sequences.

In the proposed approach, in contrast, the summation is over a fixed number w_r of subbands, and this is not a weighted sum (coefficients of summation are all unit valued). Effectively, the measurement matrix $\mathbf{A}$ is a Low Density Parity Check (LDPC) matrix with a total of w_r ones in a row and w_c ones in a column, with $w_r, w_c \ll N$, and its effect, as illustrated in Fig. 4.6 is to fold ℓ^{th} set of w_r narrowband channels to the first narrowband $\frac{B}{2}[-1, 1]$.

Unlike the MWC scheme, the SNR loss due to spectrum foldover caused by the measurement matrix is significantly reduced in the proposed approach, since $w_r \ll N$ (in contrast, all N elements of each row of the MWC measurement matrix are nonzero). Furthermore, the rectangular pulses used in MWC for the ana-

log mixing signal generate $\sin(x)/x$ profile (waveform) in the frequency domain (this is captured by diagonal matrix $\mathbf{D}$ in equation (20) of [17]). The null of this frequency-domain pulse shape coincides with the edge of the wide band W that is sensed. Effectively, the rectangular pulse shape causes a low-pass filtering effect, thereby, penalizing the higher frequency subbands in the signal reconstruction. In contrast, the coefficients of summation are all unit valued in the proposed approach, thereby entirely avoiding the noise enhancement problem. Also in the proposed scheme, it is possible to apply a variant of the standard binary LDPC message-passing decoder to recover the subband signals from the output of the measurement matrix.

Designs for Time-Varying Signal Sparsity

The minimum required number L of hardware branches is discussed in Section 4.3, where it is shown to be a function of the ratio $p \stackrel{\text{def}}{=} k/N$, which measures the sparsity of the signal $x(t)$. When p varies with the time, the design of the analog front-end hardware architecture poses several choices. One, inefficient, option is to design for the worst-case sparsity level. The advantage of this choice, in which the hardware is over-designed, is that the recovery performance improves with decreasing value of p below the maximum permissible limit. However, the drawback of this method is a potentially over-designed, and expensive RF architecture. An alternative, therefore, is to design for the average case and either perform time-segmentation [98] on each hardware branch, or frequency-segmentation [17]. In the latter method, the bandwidth of the analog LPF is changed to $\frac{u}{2}[-B, B]$, where u is a positive integer.

Model for Sparsity Estimation

For sparsity estimation, let us begin by constructing a correlation matrix [17]

$$\mathbf{R} = \int_{-B/2}^{B/2} \mathbf{y}(f)\mathbf{y}^H(f)df$$

where, $\mathbf{y}(f)$ is $L \times 1$ vector defined in (4.2) (the superscript $(\cdot)^H$ denotes the Hermitian transpose).

Assuming the signal vector $\mathbf{s}(f)$ and the noise vector $\mathbf{n}(f)$ are uncorrelated, the correlation matrix can be written, given (4.2), as

$$\mathbf{R} = \mathbf{A}\left[\mathbf{R}_s + \mathbf{R}_n\right]\mathbf{A}^T, \tag{4.3}$$

where $\mathbf{R}_s \overset{\text{def}}{=} \int_{-B/2}^{B/2} \mathbf{s}(f)\mathbf{s}^H(f)df$ is the correlation matrix of signals in different subbands, and $\mathbf{R}_n \overset{\text{def}}{=} \int_{-B/2}^{B/2} \mathbf{n}(f)\mathbf{n}^H(f)df$ is the correlation matrix of the noise affecting N subbands.

Leveraging Parseval's identity, the correlation matrices can be estimated by time-domain (instead of frequency domain) averaging, provided the averaging duration P in samples is greater than the dimension N of the vector $\mathbf{y}(n)$.

$$\mathbf{R} = \sum_{n=1}^{P} \mathbf{y}(n)\mathbf{y}^H(n)$$
$$\mathbf{R}_s = \sum_{n=1}^{P} \mathbf{s}(n)\mathbf{s}^H(n)$$

When the wideband noise is spectrally white in each of N subbands, the noise correlation matrix $\mathbf{R}_n$ becomes $\sigma_n^2\,\mathbf{I}_{N\times N}$. Similarly, we take the signals in different subbands to be generated from independent and identically distributed (iid) processes, with a variance of σ_s^2, i.e., $\mathbf{R}_s = \sigma_s^2\ \text{diag}\left\{\mathbf{v}_1^S\right\}$, where $\mathbf{v}_1^S$ is a sparse $N \times 1$ binary vector, with ones at the subband locations where the signal $S_n(f)$ is present, and zeros everywhere else.

Therefore, we can write $\mathbf{R}$ as follows:

$$\begin{aligned}
\mathbf{R} &= \mathbf{A}\left[\mathbf{R}_s + \sigma_n^2\mathbf{I}\right]\mathbf{A}^T \\
&= \mathbf{A}\mathbf{R}_s\mathbf{A}^T + \sigma_n^2\mathbf{R}_A \\
&= \sigma_s^2\mathbf{A}\,\text{diag}\left\{\mathbf{v}_1^S\right\}\mathbf{A}^T + \mathbf{R}_A\sigma_n^2 \\
&= \sigma_s^2\mathbf{R}_{A_S} + \mathbf{R}_A\sigma_n^2
\end{aligned}$$

Here, matrix $\mathbf{R}_{A_S} \stackrel{\text{def}}{=} \mathbf{A}\,\text{diag}\left\{\mathbf{v}_1^S\right\}\mathbf{A}^T$, and $\mathbf{R}_A \stackrel{\text{def}}{=} \mathbf{A}\mathbf{A}^T = \{\rho_{i,j}\}$.

The element at $(i,j)^{th}$ position of matrix $\mathbf{R}_A$ is the inner product between two arbitrary rows $\mathbf{A}_i$ and $\mathbf{A}_j$, i.e.,

$$\rho_{i,j} \stackrel{\text{def}}{=} \mathbf{A}_i\mathbf{A}_j^T = \mathbf{a}_{i,j}\mathbf{v}^1 \tag{4.4}$$

Here, $\mathbf{a}_{i,j} \stackrel{\text{def}}{=} \mathbf{A}_i \circ \mathbf{A}_j$ is a $1 \times N$ vector obtained by element-by-element (Hadamard) product of vectors $\mathbf{A}_i$ and $\mathbf{A}_j$ and $\mathbf{v}^1 \stackrel{\text{def}}{=} [1,1,\ldots,1]^T$ is a $N \times 1$ vector of all ones.

Given that the vector $\mathbf{A}_i$ is binary-valued,

$$\mathbf{a}_{i,i} = \mathbf{A}_i \circ \mathbf{A}_i = \mathbf{A}_i, \tag{4.5}$$

and we have that $\rho_{i,i} = \mathbf{A}_i\mathbf{v}^1 = w_r$. It can be shown, when $i \neq q$, $\rho_{i,j} \to 0$ as $N \to \infty$ for a given value of w_r. In this case, $\mathbf{R}_A \to w_r\,\mathbf{I}_{L\times L}$.

The following lemma states several relationships that are used subsequently.

Lemma 4.2.1 *The vector $\bar{\lambda}_R$ of eigenvalues of $\mathbf{R}_{A_S}$ approaches the vector $\mathbf{w}$ formed by the diagonal entries of $\mathbf{R}_{A_S}$ ($\mathbf{w} \stackrel{\text{def}}{=} \text{diag}\left\{\mathbf{R}_{A_S}\right\}$) as $\mathbf{R}_A$ approaches a diagonal matrix. Specifically,*

$$\bar{\lambda}_R \to \mathbf{w} \quad \textit{as } \mathbf{R}_A \to w_r\,\mathbf{I}_{L\times L} \tag{4.6}$$

Furthermore, the vector $\mathbf{w}$ can be modeled as the sparsity vector $\mathbf{v}_1$ linearly transformed by the measurement matrix $\mathbf{A}$, i.e.,

$$\mathbf{w} = \mathbf{A}\mathbf{v}_1^S \tag{4.7}$$

Consider $\{i,j\}^{th}$ element $\rho'_{i,j}$ of matrix $\mathbf{R}_{A_S\,i,j}$. This can be written as

$$\rho'_{i,j} = \mathbf{A}_i\,\text{diag}\left\{\mathbf{v}_1^S\right\}\mathbf{A}_j^T = \left(\mathbf{A}_i \circ \mathbf{A}_j\right)\,\mathbf{v}_1^S = \mathbf{a}_{i,j}\,\mathbf{v}_1^S. \tag{4.8}$$

Comparing (4.4) with (4.8), it can be seen that $0 \leq \rho'_{i,j} \leq \rho_{i,j}$. This shows that as $\mathbf{R}_A \to w_r\,\mathbf{I}_{L\times L}$, $\mathbf{R}_{A_S}$ approaches the diagonal matrix $\mathbf{w}$ (this is because $\rho_{i,j} \to 0$ if $i \neq j$, and therefore, $\rho'_{i,j} \to 0$). Given that, the eigenvalues of a diagonal matrix are the elements along its diagonal, the proof of (4.6) is completed.

Next, consider the vector $\mathbf{w} = \text{diag}\left\{\mathbf{R}_{A_S}\right\}$. The i^{th} element of this vector is given as

$$\mathbf{w}_i = \rho'_{i,i} = \mathbf{a}_{i,i}\,\mathbf{v}_1^S = \mathbf{A}_i\,\mathbf{v}_1^S. \tag{4.9}$$

The equality in the above is derived from (4.5) and (4.8). Therefore, we have

$$\mathbf{w} = \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_L \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_L \end{bmatrix} \mathbf{v}_1^S = \mathbf{A}\mathbf{v}_1^S. \tag{4.10}$$

This completes the proof of (4.7).

4.2.3 Sensing Matrix Construction

In the proposed system model and architecture, the hardware realization is based on sparse binary LDPC matrix, which has well defined construction methods in the field of communication. The construction of LDPC matrix is very important to the sensing architecture, as it affects the robustness of signal reconstruction in CS framework. In our proposal, we have used a sparse binary matrix construction which is free from cycle of 4 and cycle of 6 as described in [99].

4.3 Theoretical Bounds on the Size of Measurement Matrix

This section focuses on the minimum number L of hardware paths, i.e., the size of the measurement vector $\mathbf{y}(f)$, given the blind nature of spectrum detection problem (where no a-prior information about the spectral occupancy is assumed). We derive theoretical lower bound on the ratio L/N for ensuring a perfect detection of spectral occupancy.

Besides defining the requirement, this result can also be interpreted as the necessary requirement for a successful signal recovery/reconstruction from sub-Nyquist samples. Detection of sparsity pattern ${v_1}^S$ is the primary requirement

for signal reconstruction [17], since given the knowledge of ${v_1}^S$, the measurement $\mathbf{y}(f)$ can be modeled as $\mathbf{y}(f) = \mathbf{A}_S\mathbf{x}(f)$, where $\mathbf{A}_S$ is a matrix constructed by taking those columns of $\mathbf{A}$, which has ${v_1}^S$ a unit-valued coefficient at the corresponding positions. The reconstruction of signal $\mathbf{x}(f)$ is possible by taking pseudoinverse of $\mathbf{A}_S$, where the only requirement is that all columns of $\mathbf{A}_S$ are linearly independent. The pictorial representation of sparsity detection and signal reconstruction is shown in Fig. 4.3.

4.3.1 Derivation of Lower Bounds: Coding Theory Approach

In CS, the problem of identifying the sparsity pattern of a sparse binary vector $\mathbf{x}$ is analogous to the problem of decoding the information bits from the noisy received version of the transmitted code word, $\mathbf{c}$. The received vector $\mathbf{r}$ can be written as:

$$\mathbf{r} = \mathbf{c} + \mathbf{e}$$

In the above equation, $\mathbf{e}$ is a vector of zeros and ones, where the location of ones indicate the channel induced bit error. Code word vector $\mathbf{c}$ is generated from a vector $\mathbf{m}$ of M information bits using a block code with a generator matrix $\mathbf{G}$ of size $N \times M$ as:

$$\mathbf{c} = \mathbf{G} \times \mathbf{m}$$

Sensing matrix $\mathbf{A}$ of dimension $L \times N$ is constructed using the basis vector of the null space of $\mathbf{G}$. Number of rows L in matrix $\mathbf{A}$ equals the redundancy introduced,i.e., $L = N - M$. Since $\mathbf{A}$ spans the null space of $\mathbf{G}$, $\mathbf{A} \times \mathbf{G} = 0$.
Consider the matrix product $\mathbf{v} = \mathbf{A} \times \mathbf{r}$.

$$\mathbf{v} = \mathbf{A} \times \mathbf{r} = \mathbf{A} \times (\mathbf{c} + \mathbf{e}) = (\mathbf{A} \times \mathbf{G} \times \mathbf{m}) + \mathbf{A} \times \mathbf{e} = \mathbf{A} \times \mathbf{e}$$

Information theory provides a feasibility bound on the communication problem. It has been leveraged upon to derive the bound on the size of the CS sensing matrix $\mathbf{A}$. A binary symmetric channel (BSC) with the crossover probability p is shown in Fig. 4.7. Since the occurrence of bit errors are represented by ones in

vector **e**, the variable p denotes the probability of bit errors in communications over BSC settings.

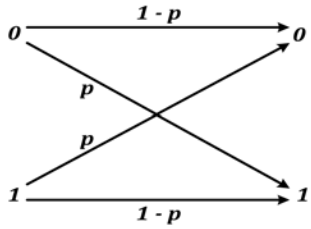


Figure 4.7: State diagram of a binary symmetric channel.

Channel capacity of a BSC is given by $C = 1 - H_b(p)$, where $H_b(p)$ is binary entropy function of p. According to the information theory, code rate is $R = \frac{M}{N} \leq C$. Therefore in CS framework, the size of sensing matrix should be such that the ratio $\frac{L}{N} = \frac{N-M}{N} = 1 - R$ is greater than or equal to $1 - C = H_b(p)$. Therefore, $H_b(p)$ is the lower bound of $\frac{L}{N}$. In case of sparsity detection using message passing algorithm, where the elements of **v** is going to take values from 0 to w_r (all with equal probability), it conveys $\log(w_r + 1)$ bits of additional information. For the type of sensing matrix used in this calculation, rate is given by $\frac{3}{w_r}$ [99]. Therefore, the binary entropy function gets divided by $\log(w_r + 1)$.

$$\frac{L}{N} > \frac{H_b(p)}{\log(w_r + 1)} \tag{4.11}$$

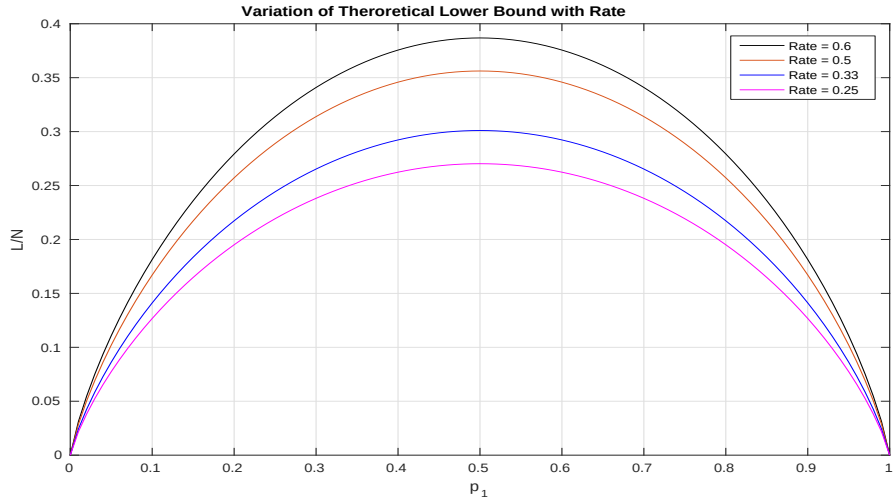


Figure 4.8: Theoretical lower bound for different rates.

This provides the theoretical lower bound. Fig. 4.8 shows the lower bound of $\frac{L}{N}$ as a function of p for different rates.

4.3.2 Derivation of Lower Bounds : An Information Theoretic Approach

We begin by positing the following lemma:

Lemma 4.3.1 *A length N sequence of discrete-valued iid symbols X can be represented by a length L sequence of iid discrete symbols Y such that $L = N \times \frac{H(X)}{H(Y)}$.*

From AEP (Asymptotic Equipartition Property) [100], the cardinality of the typical set of sequences of symbols X, as $N \to \infty$, is given as $2^{NH(X)}$, where $H(X)$ is the entropy of the discrete random variable X. Cardinality of the typical set of sequences of symbols Y of length L is given as $2^{LH(Y)}$. For lossless and efficient representation, we require a value of L such that the sizes of these two sets are equal. This value is $L = N \times \frac{H(X)}{H(Y)}$ (alternatively, the length N sequence of symbols X can be compressed to $NH(X)$ bits, whereas the sequence of symbols Y can be compressed to $LH(X)$ bits. Therefore, $L = N \times \frac{H(X)}{H(Y)}$ when these bit sizes are equal). The following theorem generalizes Lemma 4.3.1.

Theorem 4.3.1 *Suppose a sparse vector $\mathbf{x}$ of size $N \times 1$ with discrete-valued elements X is transformed by a CS measurement matrix $\mathbf{A}$ (of size $L \times N$) to a vector $\mathbf{y} = \mathbf{A}\mathbf{x}$ of size $L \times 1$. For a successful recovery of $\mathbf{x}$ from $\mathbf{y}$, the ratio of $\frac{L}{N}$ of the rows to columns of matrix $\mathbf{A}$ is lower bounded by $\frac{H(X)}{H(Y)}$.*

We first show that

$$H(\mathbf{y}) = H(\mathbf{x}). \tag{4.12}$$

Consider $H(\mathbf{x}, \mathbf{y})$, which is expressible as follows [100, (Problem 5 of Chapter 2)]:

$$\begin{aligned} H(\mathbf{x}, \mathbf{y}) &= H(\mathbf{x}) + H(\mathbf{y}|\mathbf{x}) \stackrel{a}{=} H(\mathbf{x}) \\ &= H(\mathbf{y}) + H(\mathbf{x}|\mathbf{y}) \stackrel{b}{=} H(\mathbf{y}) \end{aligned}$$

In the above, equality (a) follows, given that $\mathbf{y}$ is determined by $\mathbf{x}$, whereas equality (b) is due to one-to-one mapping between $\mathbf{y}$ and $\mathbf{x}$ is required for a guaranteed recovery of $\mathbf{x}$ from $\mathbf{y}$.

Elements X of $N \times 1$ vector $\mathbf{x}$ are assumed as iid. Therefore, $H(\mathbf{x}) = N \times H(X)$ (from the additivity property of Entropy [100]). Elements Y of $L \times 1$ vector $\mathbf{y}$ may not be independent. Therefore, $H(\mathbf{y}) \leq L \times H(Y)$. From (4.12), it follows that $N \times H(X) \leq L \times H(Y)$, i.e., the ratio L/N is lower bounded by $H(X)/H(Y)$. We now consider several example applications of theorems derived in Section 4.3.2 to determine theoretical lower bounds for different use cases.

4.3.3 Type of Applications and their Lower Bounds

Focus of our work is on an architecture in which the measurement matrix $\mathbf{A}$ takes the form of an LDPC matrix [101]. We take this matrix to be regular, i.e., let each row with N elements have a fixed w_r elements that are ones, with the rest being zeros.

Binary Input and Binary Output

As a specific case with this measurement matrix, let the input sparse vector be binary-valued with a probability of $X = 1$ denoted as p. Let the measurement matrix $\mathbf{A}$ be such that Y is fully randomized, and probability that $Y = 1$ is 0.5. With this $H(X) = H_b(p) = -p\log_2(p) - (1-p)\log_2(1-p)$ and $H(Y) = 1$. The lower bound on the size of $\mathbf{A}$, i.e., L/N, becomes $H_b(p)$.

This lower bound can also be derived from a forward error correction perspective. Denoting the transmitted $N \times 1$ binary codeword vector as $\mathbf{c}$, the received bit vector on a Binary Symmetric Channel (BSC) can be written as $\mathbf{r} = \mathbf{c} + \mathbf{x}$ (all the additions for the binary case are modulo-two). In the above, $\mathbf{x}$ is a vector of zeros and ones, with the locations of ones designating the channel-induced bit errors. If $\mathbf{A}$ is the parity check matrix for this code,

$$\mathbf{y} = \mathbf{A}\mathbf{r} = \mathbf{A}\left(\mathbf{c} + \mathbf{x}\right) = \mathbf{A}\mathbf{x}$$

Probability p of occurrence of 1 in $\mathbf{x}$ is equivalent in this setting to the probability of bit errors over BSC channel. Since the channel capacity for a BSC with crossover probability p is known to be $C = 1 - H_b(p)$, the size of the measurement matrix $\mathbf{A}$ is constrained such that the ratio $\frac{L}{N} = 1 - R$ is greater than or equal to $1 - C = H_b(p)$ (here, $L = N - M$ is the extra redundancy introduced in a message vector of length M bits to derive the codeword c of N bits, and $R = M/N$ is the code rate).

Binary Input and Nonbinary Output

In the proposed CS architecture, measurement vector $\mathbf{y}$ (= $\mathbf{Ax}$) is non-binary (as modulo-2 operation is not performed).

Since the matrix $\mathbf{A}$ has a total of w_r ones and $N - w_r$ zeros randomly located in any row, an element $Y \in \mathcal{Y} = \{0, 1, \ldots, w_r\}$ of $\mathbf{y}$, has Binomial Distribution, $P_Y(k) = \binom{n}{k} p^k (1-p)^{w_r - k}$. Its entropy $H(Y)$ can be approximated as $\frac{1}{2}\log_2(2\pi e n p(1-p))$. This allows a bound on the maximum value of $\frac{L}{N}$, which occurs when $p = 0.5$. For this value of p, $H(X) = H_b(p) = 1$ bit and $H(Y) \approx \frac{1}{2}\log_2(\pi e n/2) \approx 1 + \log_2(n)/2$ bit. Maximum value of the ratio L/N is lower bounded in this case by $1/(1 + 0.5\log_2(n))$.

Nonbinary Discrete Input and Nonbinary Output

A generalized case of the above is when the input is non-binary but is drawn from a discrete-valued set $X \in \mathcal{X} = \{0, 1, \ldots, M-1\}$, with the probability of $X = m$ equal to p_m. In this case, $Y \in \mathcal{Y} = \{0, 1, \ldots, (M-1) * w_r\}$ has the summed multinomial distribution.

$$p(y) = \sum_{k \in \mathcal{K}_y} \frac{w_r!}{w_0^{(k)}!\ w_1^{(k)}!\ \ldots\ w_{M-1}^{(k)}!} \times \left(p_0^{w_0^{(k)}} \cdot p_1^{w_1^{(k)}} \cdot \ldots \cdot p_{M-1}^{w_{M-1}^{(k)}} \right)$$

Here, $\mathcal{K}_y$ is a set for which $\sum_{m=0}^{M-1} w_m^{(k)} m = y$, with $w_m^{(k)}$ ($\sum_{m=0}^{M-1} w_m^{(k)} = w_r$) denoting the number of occurrences of $X = m$ in the summation performed in m^{th} row of $\mathbf{A}$.

4.4 Algorithms for Blind Support Detection

The performance of CS systems depends on support detection algorithms. The algorithm needs to perform blind active support detection. In the proposed framework, the active support detection task amounts to sparsity detection (Refer Section 4.2).

The two algorithms for sparsity detection presented in this section are based on hard decision decoding used in channel coding. The first algorithm is motivated by Gallager algorithm for LDPC decoding and second algorithm is based on Sudocode. Additionally, the two novel BP algorithms developed to solve the sparsity detection problem are further described in Chapter 5.

Notations

Following are the standard conventions used in description of message passing algorithms over bipartite graphs [102], $q_{n\to m}$ denote the message sent by n^{th} variable node b_n to the check node c_m, and $u_{m\to n}$ denote the message sent by the check node c_m to the variable node b_n. V_n denotes a set of indices of check nodes connected to n^{th} variable node, and C_m denotes a set of indices of variable nodes connected to m^{th} check node. $C_{m,n} = C_{m\setminus n}$ is the set C_m without n^{th} variable node, and $V_{n,m} = V_{n\setminus m}$, i.e., the set V_n without m^{th} check node.

4.4.1 Algorithm A: Based on Gallager LDPC Decoding Technique

The algorithm is motivated by Gallager LDPC decoding technique and is summarized here with its listing shown on Page 61.

- *Initialization:*
  - Lines 1 to 3: In the initialization step, only zero-valued check nodes pass a message to the connected variable nodes, all of which gets initialized with zero. Other variable nodes are set to undetermined state.
  - Lines 4 to 6: If *any* of the check nodes connected to a variable node has sent a message 0, the variable node sends 0 to *all* other check nodes

connected to it.

- Lines 8 to 15 (Message $u_{m\to n}$ from m^{th} check node to n^{th} variable node):
  - If a variable node has sent either 0 or 1 valued message to a check node, the check node sends this message back to the variable node.
  - If $u'_{m\to n} = r_m - \sum_{n'\in C_{m,n}} q_{n'\to m}$ equals either 0 or 1, send $u'_{m\to n}$ as the message $u_{m\to n}$, else set $u_{m\to n}$ to undetermined $*$ state.
- Lines 16 to 23 (Message $q_{n\to m}$ from n^{th} variable node to m^{th} check node):
  - Calculate subset $V'_{n,m}$ of $V_{n,m}$ for which messages from check nodes connected to n^{th} variable nodes are either 0 or 1.
  - If cardinality of $V'_{n,m}$ is zero, $q_{n\to m} = *$, else $q_{n\to m}$ equals 0 if majority of the messages $u_{m'\to n}$, with $m' \in V'_{n,m}$ are 0; otherwise it is 1.

Algorithm: A

for *all check node indices* $m \in \{1, \ldots, M\}$ **do**
    if $c_m = 0$ **then** $\forall n \in C_m$, $u_{m \to n} \leftarrow 0$
    else $u_{m \to n} = *$
for *all variable node indices* $n \in \{1, \ldots, L\}$ **do**
    if $\exists\, m' \in V_n$, *such that* $u_{m' \to n} = 0$ **then** $q_{n \to m} \leftarrow 0$
    else $q_{n \to m} \leftarrow *$
do
    for *all check node indices* $m \in \{1, \ldots, M\}$, *and variable node indices* $n \in C_{m,n}$ **do**
        if $q_{n \to m} \neq *$ **then** $u_{m \to n} \leftarrow q_{n \to m}$
        else
            if $q_{n' \to m} = *$ *for any* $n' \in C_{m,n}$ **then** $u_{m \to n} \leftarrow *$
            else
                $u'_{m \to n} = r_m - \sum_{n' \in C_{m,n}} q_{n' \to m}$
                if $u'_{m \to n} = 0$ *or* $u'_{m \to n} = 1$ **then** $u_{m \to n} \leftarrow u'_{m \to n}$
                else $u_{m \to n} \leftarrow *$
    for *all variable node indices* $n \in \{1, \ldots, L\}$ *and check node indices* $m \in V_{n,m}$ **do**
        Calculate set $V'_{n,m}$ where $\forall m' \in V'_{n,m}, u_{m' \to n} \neq *$
        $s_m \leftarrow$ cardinality of set $V'_{n,m}$
        $t \leftarrow \sum_{m' \in V'_{n,m}} u_{m' \to n}$
        if $s_m = 0$ **then** $q_{n \to m} \leftarrow *$
        else
            if $s <= s_m/2$ **then** $q_{n \to m} \leftarrow 0$
            else $q_{n \to m} \leftarrow 1$
while *at least one variable node is undetermined* ***or*** *number of iterations is less than a threshold*

4.4.2 Algorithm B : Based on Sudocodes

Sudocodes are new compressive sampling schemes for measurement and reconstruction of sparse signals using algorithms on graphs [103, 104]. The second algorithm, Algorithm B, closely follows sudocodes decoding algorithm. Its listing, provided on Page 62, is self-explanatory.

Algorithm: B

Similar to algorithm A, in the initialization step, all the variable nodes are set to undetermined $*$ state

Only the zero-valued check nodes are selected and the variable nodes connected to the selected check nodes are set to zero

do

  for ($i = 1;\ i \leq w_r;\ i \leftarrow i + 1$ **)** {

    for *each check node with a value of i* **do**

      if *sum of the determined variable nodes equals i* **then**

        all the undetermined variable nodes connected to this check node are set to zero.

      if *number of undetermined variable nodes equals i* **then**

        all of i undetermined variable nodes are set to unity.

while *at least one variable node is undetermined* ***or*** *number of iterations exceeds a threshold*

4.4.3 Performance comparison

The performance of Algorithm A and Algorithm B for different size of measurement matrix and different sparsity levels (p_1 represents probability of ones in a given block of size N) was verified through Monte Carlo simulation. The performance comparison as shown in Fig. 4.9, suggests that Algorithm A outperforms Algorithm B. The details of Belief Propagation based two improved sparsity detection algorithms developed under this dissertation work are presented in next

chapter.

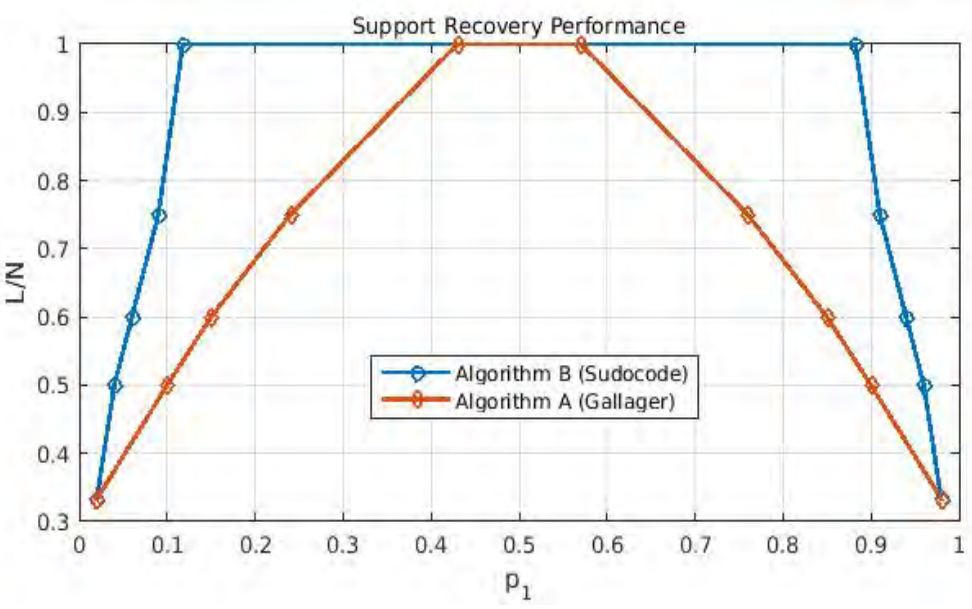


Figure 4.9: Algorithm performance comparison (size of A vs sparsity).

4.5 Signal Reconstruction

The signal reconstruction method in the proposed architecture is similar to that described in MWC [38]. However, in our work, we propose to use different algorithms other than OMP for active band detection. Two methods motivated by Gallager LDPC decoding and Sudocoding with binary input (hard decision) is used for active support detection. Two novel message passing algorithms developed for active support detection is described in next chapter. Subsequently, the performance of all algorithms are compared with theoretical bounds to prove the robustness claim.

The Model of Discrete Time Measurements [4.2.2] of proposed scheme describes $Y(f) = AX(f)$ where A is the sensing matrix. The multi-band input signal to the model is assumed to be sparse and binary (All bands are assumed to have equal bandwidth with same power per unit bandwidth). In the proposed scheme, the discretized output of low pass filter gives $y[n]$, which upon passing through the energy detector gives discrete output ranging between 0 to w_r. This binary output is then given to the modified Sudocode and the Gallager algorithms to

identify the active supports. Columns of the sensing matrix A, corresponding to the detected active supports, forms the modified sensing matrix A_s. The product of pseudo-inverse of the matrix A_s with the measured signal vector (**y[n]**), gives the recovered signal ($\hat{x}[n]$). The architecture recovery performance was verified with four randomly located input bands with CW and QPSK modulated signals.

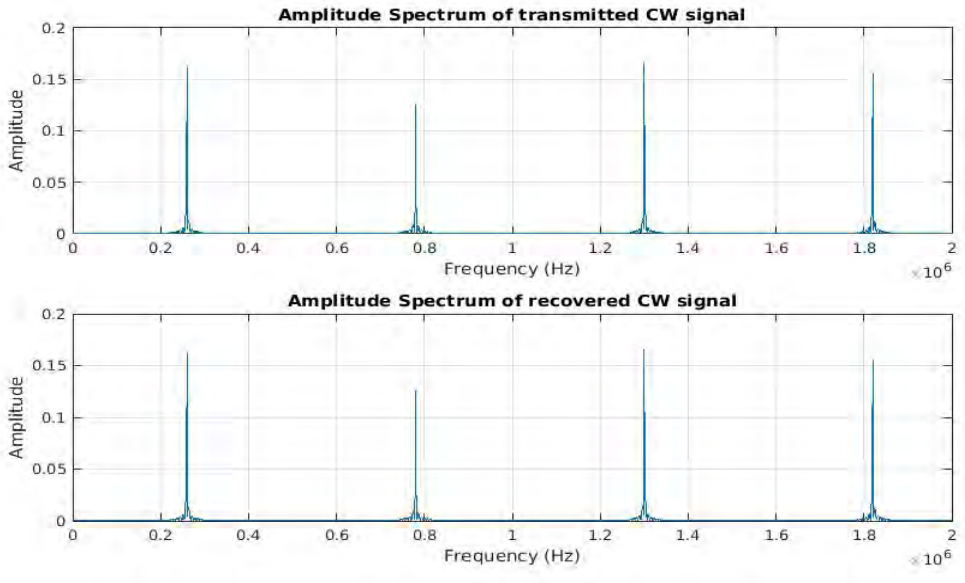


Figure 4.10: Recovery performance of proposed architecture in frequency domain with CW input signal.

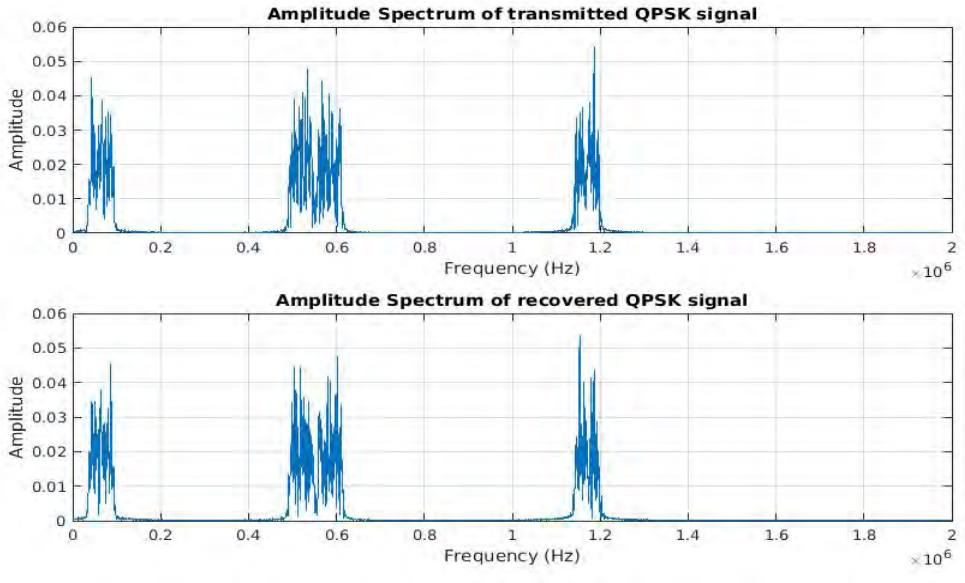


Figure 4.11: Recovery performance of proposed architecture in frequency domain with QPSK input signal.

Fig. 4.10 represents the recovered signal with CW input signal at the center of four randomly selected bands. Fig. 4.11 shows the recovered QPSK modulated input signal at the center of four randomly selected bands.

4.6 Noise Performance of Proposed Architecture

The noise performance of CS receiver is very important metric for communication applications. Most of the sensing architectures like MWC has inherent problem of noise fold over [80]. In this section, the theoretical noise performance analysis of the proposed architecture and simulation based verification is presented. The proposed CS receiver performance is verified by doing Bit Error Rate (BER) simulation analysis with QPSK modulated signal. Also, the BER performance of the proposed CS receiver has been compared with MWC based CS receiver.

Theoretical Noise Performance Analysis

Consider that a multiband signal with additive channel noise enters into the proposed sensing device shown in Fig. 4.2. For noise performance analysis let us consider the model of the sub-Nyquist rate samples at the output of the low-pass filter $h(t)$ (e.g., which is typically square-root raised cosine filter for digital communication transceivers)

$$\mathbf{y}(n) = \mathbf{A}\mathbf{x}(n) = \mathbf{A}(\mathbf{s}(n) + \mathbf{n}(n)) \tag{4.13}$$

Here, A is the sensing matrix and $x(n)$ is the multiband input signal (with N subbands) which is sparse (k sparse) in frequency domain. When N - k locations of zero-valued elements of the sparse vector **x**(n) are known, the corresponding columns of **A**, that multiply with zero-valued elements of subbands signal vector **s**(n), can be removed. With this, the original CS measurement matrix **A** is reduced to a matrix $\mathbf{A}_S$ of size L x k, where $L > k$ and represents the number of parallel branches or rows of sensing matrix. Here, S denotes the sparsity pattern (i.e., the locations of subbands x(n) which are occupied by valid signals), $\tilde{S}$ denotes the

complementary set containing the indices of subbands with only noise, $\mathbf{A}_S$ is a matrix of size L x k.

$$\mathbf{y}(n) = \mathbf{A}_S(\mathbf{s}_S(n) + \mathbf{n}_S(n)) + \mathbf{A}_{\tilde{S}}\mathbf{n}_{\tilde{S}}(n) \tag{4.14}$$

Here, $\mathbf{s}_S$(n) denotes a vector of size k x 1 with signals of active subbands, and $\mathbf{n}_S$(n) is a vector of the same size denoting the noise affecting these signals. Matrix $\mathbf{A}_{\tilde{S}}$, of size L x $(N-k)$, comprises of those columns of $\mathbf{A}$ that multiply with the signals of non-active subbands, and vector $\mathbf{n}_{\tilde{S}}$(n) of size $(N-k)$ x 1 denotes the noise present in the nonactive subbands. Assuming that k columns of $\mathbf{A}_S$ are linearly independent, the standard Least Squares solution (i.e., the pseudoinverse of $\mathbf{A}_S$)

$$\mathbf{A}_s{}^{\dagger} = (\mathbf{A}_s{}^{T}\mathbf{A}_s)^{-1}\mathbf{A}_s{}^{T} \tag{4.15}$$

is used to reconstruct the original sparse excitation pattern $\mathbf{x}(n)$ from the measurement vector $\mathbf{y}(n)$.

$$\hat{\mathbf{s}}_S(n) = \mathbf{A}_S{}^{\dagger}y(n) \tag{4.16}$$

$\hat{\mathbf{s}}_S(n) = \mathbf{A}_S{}^{\dagger}(\mathbf{A}_S(\mathbf{s}_S(n) + \mathbf{n}_S(n)) + \mathbf{A}_{\tilde{S}}\mathbf{n}_{\tilde{S}}(n))$
$\hat{\mathbf{s}}_S(n) = \mathbf{s}_S(n) + \mathbf{n}_S(n) + \mathbf{A}_S{}^{\dagger}\mathbf{A}_{\tilde{S}}\mathbf{n}_{\tilde{S}}(n)$
Compared to the ideal signal recovery in which $\hat{\mathbf{s}}_S(n)$ is obtained as

$$\hat{\mathbf{s}}_S(n) = \mathbf{s}_S(n) + \mathbf{n}_S(n), \tag{4.17}$$

the proposed scheme results in enhancement of noise by an additional term

$$\mathbf{n}'(n) = \mathbf{A}_S{}^{\dagger}\mathbf{A}_{\tilde{S}}\mathbf{n}_{\tilde{S}}(n) \tag{4.18}$$

The variance of the noise that affects the vector of the recovered signals of active subbands is given as follows:

$$E[(\mathbf{n}_S(n) + \mathbf{A}_S{}^{\dagger}\mathbf{A}_{\tilde{S}}\mathbf{n}_{\tilde{S}}(n))(\mathbf{n}_S(n) + \mathbf{A}_S{}^{\dagger}\mathbf{A}_{\tilde{S}}\mathbf{n}_{\tilde{S}}(n))^{H}] = \sigma_n{}^2 I_{k\,x\,k} + \sigma_n{}^2 Q_A \tag{4.19}$$

Here,

$$Q_{\mathrm{A}} \stackrel{\Delta}{=} \mathbf{A_S}^{\dagger}\mathbf{A}_{\tilde{S}}\mathbf{A}_{\tilde{s}}{}^{\mathrm{T}}(\mathbf{A_S}^{\dagger})^{\mathrm{T}}. \tag{4.20}$$

Also, the variance on k th subband increases from $\sigma_{\mathrm{n}}{}^2$ to $\sigma_{\mathrm{n}}{}^2(1+\mathrm{Q_A(k,k)})$, where $Q_{\mathrm{A}}(k,k)$ denotes k^{th} element on the diagonal of Q_{A}.

Simulation Based Noise Performance Analysis

The setup for noise performance evaluation of proposed CS based receiver is shown in Fig. 4.12. In this simulation, a multiband QPSK modulated signal has been generated and fed to proposed CS based receiver for BER performance evaluation. Later, the same input is fed to MWC based receiver for BER performance comparison.

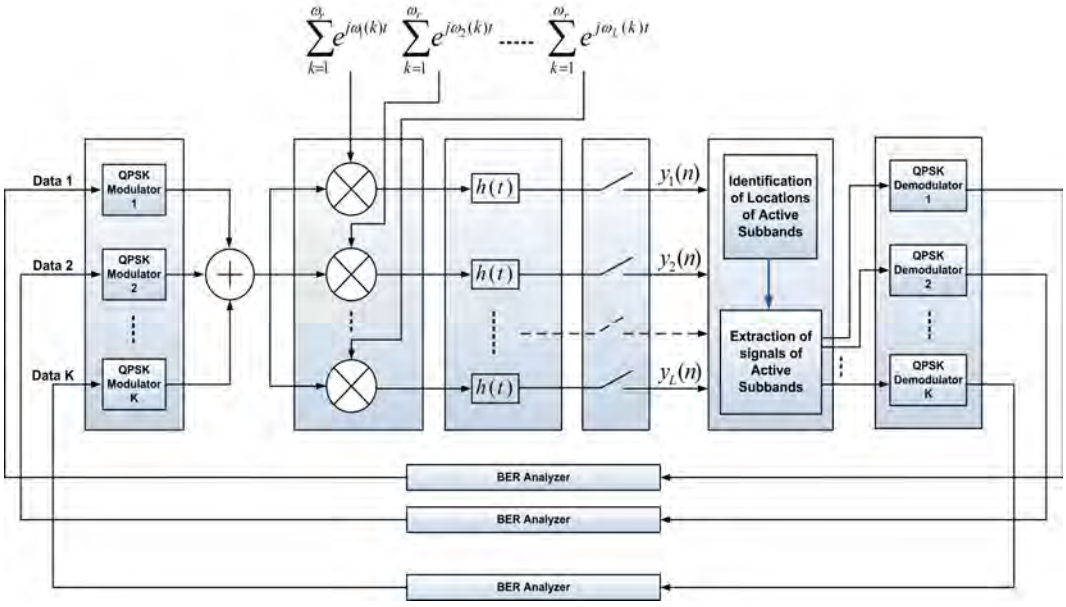


Figure 4.12: Simulation setup for noise performance evaluation of CS.

The simulation is conducted using a wideband of width W Hertz comprising of a total of $N = W/B = 36$ narrow bands of width B Hertz. A total of $\kappa = 4$ active subbands are simulated at subband locations $2, 8, 25$ and 33 (as shown in Fig. 4.13).

Each subband carries a QPSK-modulated symbol stream, with symbol rate of $R_S = B/(1+\alpha_{SRRC})$, where the roll-off factor α_{SRRC} of the square-root raised cosine filter is kept to be 0.25. The modulated stream added with AWGN noise

is fed to proposed CS receiver. The CS receiver uses the developed algorithm for detection of active sub-band locations. The band location along with ADCs output samples are used for signal reconstruction and symbol demodulation. The demodulated symbol is used for BER comparison.

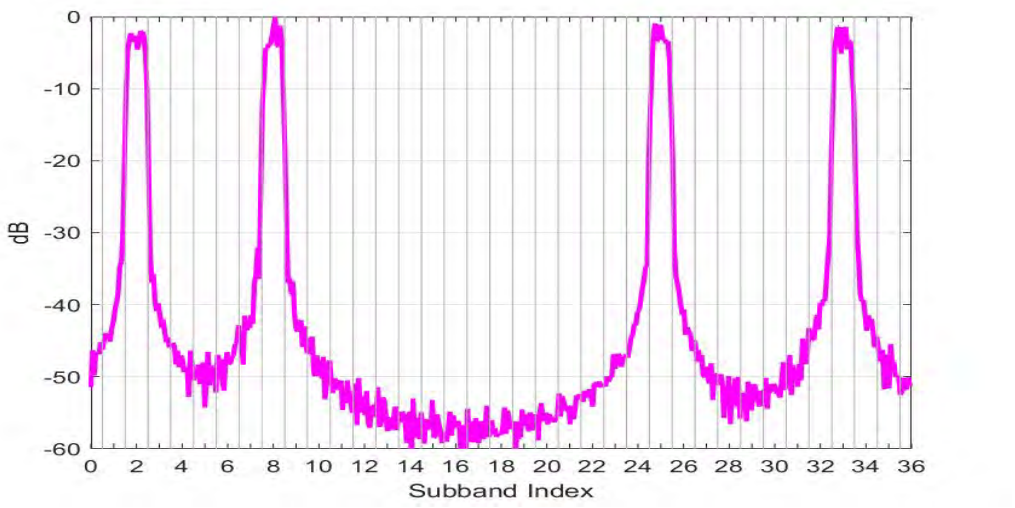


Figure 4.13: Simulated wideband signal (noiseless case) with $N = 36$ narrow bands and four active bands.

The average bit error probability after signal reconstruction is shown in Fig. 4.14 for the proposed method and for the MWC approach [17].

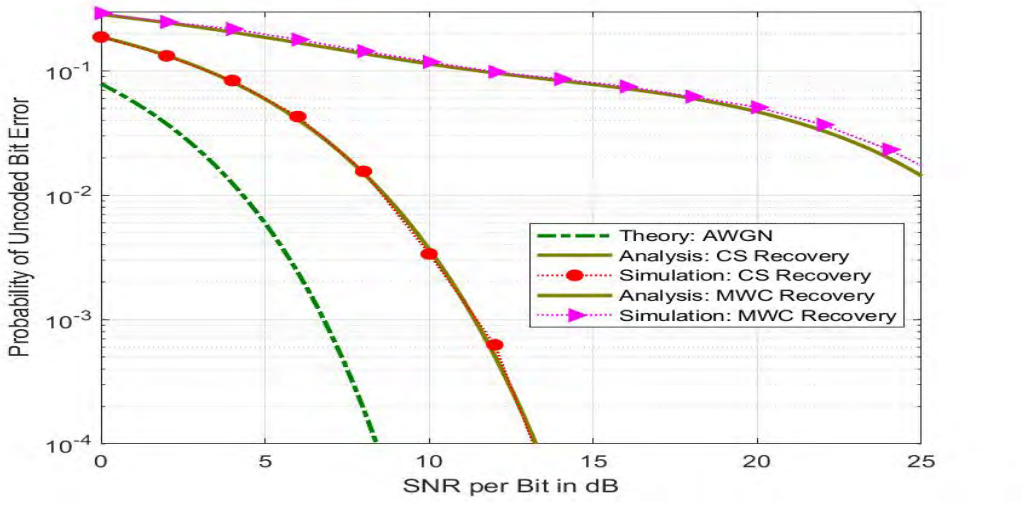


Figure 4.14: QPSK bit error probability: proposed approach versus MWC.

The simulated bit error probability compares well with the analytical result

obtained as follows:

$$P_b^{\text{err}} = \sum_{k=1}^{\kappa} Q\left(\sqrt{\frac{2E_b}{N_0 \times (1 + \mathbf{Q_A}(S(k), S(k)))}}\right) \tag{4.21}$$

Here, $\mathbf{Q_A}(S(k), S(k))$ denotes the element at location $S(k)$ along the diagonal of matrix $\mathbf{Q_A}$ defined in (4.20), and S is the sparsity vector defined in Section 4.6. *It is seen that the MWC result is significantly deteriorated compared to that obtained using the proposed method*. This demonstrates that the proposed CS architecture offers better noise performance compared to MWC.

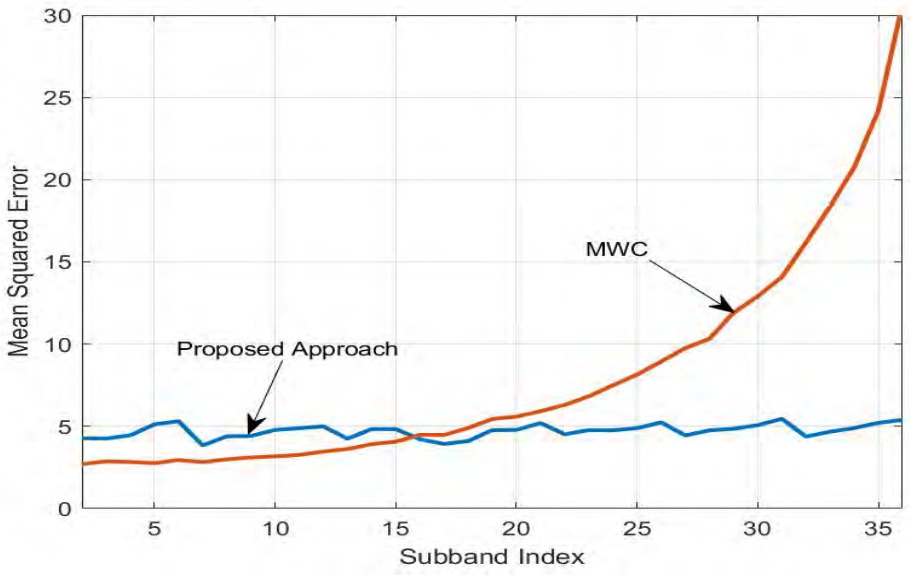


Figure 4.15: Noise amplification at different subbands; MWC versus proposed approach.

Similarly, as shown in Fig. 4.15, the Least-Squares solution exhibits (as described in Section 4.2) noise enhancement for the subbands located near W Hz. This noise enhancement is not seen in the proposed method since the frequency-domain pulse shaping is rectangular over the entire bandwidth of W Hz. Therefore, the proposed CS scheme offers better noise performance than MWC.

This analysis completes the basic description about the proposed CS architecture for wideband communication and its advantages over MWC.

Summary: The two algorithms described in this chapter are with the assumption of binary inputs. However, the algorithms do account for variance in energy de-

tector output leading to real samples for input signal combined with channel noise (i.e. non-zero sigma). Two new BP algorithms for sparsity detection, their performance analysis for measurements with non-zero sigma values are presented in the next chapter. The next chapter also includes performance comparison of these algorithms with theoretical lower bound on size of measurement matrix vs sparsity and their semi-analytical convergence analysis.

CHAPTER 5

Belief Propagation based Sparsity Detection Algorithms

In several CS architectures, the active sub-band location is first identified and then signal reconstruction is done using knowledge of the active sub-bands. When a multi-band signal is provided a input to the RF front-end of an architecture as described in Section 4.2, the problem of spectrum sensing and reconstruction reduces to a sparsity detection problem. The sparsity detection provides the location of active bands of input multiband signal, which in turn is used to select the column of sensing matrix to form a modified sensing matrix to be used for signal reconstruction. This chapter describes two novel algorithms for sparsity detection, their convergence analysis, and performance comparison against theoretical lower bounds.

Algorithmic setup for Sparsity Detection

In the proposed CS framework, the identification of active band locations of input multi-band signal amounts to sparsity detection in vector $\mathbf{x}$ given the measurement vector $\mathbf{y}$ and knowledge of sensing matrix $\mathbf{A}$ where, $\mathbf{A}$ is sparse binary LDPC matrix and $\mathbf{y}$ is the output of energy detector. It is assumed that all the active bands of input signal contains similar energy (same power flux density) making $\mathbf{y}$ a vector with elements drawn from a set of discrete alphabets.

To achieve the objective of sparsity detection, we now propose two novel message passing approaches based on probabilistic (belief) messages. Our approach differs from those in literature in the many ways. In the classical compressive

sensing framework, $\mathbf{y} = \mathbf{A}\mathbf{x}$, where, the sparse signal $\mathbf{x}$ which is to be determined is said to have elements drawn from $\mathcal{R}$, in our approach, elements of $\mathbf{x}$ are drawn from finite alphabet. In this work two cases are considered, first when $\mathbf{x}$ have only binary elements and in the second case $\mathbf{x}$ have elements drawn from the discrete non-binary set of $\{0, 1, 2\}$. The sensing matrix $\mathbf{A}$, in this work is identical to LDPC parity check matrix $\mathbf{H}$.

LDPC decoding techniques are well developed and better understood. In order to establish better correspondence for all further explanation of the sparsity detection algorithm, output of energy detector of each branch would correspond to bit nodes value and each measurement would correspond to check node value for a given sensing (LDPC like) matrix.

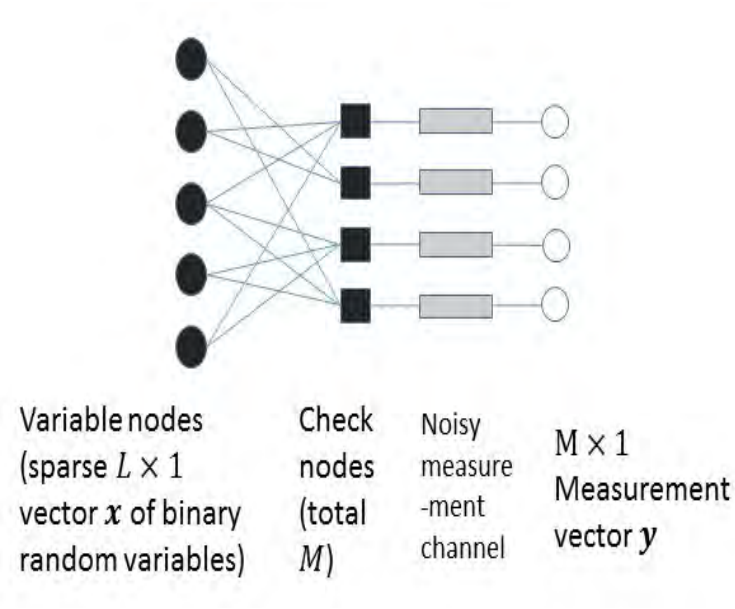


Figure 5.1: Tanner graph model of the CS spectrum sensing architecture.

In binary LDPC codes, bit parity at a given check node is ensured, i.e., modulo-2 sum of the bit nodes connected to a given check node is ensured to be zero. In the proposed CS framework, the sum (not modulo-2) of a total of w_r bit nodes connected to a check node takes an integer value from 0 (when all w_r bits are zero) to w_r (when all w_r bits are unity, i.e. when all w_r spectrum is occupied). At binary LDPC decoder, (noisy) estimates of the bit (or variable) nodes are available from the channel measurements. The decoder estimates the encoded bits that is likely to have been transmitted given the noisy channel output. In the CS setting

described here, it is the check nodes (instead of the bit nodes) whose noisy estimates are available as vector $\mathbf{y}$, and the task is to estimate the bit nodes given the measured values of the check nodes as shown in Fig. 5.1.

5.1 Proposed Sparsity Detection Algorithms

This section describes two novel algorithms derived for sparsity detection. The algorithms are based on belief message propagation.

Notations

Following the standard convention used in description of message passing algorithms over bipartite graphs [102], $q_{n\to m}$ denote the message sent by n^{th} variable node b_n to the check node c_m, and $u_{m\to n}$ denote the message sent by the check node c_m to the variable node b_n, V_n denotes a set of indices of check nodes connected to n^{th} variable node, and C_m denotes a set of indices of variable nodes connected to m^{th} check node, $C_{m,n} = C_{m\setminus n}$ is the set C_m without n^{th} variable node, and $V_{n,m} = V_{n\setminus m}$, i.e., the set V_n without m^{th} check node.

5.1.1 Message Passing Algorithm with Independent Probability Estimate (IPE)

This message passing algorithm assumes that occupancy of each band of input signal is independent of each other. Given the measurement vector y (check node vector) and sensing matrix A, the algorithm describes the probabilistic decoding approach to determine x (bit node vector representing input spectrum occupancy) as listed on Page 74. The steps of algorithm processing is described here:

- On lines 2 and 3, variable nodes messages to check nodes, $q_{n\to m}$ are initialized by the log likelihood ratio (LLR) $\log\left(p/(1-p)\right)$. Here, elements of sparse matrix $\mathbf{x}$ are assumed to be iid, where each one takes a value of 1 with probability p.

- Line 5 shows the message from the check nodes to the variable nodes, $u_{m\to n}$. This message is calculated using function $f(\cdot)$ that is described below.
- Line 6 shows the message from the variable nodes to the check nodes. This message is simply the sum of the likelihood ratios received from the neighbor check nodes.

---

1 **Algorithm: MP with IPE assumption**
2 **for** *all variable node indices* $n \in \{1, \ldots, L\}$ **do**
3 $\quad q_{n\to m} \leftarrow p/(1-p)$
4 **do**
5 $\quad$ **for** *all check node indices* $m \in \{1, \ldots, M\}$ **do**
$\quad\quad u_{m\to n} \leftarrow f\left[q_{n_1\to m}, q_{n_2\to m}, \ldots, q_{n_{w_r-1}\to m}, p_{s_m|y_m}\right], \forall n_i \in C_{m,n}$
6 $\quad$ **for** *all variable node indices* $n \in \{1, \ldots, L\}$ **do** $q_{n\to m} \leftarrow \sum_{m_i \in V_{n,m}} u_{m_i\to n}$
7 **while** *number of iterations is less than a threshold*

The heart of Algorithm (MP with IPE) is the function $f(\cdot)$ in Line 5. It is mainly this function besides the initialization scheme on Lines 2 and 3, that makes this Algorithm different from the standard BP algorithm being used in LDPC decoding. This function generates at its output the log likelihood ratio (LLR) of n^{th} variable node given the probabilities of its neighbor variable nodes connected to m^{th} check node.

Let $p^0_{m,n}$ denote the probability that n^{th} variable node is zero given the probabilities of all the variable nodes in the set $C_{m,n}$. Using Bayesian rule, this can be written as follows:

$$p^0_{m,n} = \sum_{s_m=0}^{w_r} p^0\left(u\left(m,n\right)|s_m\right) p\left(s_m|y_m\right) \tag{5.1}$$

Here, $p^0\left(u\left(m,n\right)|s_m\right)$ is the conditional probability of n^{th} variable node being zero given a specific $s_m \in \{0, 1, \ldots, w_r\}$ and $p\left(s_m|y_m\right)$ denotes the conditional probability of s_m given the measured value of y_m.

Conditional probability $p^0\left(u\left(m,n\right)|s_m\right)$ can be calculated as being proportional

to the probability of a bit-sequence set $p\left(\mathbf{b}^{w_r-1,s_m}\right)$, where

$$\mathbf{b}^{w_r-1,s_m} = \{\mathbf{b}_0^{w_r-1,s_m}, \mathbf{b}_1^{w_r-1,s_m}, \ldots, \mathbf{b}_{N_{w_r-1,s_m}-1}^{w_r-1,s_m}\} \tag{5.2}$$

is a set of bit sequences of length $(w_r - 1)$ with cardinality of $N_{w_r-1,s_m} = \binom{w_r-1}{s_m}$. Bits of each member sequence in this set add up to s_m. Similarly, bit-sequences that the variable nodes in set $C_{m,n}$ can take such that m^{th} check node has a value of s_m and n^{th} variable node has a value of 1 can be represented as a set $\mathbf{b}^{w_r-1,s_m-1} = \{\mathbf{b}_0^{w_r-1,s_m-1}, \mathbf{b}_1^{w_r-1,s_m-1}, \ldots, \mathbf{b}_{N_{w_r-1,s_m-1}-1}^{w_r-1,s_m-1}\}$. Binary sequences in this set have a length $w_r - 1$, and the bits of each member sequence add up to $s_m - 1$.

Conditional probability $p^0\left(u\left(m,n\right)|s_m\right)$ is obtained by the following ratio.

$$p^0\left(u\left(m,n\right)|s_m\right) = \frac{p_{m,n}^0(\mathbf{b}^{w_r-1,s_m})}{p_{m,n}^0(\mathbf{b}^{w_r-1,s_m}) + p_{m,n}^1(\mathbf{b}^{w_r-1,s_m-1})} \tag{5.3}$$

Here, $p_{m,n}^0(\mathbf{b}_n^{w_r-1,s_m})$ denotes the total probability of the set $\mathbf{b}^{w_r-1,s_m}$ at m^{th} check node and for n^{th} variable node and it is the sum of probabilities of individual member sequences:

$$p_{m,n}^0\left(\mathbf{b}^{w_r-1,s_m}\right) = \sum_{i=0}^{N_{w_r-1,s_m}-1} p_{m,n,0}\left(\mathbf{b}_i^{w_r-1,s_m}\right) \tag{5.4}$$

Due to iid assumption of the variable nodes in the set $C_{m,n}$, the probability $p_{m,m,0}\left(\mathbf{b}_i^{w_r-1,s_m}\right)$ in the above of i^{th} binary sequence $\mathbf{b}_i^{w_r-1,s_m}$ is the product of individual bit probabilities:

$$p_{m,n,0}(\mathbf{b}_i^{w_r-1,s_m}) = \prod_{j=1}^{w_r-1} p'_{m,n'}(\mathbf{b}_{i,j}^{w_r-1,s_m}) \tag{5.5}$$

Here, $\mathbf{b}_{i,j}^{w_r-1,s_m}$ denotes j^{th} bit of i^{th} sequence of set $\mathbf{b}^{w_r-1,s_m}$ in (5.2). Variable n' in (5.5) denotes j^{th} element of set $C_{m,n}$, and $p'_{m,n'}(\mathbf{b}_{i,j}^{w_r-1,s_m})$ denotes the probability of this variable node.

Similarly,

$$p_{m,n}^1\left(\mathbf{b}^{w_r-1,s_m-1}\right) = \sum_{i=0}^{N_{w_r-1,s_m-1}-1} p_{m,n,1}\left(\mathbf{b}_i^{w_r-1,s_m-1}\right) \tag{5.6}$$

First term of (5.1), expanded upon in (5.3), can be computed using (5.4) and (5.6). Now, we derive an expression for the second term of (5.1), i.e., $p\left(s_m|y_m\right)$. Towards this, we begin by letting $s_m \in 0, 1, \ldots, w_r$ denote the integer value that m^{th} check node can take. Measured value y_m of m^{th} check node implies a probability distribution on s_m, which is given by bayesian formula

$$p(s_m|y_m) = \frac{p(y_m|s_m)p(s_m)}{\sum_{s_m=0}^{w_r} p(y_m|s_m)p(s_m)} \tag{5.7}$$

Given the iid assumption regarding elements of $\mathbf{x}$, the prior probability $p(s_m)$ is given by binomial PMF:

$$p(s_m) = \binom{w_r}{s_m} p^{s_m}(1-p)^{w_r - s_m} \tag{5.8}$$

Conditional distribution $p(y_m|s_m)$ can be obtained by assuming certain distribution of the normalized energy estimator. Assuming that the estimator is unbiased and has Gaussian distribution with variance of σ^2,

$$p(y_m|s_m) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\left(y_m - s_m\right)^2}{2\sigma^2}\right)$$

This completes the calculation of $p^0_{m,n}$ in (5.1). Complementary probability of n^{th} variable node being 1 at m^{th} check node is simply $p^1_{m,n} = 1 - p^0_{m,n}$. With this, the message $u_{m\to n}$ from m^{th} check node to n^{th} variable node (i.e., the output of function $f\left(\circ\right)$ on Line 5 of Algorithm: MP with IPE assumption) is the log likelihood ratio $u_{m\to n} = \log\left(\frac{p^1_{m,n}}{p^0_{m,n}}\right)$.

The simulation result and its comparison with the theoretical performance bound is presented later on.

5.1.2 Message Passing Algorithm with Joint Probability Estimate (JPE)

As explained in previous section, the heart of the BP algorithm is calculation of function $f(\cdot)$ and in turn the probability calculations. This second algorithm is based on the joint probability estimate of all variable nodes connected to a check node given the discrete value of that check nodes. The algorithm provides the estimate of the all variable nodes for a given check node vector and sensing matrix as explained in theorem below.

Theorem Statement

A vector of N binary variable nodes $(b_1, b_2, ..., b_N)$ with known prior probabilities when connected to a check node (C), whose value (S_m) ranges between zero to the number of variable nodes connected to it $(i.e.\ N)$, then the probability of n^{th} variable node being zero or one is given by :

$$P(b_n = 0|S_m) = \begin{cases} p_T^1 - (S_m - 1) & \text{if, } S_m \in \{1, 2, \ldots, N-1\} \\ 1 & \text{if, } S_m = 0 \\ 0 & \text{if, } S_m = N \end{cases} \tag{5.9}$$

and,

$$P(b_n = 1|S_m) = 1 - P(b_n = 0|S_m) \quad \text{for, } S_m \in \{0, 1, 2, \ldots, N-1, N\} \tag{5.10}$$

Where, P_T^1 is the sum of the probabilities of all individual variable nodes being one which are connected to that check node except for which the probability is being calculated. It can be mathematically represented as:

$$P_T^1 = P_{b_1}^1 + P_{b_2}^1 + \cdots + P_{b_{n-1}}^1 + P_{b_{n+1}}^1 + \cdots + P_{b_N}^1 \tag{5.11}$$

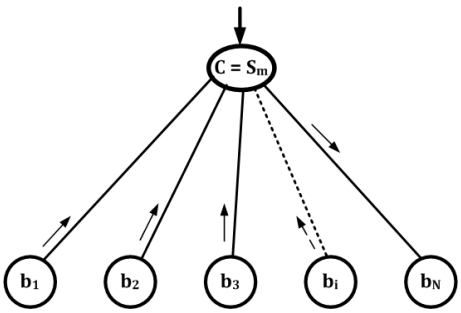


Figure 5.2: Connectivity graph for N variable nodes connected to a check node.

Proof of Theorem

With fixed number of variable nodes (N) connected to a single check node and with known check node value S_m, the possible states of variable node vectors connected to a check node can be shown as in Fig. 5.3. Each row represents a possible state of the variable node vector. The upper half represents all the variable node vectors with N^{th} variable node (b_N) equal to 0 and the lower half represents all variable node vectors with the N^{th} bit node b_N equal to 1 case.

b_N	b_{N-1}	b_{N-2}	...	b_3	b_2	b_1	
0	1	1	⋯	0	1	0	UPPER HALF
0	0	1	⋱	1	0	1	
0	1	0	⋱	0	1	1	
⋮	⋮	⋮	⋱	⋮	⋮	⋮	
0	0	1	⋱	1	1	0	
0	1	1	⋱	0	1	0	
1	0	0	⋱	1	0	1	LOWER HALF
1	0	1	⋱	1	0	0	
⋮	⋮	⋮	⋱	⋮	⋮	⋮	
1	1	1	⋱	0	0	0	
1	0	0	⋯	0	1	1	

Figure 5.3: Possible states of variable node vectors for a given N and S_m.

For a given S_m and N, number of variable node vectors with n^{th} variable node being 0 equals ${}^{N-1}C_{S_m}$ and number of variable node vectors with n^{th} variable node being 1 equals ${}^{N-1}C_{S_m-1}$.

In the case, for any N and $S_m = 0$, variable node vector results into all variable node equals to 0, i.e., $P(b_n = 0|S_m = 0) = 1$. Similarly, with $S_m = N$, it results

into only one variable node vector with all variable node equals to 1, i.e. $P(b_n = 0|S_m = N) = 0$.

For the cases, when check nodes takes value (S_m) other than 0 and N, $P_D^0 = P(b_n = 0|S_m)$ signifies the sum of joint probability of all variable node vectors when the desired variable node (b_n) equals 0 given the check node value S_m. Also, $P_D^1 = P(b_n = 1|S_m)$ represents sum of joint probability of all variable node vectors when the desired variable node (b_n) equals 1, given the check node value S_m.

From all the possible states of the variable node vectors (as is shown in the Fig. 5.3, number of possible combinations of variable nodes with i^{th} variable node equal to one (i.e. $(N1)_{b_i}$) equals ${}^{N-1}C_{S_m-1}$. Number of possible combinations of variable nodes vectors with i^{th} variable node being one and n^{th} variable node (b_n) equal to zero (i.e. $(N1)_{D_0,b_i}$) is given by ${}^{N-2}C_{S_m-1}$. Similarly, number of possible combinations of variable nodes vectors with i^{th} variable node being one and n^{th} variable node (b_n) equal to one (i.e. $(N1)_{D_1,b_i}$) is given by ${}^{N-2}C_{S_m-2}$. Therefore,

$$(N1)_{b_i} = (N1)_{D_0,b_i} + (N1)_{D_1,b_i} \tag{5.12}$$

Equation (5.11) can be written as:
P_T^1 = Joint probability of variable vector corresponding to $(N1)_{b_1}$+ Joint probability of variable vector corresponding to $(N1)_{b_2}$+ $\ldots$ + Joint probability of variable vector corresponding to $(N1)_{b_{N-1}}$.

The value of $(N1)_{b_i}$ remains constant for all b_i therefore the above equation can also be written as:

$$P_T^1 = (N-1)(N1)_{b_i}$$

Using (5.12) and the above described notations, we can write,

$$P_T^1 = (N-1)\left[{}^{N-2}C_{S_m-1} + {}^{N-2}C_{S_m-2}\right] \tag{5.13}$$

Multiplying and dividing the first term by S_m and second term by $(S_m - 1)$ in the (5.13), we get

$$P_T^1 = S_m\,{}^{N-1}C_{S_m} + (S_m - 1)\,{}^{N-1}C_{S_m-1};$$

Here,

${}^{N-1}C_{S_m}$ = Number of variable vectors with n^{th} variable node being 0 = P_D^0

${}^{N-1}C_{S_m-1}$ = Number of variable vectors with n^{th} variable node being 1 = P_D^1

$$P_T^1 = S_m P_D^0 + (S_m - 1) P_D^1 \tag{5.14}$$

and

$$P_T^0 = (N - S_m - 1) P_D^0 + (N - S_m) P_D^1 \tag{5.15}$$

P_T^0 can also be written as,

$$P_T^0 = (1 - P_{b_1}^1) + (1 - P_{b_2}^1) + \cdots + (1 - P_{b_{N-1}}^1) = (N - 1) - P_T^1$$

Therefore, (5.15) can be rewritten as,

$$(N - 1) - P_T^1 = (N - S_m - 1) P_D^0 + (N - S_m) P_D^1 \tag{5.16}$$

Solving (5.14) and (5.16) for P_D^0 we get,

$$P_D^0 = P_T^1 - (S_m - 1)$$

Therefore, for any N and S_m

$$P(b_n = 0 | S_m) = \begin{cases} p_T^1 - (S_m - 1) & \text{if, } S_m \in \{1, 2, \ldots, N-1\} \\ 1 & \text{if, } S_m = 0 \\ 0 & \text{if, } S_m = N \end{cases}$$

and

$$P(b_n = 1 | S_m) = 1 - P(b_n = 0 | S_m)$$

This completes the proof for joint probability calculation theorem described above.

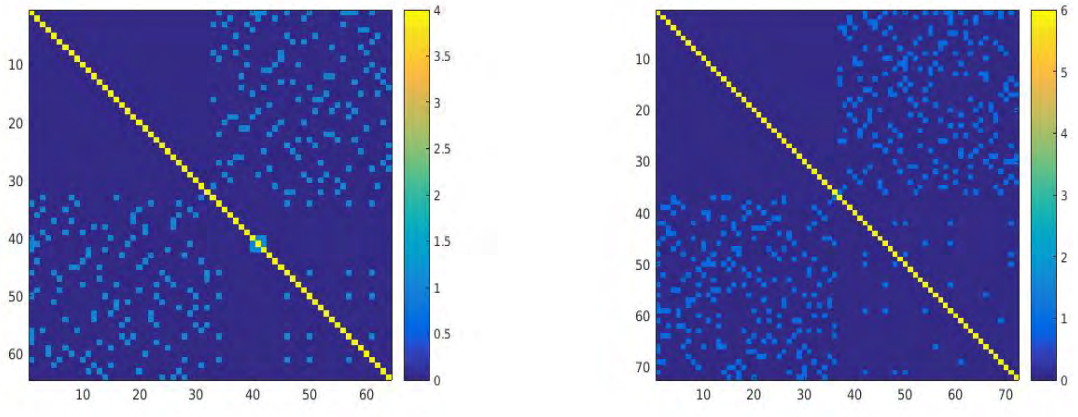
5.2 Simulation based Performance Verification

In this section, we provide a simulation based performance verification of the algorithms under different sparsity assumptions.

For all simulation work, parity check matrix or sensing matrix (**A**), is generated using the method described in [99], wherein cycles of length 4 and 6 is removed, therefore girth of the sensing matrix for this method is 8.

5.2.1 Characterization of Measurement Matrix

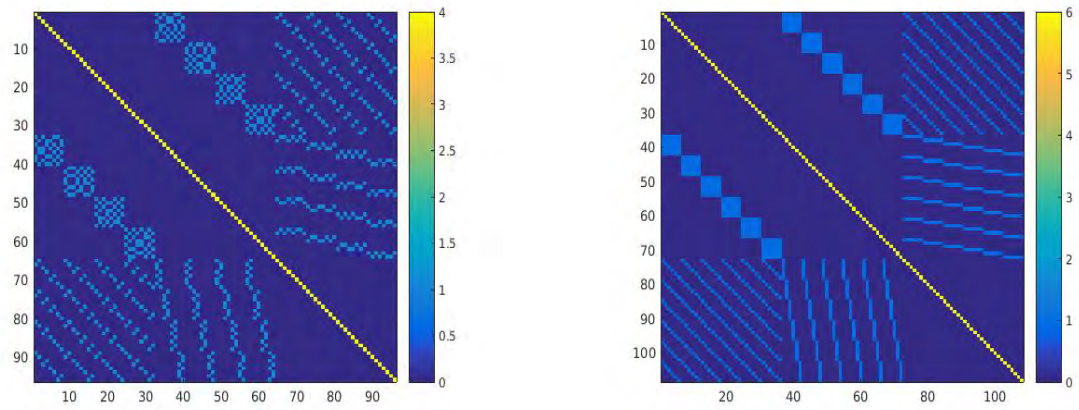
In order to ensure one to one correspondence between the measurement vector $\boldsymbol{y}$ and input vector $\boldsymbol{x}$ within CS framework, the sensing matrix should have low mutual coherence. The mutual coherence of sensing matrix $\boldsymbol{A}$ was verified through simulation by calculating their coherence matrix ($\boldsymbol{AA^T}$) for a given w_r. The diagonal elements of the calculated coherence matrix are equal to w_r values (for binary sparse matrix) and all the off diagonal values represent mutual coherence among columns of sensing matrix $\boldsymbol{A}$. Fig. 5.4 and Fig. 5.5 shows the coherence matrix of the generated sensing matrix using the two different methods.



(a) With $w_r = 4$, $L = 96$, $M = 128$.

(b) With $w_r = 6$, $L = 108$, $M = 216$.

Figure 5.4: Coherence measure for matrix A (Gallager method).



(a) With $w_r = 4$, $L = 96$, $M = 128$.

(b) With $w_r = 6$, $L = 108$, $M = 216$.

Figure 5.5: Coherence measure for matrix A (without cycles of length 4 and 6).

All the non-diagonal elements of the coherence matrix as shown in plot has very low value (≤ 1), which corresponds to very low mutual coherence value for given w_r and satisfies (1.8). For the mutual coherence value we need to divide each element of the matrix by w_c^2. This shows that the sensing matrix is suitable for CS implementation and also ensures one to one correspondence between measurement vector and input vector.

5.2.2 Simulation Results for IPE and JPE Algorithms

In this section, all the algorithms were simulated for the regular LDPC matrix based CS implementation. Measurement matrix **A**, is generated for different rates by varying w_r value while keeping w_c constant. The value of w_c is kept equal to 3 [99], where, w_c is the number of ones in each column of the parity check matrix and w_r is the number of ones in each row of matrix **A**. This generation algorithm generates a matrix such that the ratio of number of rows to that of number of columns is equal to $3/w_r$. For simulation, it has been assumed that noise is not present ($\sigma = 0$).

The objective of the sparsity detection algorithm is to estimate the locations of ones in vector **x** using the sensing matrix **A**. Number of ones in vector **x** is represented by probability p, which is varied from 0 to 1.

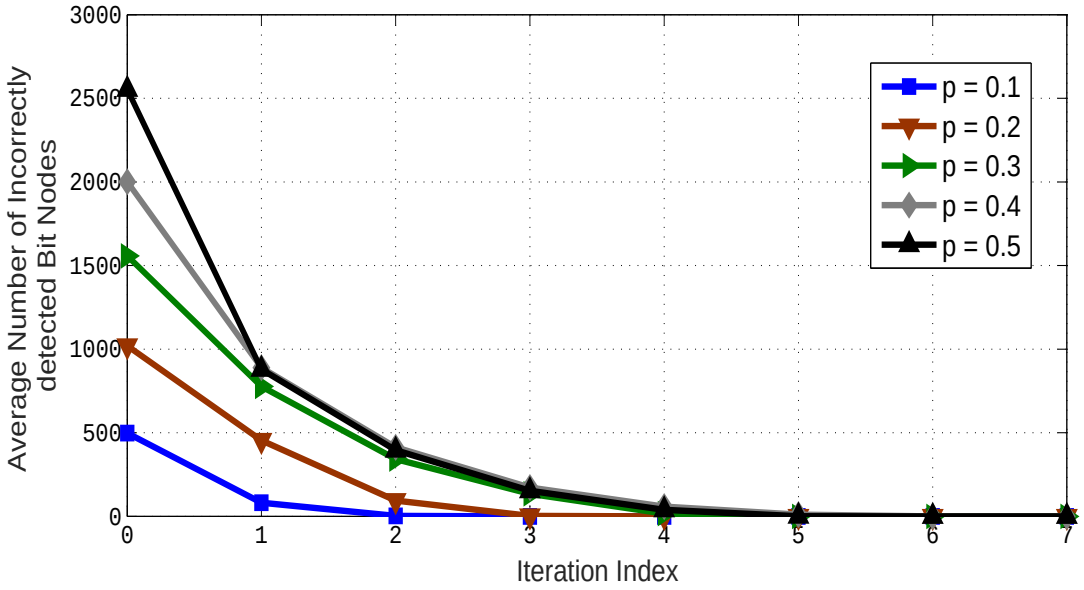


Figure 5.6: Convergence of IPE algorithm for different p (rate $= 0.75$ and $N = 5120$.

The algorithm convergence for different occupancy probability or sparsity with number of iterations is shown in Fig. 5.6. It is observed from the result that IPE takes longer time to converge as sparsity decreases or p_1 increases.

For the binary input case for the vector **x**, a performance comparison against theoretical bound for all sparsity levels is shown in the Fig. 5.7.

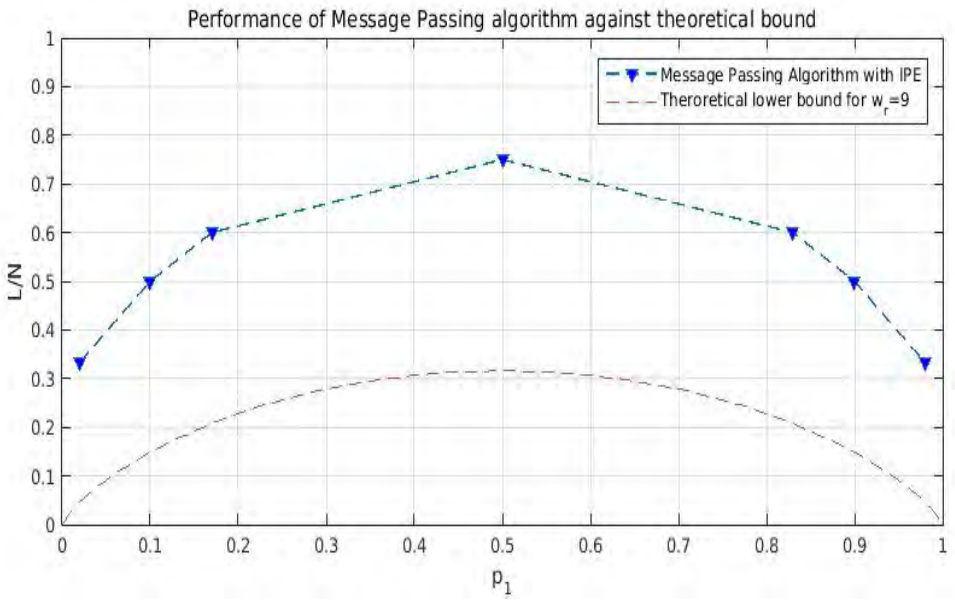


Figure 5.7: Performance comparison of IPE algorithm.

The Monte Carlo simulation (10000 simulation runs) of proposed message passing algorithm is done and the points where algorithm converged for all the trials are shown in the Fig. 5.7. Form this figure, it is seen that the algorithm works for all sparsity levels for rates greater than 0.75 indicating that the proposed algorithm and CS framework offers one fourth hardware savings under all sparsity level of input signal.

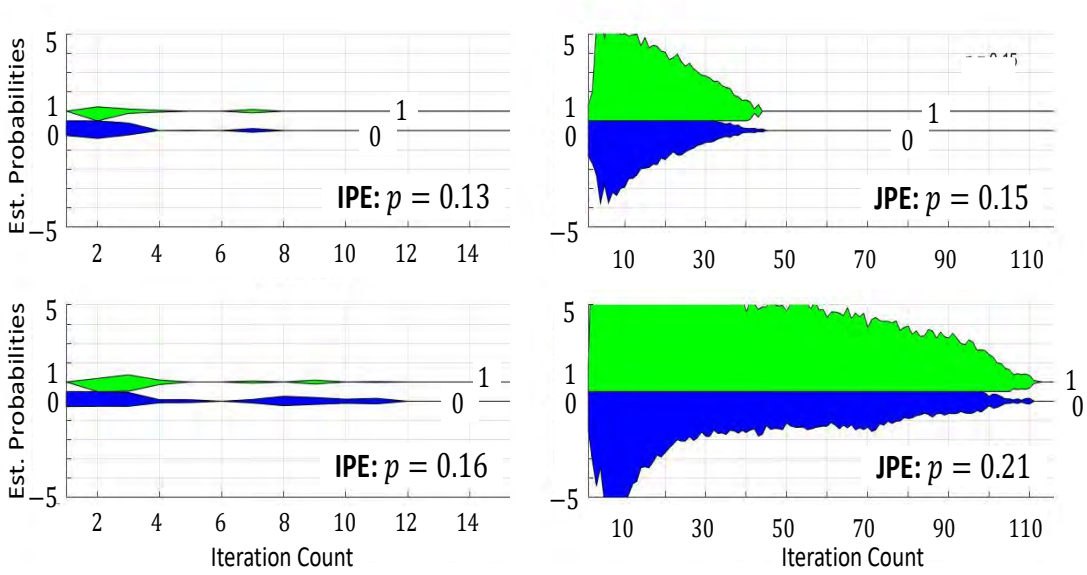


Figure 5.8: Convergence of the estimated probabilities of the variable nodes to 0 or 1 for IPE and JPE ($L/N = 0.5, N = 4958$).

The proposed JPE algorithm generates probability value outside the range of $[0, 1]$ due to initial values. Fig. 5.8 shows the range of probabilities generated during different iteration count for IPE and JPE. This eventually converges to either 1 or 0 with iteration. Table 5.1 shows the average residual estimation error for IPE and JPE when the measurement matrix $\mathbf{A}$ size is $L = 3840 \times N = 5120$. In this case, it is observed that JPE requires roughly three times more iteration than IPE. The performance of new algorithms (IPE & JPE) is simulated and their comparison with other developed algorithms reported in previous chapter such as Sudocode, Gallager (Hard Decision) for all sparsity levels are shown in Fig. 5.9.

Table 5.1: Average number of sparsity detection errors for IPE and JPE for increasing iteration count i_c when $L = 3840$ and $N = 5120$

i_c	p for IPE				i_c	p for JPE			
(IPE)	0.1	0.2	0.3	0.4	(JPE)	0.1	0.2	0.3	0.4
0	498	1020	1558	2553	0	512	1025	1538	2564.5
2	3	94	342	396	6	0.3	134.8	1066	1871.2
4	0	0	16	39	12	0	0	393	1247.3
6	0	0	0	0	18	0	0	17.1	683.3
8	0	0	0	0	24	0	0	0	171.3
10	0	0	0	0	30	0	0	0	1.3
12	0	0	0	0	36	0	0	0	0

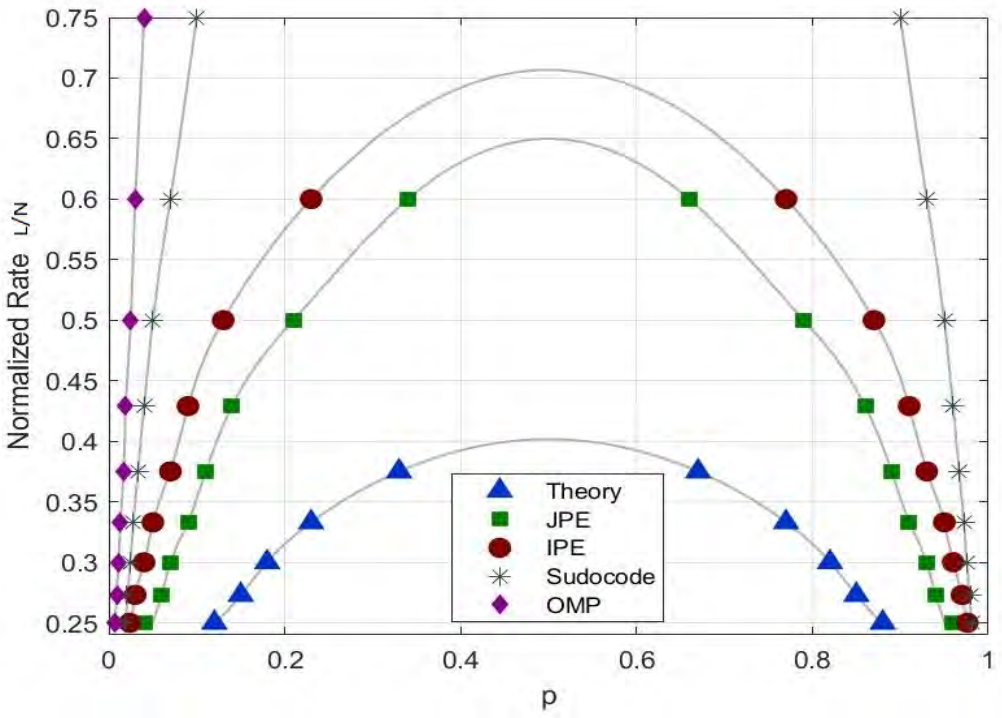


Figure 5.9: A simulation based comparison of sparsity detection algorithms.

The simulation result demonstrates that BP algorithm based on JPE is closer to theoretical lower bound and out performs other algorithms at all sparsity levels. Also, Sudocode and Gallager algorithms are suited only up to a specified sparsity levels and do not work for densely occupied channels. The simulated probability threshold values p^*, i.e, the largest probability value in the range of $[0, 0.5]$ at which the algorithm is successfully recovering the sparsity pattern for a given $L/N = 0.5$ is shown in Fig 5.10.

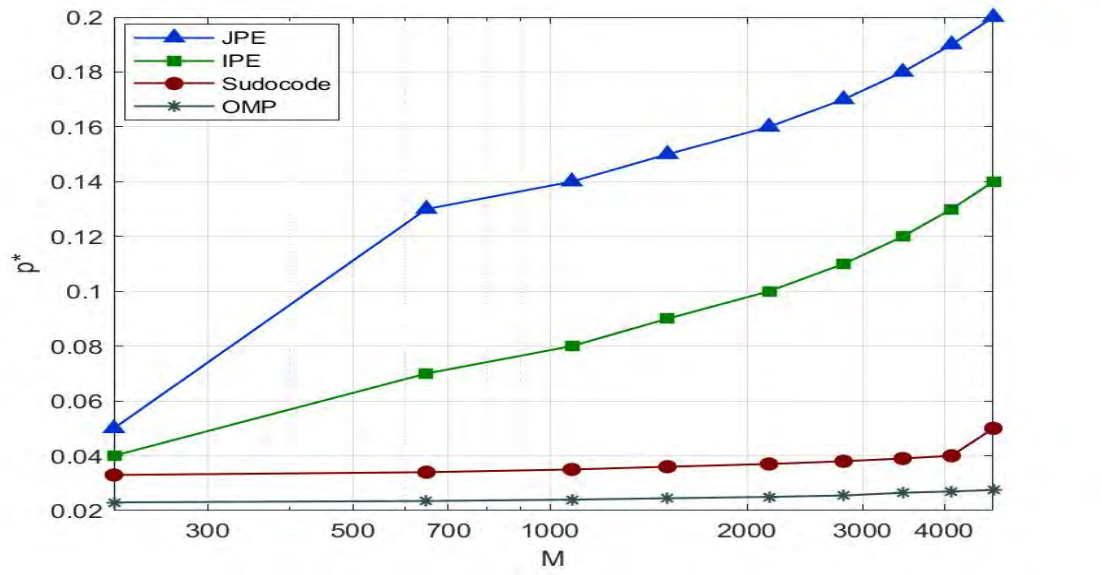
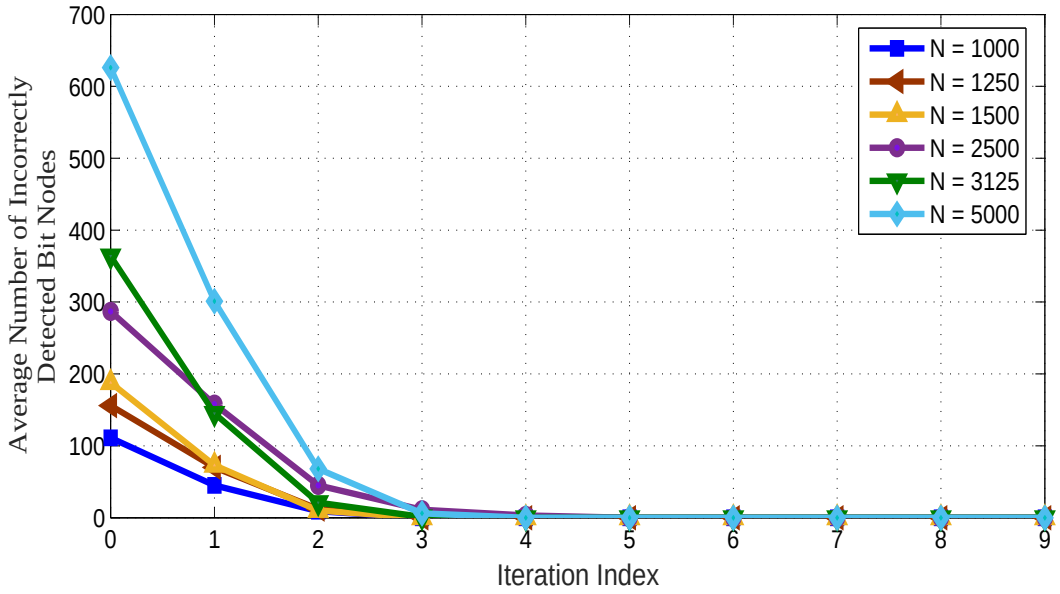
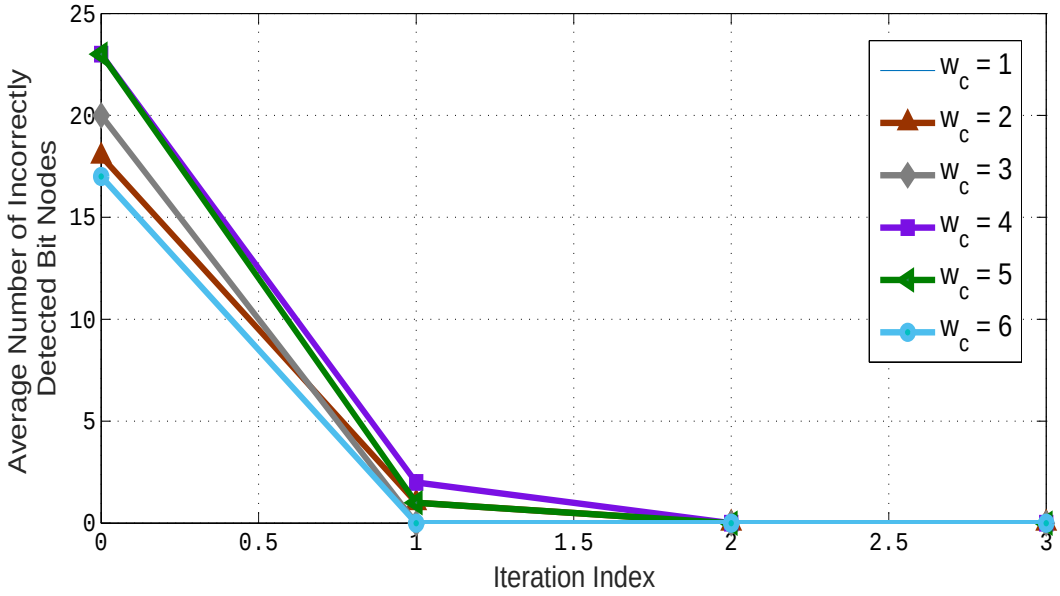


Figure 5.10: Threshold p^* for rate $L/N = 0.5$ with different values of N.

The algorithm performance was also simulated for different block lengths (or wideband communication with more number of channels) and different w_c values. Fig. 5.11 (a) shows the algorithm convergence plot for different block length keeping w_c, w_r and p_1 values fixed. Fig. 5.11 (b) shows the algorithm convergence for different w_c values.



(a) IPE convergence for different block length ($p_1 = 0.12$ and $L/N = 0.6$).



(b) IPE convergence for different w_c values ($p_1 = 0.01$, $N = 2187$ and $L/N = 0.33$).

Figure 5.11: IPE Convergence with different parametric variations.

The algorithm convergence is also observed for different block length. This indicates that proposed BP algorithms are suited for wideband communication with more number of channels. It is observed that simulation shows faster convergence for large value of w_c.

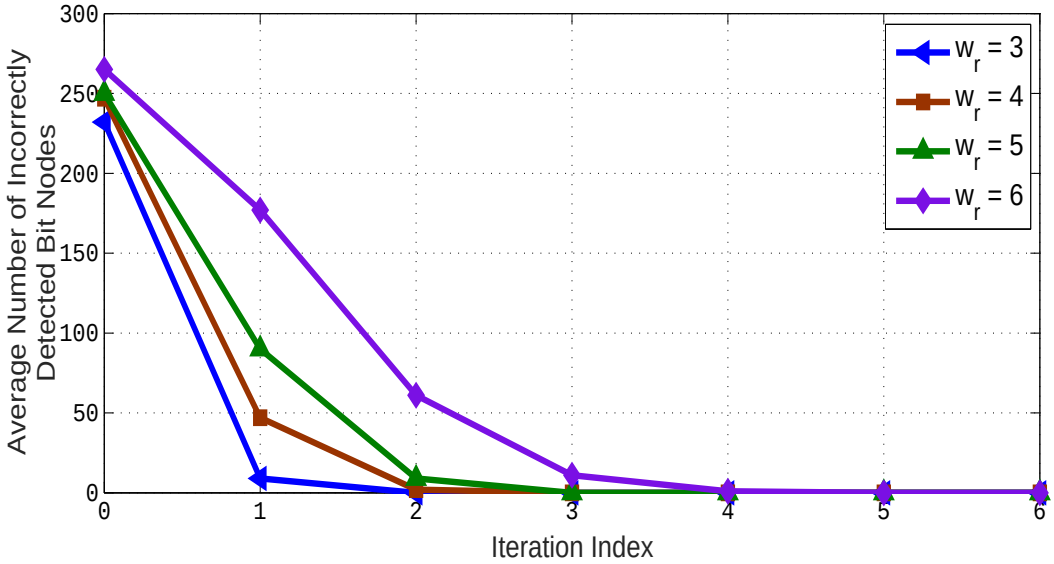


Figure 5.12: IPE convergence for different values of w_r ($w_c = 3$, $p_1 = 0.1$).

Fig. 5.12 shows the convergence plot for different w_r keeping w_c and p values constant.It is observed that more number of iterations are required for algorithm to convergence with increase in w_r value. Using Gallager method of matrix generation it has been verified that algorithm converges successfully for different w_r values.

Convergence analysis in presence of noise

In the assumed signal model, noise is classified depending upon their source into model noise $\boldsymbol{e_1}$ and measurement noise $\boldsymbol{e_2}$. The signal model can therefore be written as,

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x} + \boldsymbol{e_1}) + \boldsymbol{e_2} \tag{5.17}$$

For mathematical simplicity, the two noise vector e_1 and e_2 are considered to be taken as iid Gaussian with zero mean and variance σ_i^2, $i = 1, 2$. Convergence of proposed IPE and the JPE algorithms in presence of measurement noise ($\boldsymbol{e_2}$) and keeping model noise $\boldsymbol{e_1} = 0$ is shown in Fig. 5.13 and Fig. 5.14. This convergence analysis result shows that JPE requires more iterations than IPE to converge in presence of measurement noise.

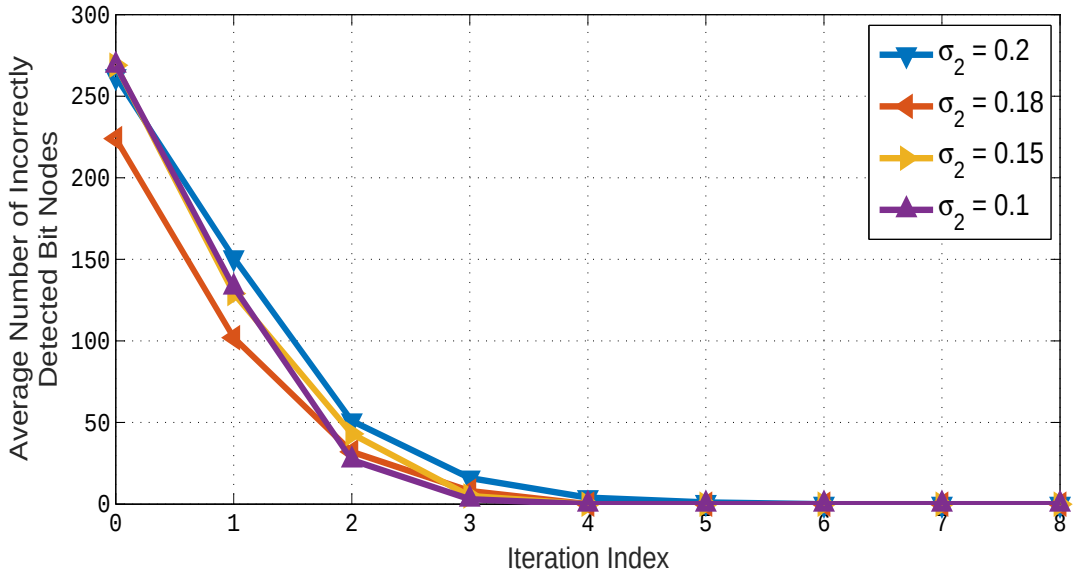


Figure 5.13: Convergence of IPE algorithm in presence of measurement noise (σ_2 is varied and $\sigma_1 = 0$, $p_1 = 0.2$, $N = 1280$ and rate $L/N = 0.75$).

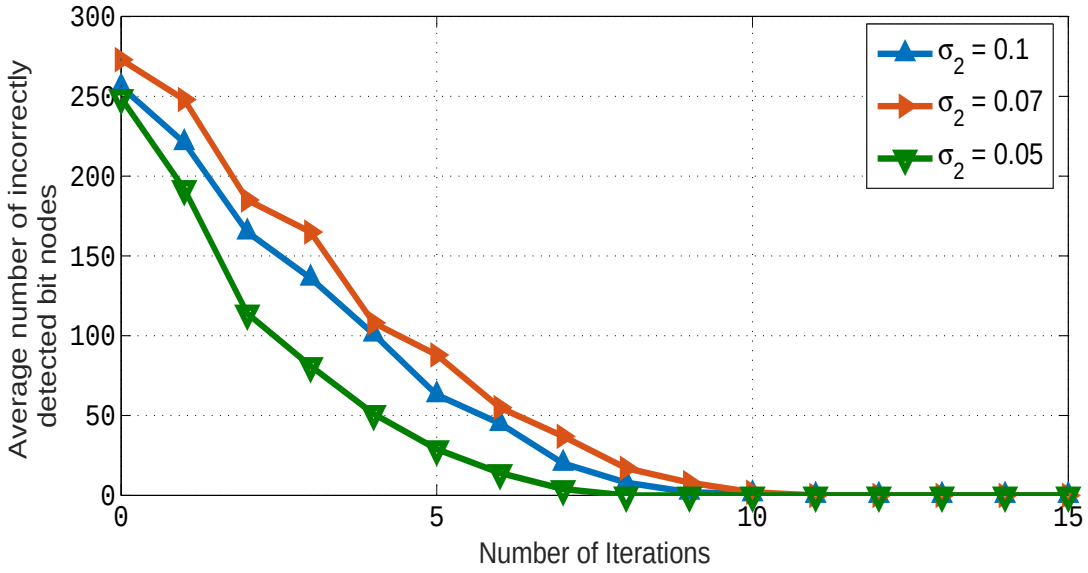


Figure 5.14: Convergence of JPE algorithm in presence of measurement noise (σ_2 is varied and $\sigma_1 = 0$, $p_1 = 0.2$, $N = 1280$ and rate $L/N = 0.75$).

Probability of success rate (correctly identifying the sparsity locations) of the proposed IPE and JPE algorithm is compared to the OMP algorithm for different cases of measurement and model noise.

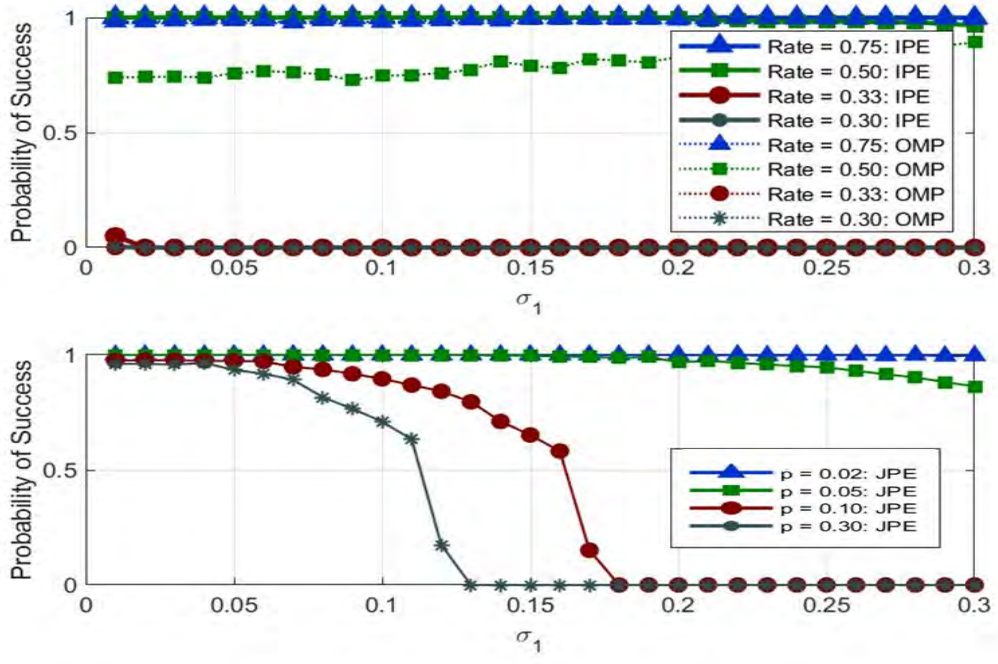


Figure 5.15: Success rate for different σ_1 ($\sigma_2 = 0$, $p = 0.04$ and $N \approx 1280$).

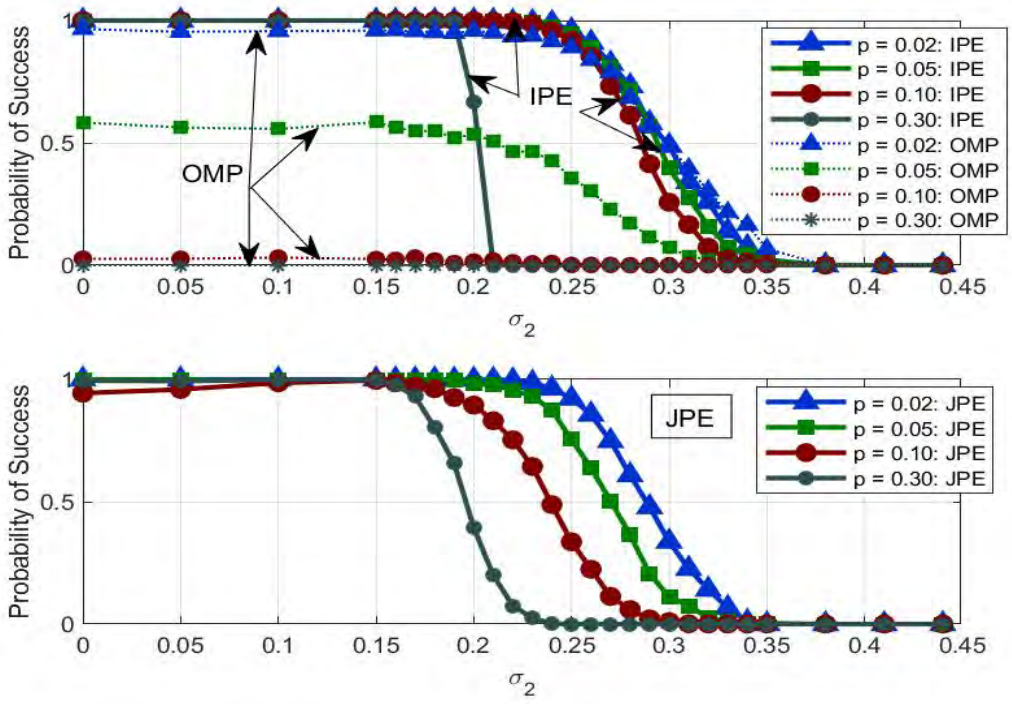


Figure 5.16: Success rate for different σ_2 and p ($\sigma_1 = 0.02, L = 960$ and $N = 1280$ and rate $L/N = 0.75$).

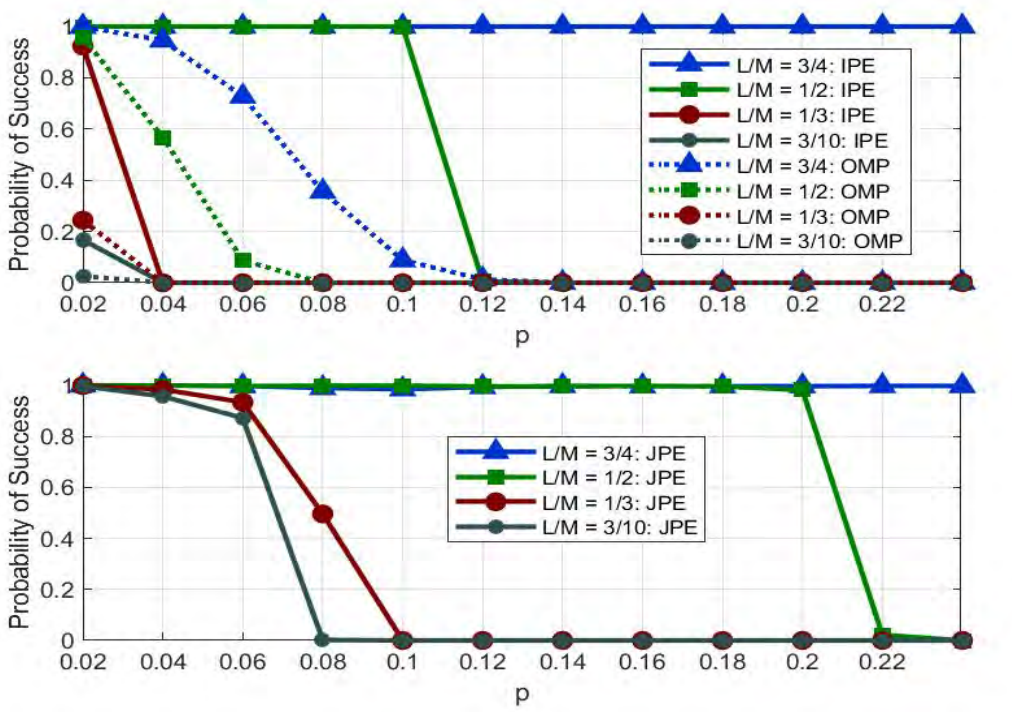


Figure 5.17: Success rate for varying p at different rates L/N ($\sigma_1 = 0.02, \sigma_2 = 0.04$ and $N \approx 1280$).

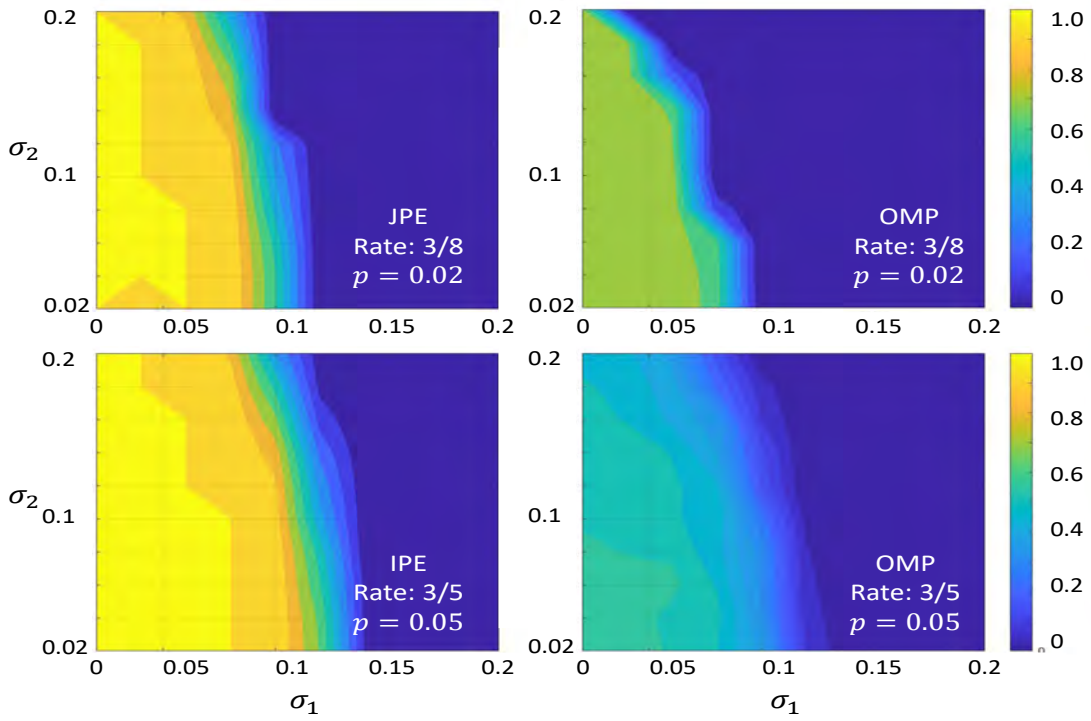


Figure 5.18: Success rate for varying σ_i, $i = 1, 2$ at different values of p and L/N.

Fig. 5.15, 5.16 and 5.17 shows the probability of success for different noise cases considered. The proposed algorithms is seen to outperform OMP algorithm for lower rates (i.e, lower L/N values). This is evident from Fig. 5.18. This completes the convergence analysis of the developed algorithm. The theoretical analysis matches the simulation results thus, confirming its applicability to sparsity detection task of CS based wideband communication applications.

5.3 Convergence Analysis using Density Evolution and Extrinsic Information Transfer (EXIT) Chart

Convergence analysis of the proposed algorithms (without noise) has also been conducted using density evolution approach as well as EXIT chart type of analysis [101, 105]. For this, the matrix $\mathbf{A}$ is randomly generated in each semi-analytic trial. The sparse binary vector $\mathbf{c}$ and the integer-valued measurement vector $\mathbf{w}$ are generated. The probability of error at the variable node is statistically evaluated at different iterations of the IPE algorithm. The average probability of error

at the variable nodes is used to derive the density evolution result and the extrinsic error probability transfer curve. Similar statistical analysis of JPE algorithm is conducted as well. The steps of convergence analysis using semi-analytical approach is explained in the given algorithm on Page 92.

Algorithm: To verify the convergence Analysis of the proposed algorithm

do

  for *all check node indices* $m \in \{1, \ldots, M\}$ **do**

    Randomly select a bit node connected to it

    if *corresponding element in the vector* $\boldsymbol{c}$ *equals* 0 **then**

      Calculate the probability of that bit node being 1, i.e calculate p_1

    else

      Calculate the probability of that bit node being 0, i.e calculate p_0

  The calculated probability value corresponds to the error of the algorithm over each iteration

while *For all iterations*

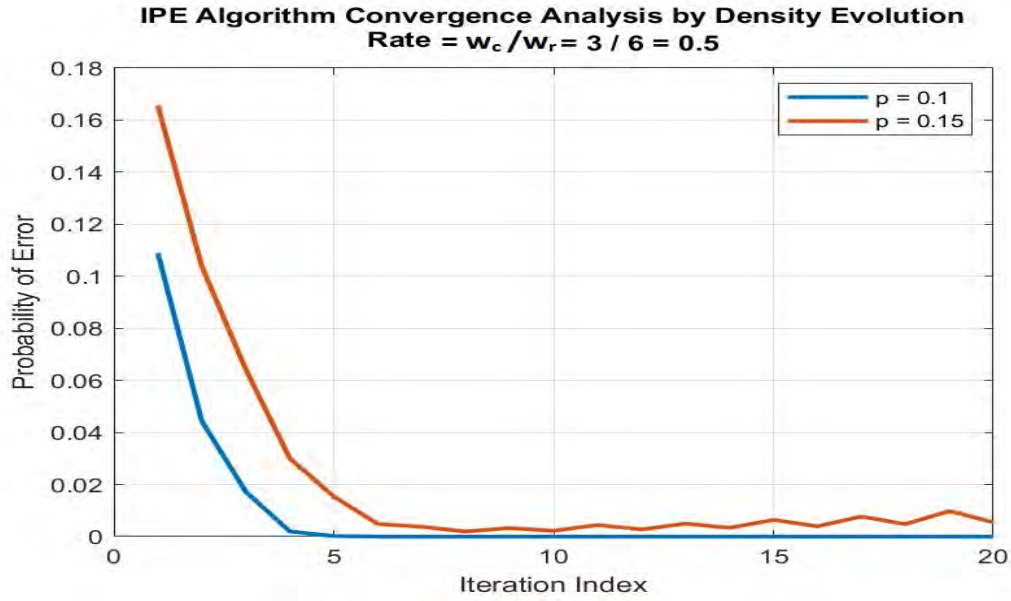


Figure 5.19: Statistical evaluation of convergence of IPE algorithm by density evolution method.

The simulation result for convergence analysis of IPE algorithm using density evolution approach is shown in Fig. 5.19. The simulation was conducted for a given rate with two different probability value and the algorithm convergence was in accordance with algorithm simulation results.

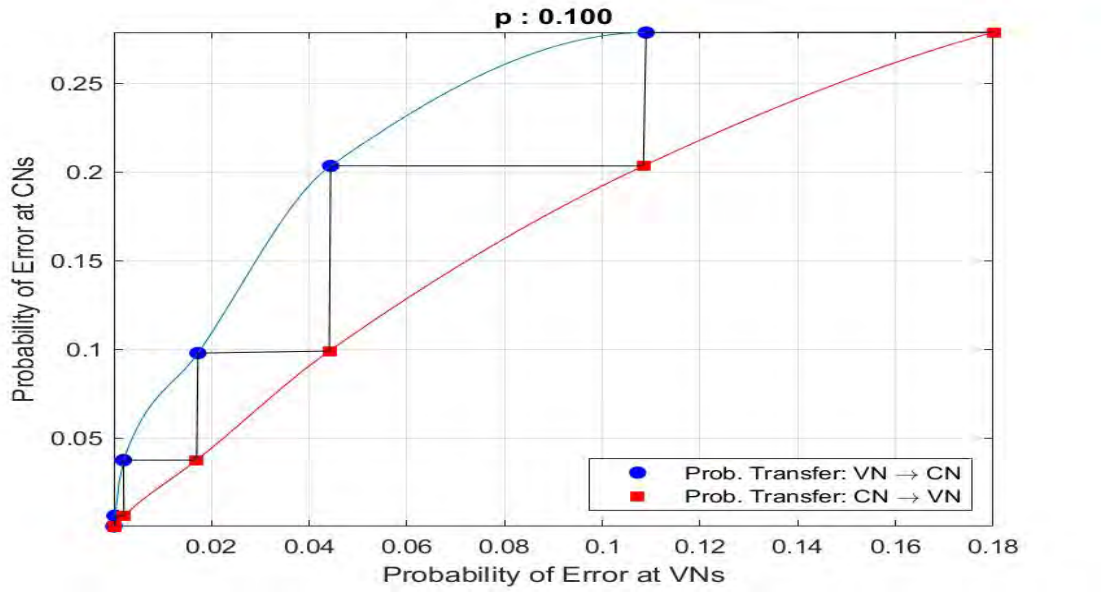


Figure 5.20: Statistical evaluation of convergence of IPE algorithm by EXIT chart method ($p = 0.1; rate = 1/2$.)

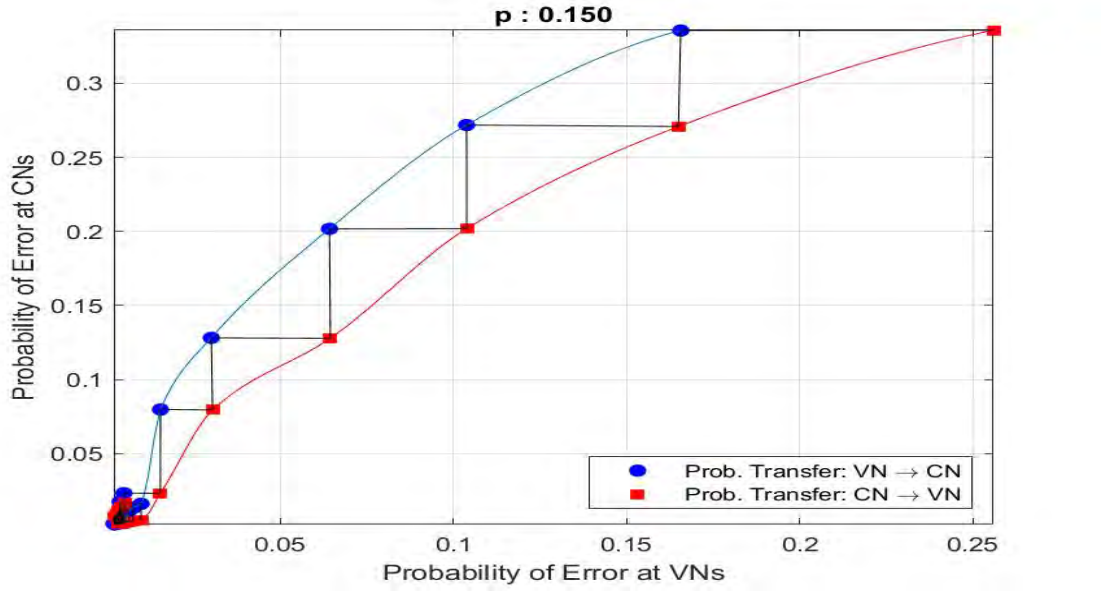


Figure 5.21: Statistical evaluation of convergence of IPE algorithm by EXIT chart method ($p = 0.15; rate = 1/2$.)

The simulation result for convergence analysis of IPE algorithm using EXIT

Chart approach is shown in Fig. 5.20 and Fig. 5.21. The analytical convergence analysis result matches the algorithm simulation results.

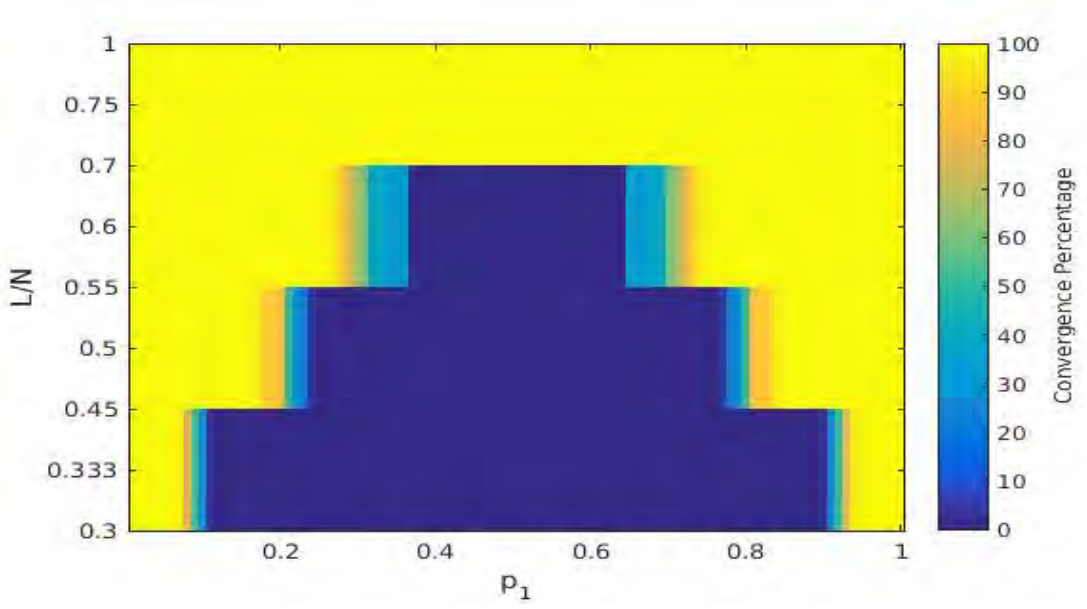


Figure 5.22: Convergence of the proposed IPE algorithm (Heat Map).

Additionally, a Monte Carlo simulation for size of measurement matrix (L/N) vs sparsity level (p_1) using semi-analytical approach was performed. The p_1 value was varied in steps of 0.001 and a total of 1000 trials was simulated. The out of semi-analytical convergence is shown in Fig. 5.22 in the form of heat map. The heat map provides the sparsity bound for a given size of measurement matrix to decide on hardware for a given applications.

5.4 Computational Complexity of BP Algorithms

An approximate evaluation of the computational cost of IPE and JPE is summarized in Table 5.2. The computational complexity of proposed algorithms are compared with OMP.

For each variable node (VN) of degree w_c, both the algorithms require $w_c\,(w_c-1)$ computations (refer Algorithm 1 on Page 74). Furthermore, corresponding to each VN, there is L/M CNs requiring $w_r\,(w_r-1)\,\beta$ computations. Here β is a scalar multiplier that accounts for conditional checks implemented at

Table 5.2
Computational Cost per Bit of IPE, JPE and OMP

I/J-PE	OMP
$i_{c,\perp}\left(\beta(w_r-1)w_r\frac{L}{N}+(w_c-1)w_c\right)$	$\left(1+\frac{L}{N}\right)\kappa(\kappa+1)/2$

the CN (refer Algorithm 1 description on Page 74) for both IPE and JPE, and additionally the binary bit string listing required for the IPE, which is on the order of $\binom{w_r}{s_m}$ for a CN with weight s_m. Table 5.2 also lists, from [106, Table 1], the computational cost of an efficient implementation (based on the Matrix Inversion Lemma) of the OMP.

5.5 Extension of Sparsity Detection Algorithms

This section presents two extensions of sparsity detection algorithms. The first deals with extension of sparsity detection algorithm when the input to the algorithm is discrete non-binary and second one is a deterministic algorithm for sparsity detection in noiseless case more suited for source coding and decoding application.

5.5.1 Algorithm for Discrete Non-binary Input

This sparsity detection algorithm refers to a CS problem $\mathbf{y} = \mathbf{A}\mathbf{x}$, the sparse signal $\mathbf{x}$, which is to be determined, have elements from the discrete set $\{0, 1, 2\}$, representing different energy levels of different input bands. The analogy could be similar to use of different waveform (modulation and coding) in different communication channels. The proposed method relies on the message passing algorithm for decoding the elements of the sparse signal $\mathbf{x}$ from the received signal $\mathbf{y}$.

Algorithm with IPE for non-binary case

Message Passing algorithm for decoding the elements in the sparse vector $\mathbf{x}$ is described below:

- Messages from the variable node to the check nodes are initialized by an ar-

ray of log likelihood ratio (LLR) for the different possible elements in vector **x**. In this case an array of LLR for the elements being zero, one or two is passed in the initialization step.

- Message from the check nodes to the variable nodes are calculated using a function $f(.)$. A detailed explanation about the function, $f(.)$ is given below.
- Variable nodes then calculates sum of the messages received from the neighbor check nodes and send it back to the check node.

Function $f(.)$ generates log likelihood ratio of the n^{th} variable node being zero, one or two, given the probabilities of its neighbor variable nodes connected to m^{th} check node. Let $p^i_{m,n}$, where $i = 0, 1, 2$ denotes the probability that the n^{th} variable node is having value i given the probabilities of all the variable nodes in the set $C_{m,n}$. Using Bayesian rule one can write:

$$p^i_{m,n} = \sum_{s_m=0}^{2w_r} p^i(u(m,n)|s_m)p(s_m|y_m) \qquad i = 0, 1, 2 \tag{5.18}$$

Here, $p^i(u(m,n)|s_m)$ is the conditional probability of n^{th} variable node being i given a specific $s_m \epsilon \{0, 1, 2, .., w_r, w_r + 1, .., 2w_r\}$ and $p(s_m|y_m)$ denotes the conditional probability of s_m given the measured value of y_m.

Conditional probability $p^i(u(m,n)|s_m)$, can be calculated as being proportional to the probability of the bit sequence set $p(\mathbf{b}^{w_r-1,s_m-i})$ where,

$$\mathbf{b}^{w_r-1,s_m-i} = \{\mathbf{b}_0^{w_r-1,s_m-i}, \mathbf{b}_1^{w_r-1,s_m-i}, \mathbf{b}_2^{w_r-1,s_m-i}, .., \mathbf{b}_L^{w_r-1,s_m-i}\} \tag{5.19}$$

Each sequence in the set is of length $w_r - 1$ and sum of each element in the sequence equals $s_m - i$. Number of such bit sequence for a given sum is given by L. An example of a set of such bit sequences for $w_r = 3$ is illustrated in the table on Page 97, where each row corresponds to a bit sequence.
Conditional probability $p^i(u(m,n)|s_m)$ is obtained by the following ratio:

$$p^i(u(m,n)|s_m) = \frac{p^i_{m,n}(\mathbf{b}^{w_r-1,s_m-i})}{\sum_{j=0}^{2} p^j_{m,n}(\mathbf{b}^{w_r-1,s_m-j})} \qquad i = 0, 1, 2 \tag{5.20}$$

bit 1	bit 2	bit 3	Sum
0	0	0	0
1	0	0	1
2	0	0	2
0	1	0	1
1	1	0	2
2	1	0	3
0	2	0	2
1	2	0	3
2	2	0	4
0	0	1	1
1	0	1	2
2	0	1	3
0	1	1	2
1	1	1	3
2	1	1	4
0	2	1	3
1	2	1	4
2	2	1	5
0	0	2	2
1	0	2	3
2	0	2	4
0	1	2	3
1	1	2	4
2	1	2	5
0	2	2	4
1	2	2	5
2	2	2	6

Table 5.3: Set of bit sequence for $w_r = 3$.

Here, $p^i_{m,n}(\mathbf{b}^{w_r-1,s_m-i})$ denotes the total probability of the set $\mathbf{b}^{w_r-1,s_m-i}$ at m^{th} check node for n^{th} variable node to have value i, and it is the sum of probabilities of individual member sequences of the bit set.

$$p^i_{m,n}(\mathbf{b}^{w_r-1,s_m-i}) = \sum_{j=0}^{L} p_{m,n,i}(\mathbf{b}_j^{w_r-1,s_m-i}) \tag{5.21}$$

Due to the iid assumption of the variable nodes in the set $C_{m,n}$, the probability $p_{m,n,i}(\mathbf{b}_j^{w_r-1,s_m-i})$ in the equation (5.21) can be written as the product of the indi-

vidual bit probability in the sequence $\mathbf{b}_j^{w_r-1,s_m-i}$.

$$p_{m,n,i}(\mathbf{b}_j^{w_r-1,s_m-i}) = \prod_{k=1}^{w_r-1} p'_{m,n'}(\mathbf{b}_{j,k}^{w_r-1,s_m-i}) \tag{5.22}$$

Here, $\mathbf{b}_{j,k}^{w_r-1,s_m-i}$ denotes k^{th} bit of the j^{th} bit sequence of the set $\mathbf{b}^{w_r-1,s_m-i}$. Variable n' in equation (5.22) denotes k^{th} element in the set $C_{m,n}$, and $p'_{m,n'}(\mathbf{b}_{j,k}^{w_r-1,s_m-i})$ denotes the probability of that variable node.

Derivation of second term in the (5.18), $p(s_m|y_m)$, is described in the following section. s_m denotes the integer value which the m^{th} check node can take. In this case $s_m \epsilon \{0, 1, 2, .., 2w_r\}$. Measured value y_m of m^{th} check node have a probability distribution on s_m, which is given by the Bayesian formula $p(s_m|y_m) = p(y_m|s_m) / \sum_{s_m=0}^{2w_r} p(y_m|s_m)p(s_m)$. Given the iid assumption regarding the elements of $\mathbf{x}$, the prior probability $p(s_m)$ is given by Binomial Probability Mass Function (PMF):

$$p(s_m) = \binom{2w_r}{sm} p^{s_m}(1-p)^{2w_r-s_m} \tag{5.23}$$

Conditional distribution $p(y_m|s_m)$ can be obtained by assuming that the energy estimator has Gaussian distribution with variance σ^2,

$$p(y_m|s_m) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_m - s_m)^2}{2\sigma^2}\right)$$

This completes the calculation of $p_{m,n}^i$ in (5.18).

Above algorithm is repeated to calculate $p_{m,n}^0$, $p_{m,n}^1$ and $p_{m,n}^2$. The message $u_{m\to n}$, from m^{th} check node to n^{th} variable node is an array comprising of $\log\left(\frac{p_{m,n}^0}{p_{m,n}^1 + p_{m,n}^2}\right)$, $\log\left(\frac{p_{m,n}^1}{p_{m,n}^0 + p_{m,n}^2}\right)$ and $\log\left(\frac{p_{m,n}^2}{p_{m,n}^0 + p_{m,n}^1}\right)$.

Simulation Result: Sparsity Detection with non-binary discrete input

The outcome of simulation based performance verification of IPE based sparsity algorithm with 3-ary input elements (0,1,2) is shown in Fig. 5.23.

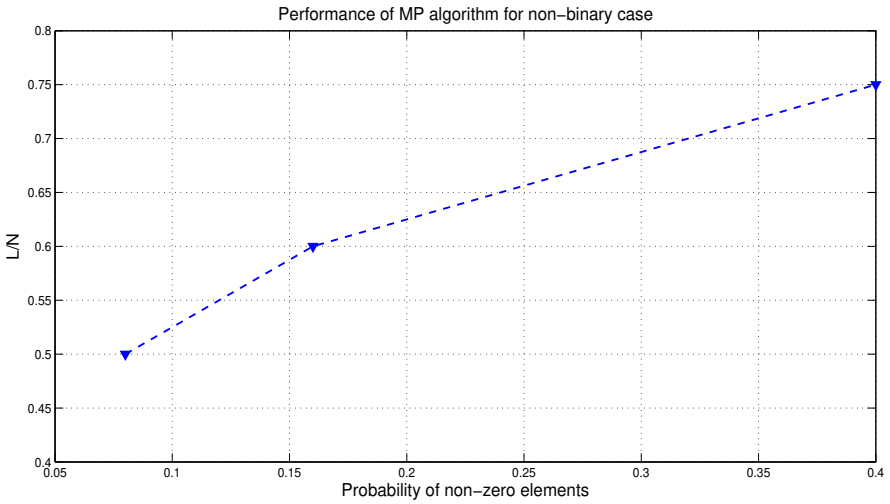


Figure 5.23: Simulation result for sparsity detection with non-binary discrete inputs.

The proposed algorithm was simulated for different rates and for a range of probabilities of non-zero elements (p_1 of the input signal $\mathbf{x}$). The non-zero probability (p_1) are equally divided between two discrete input levels (x being one or two). The plot shown in Fig. 5.23 is for p_1 vs rate. The result shows that the IPE algorithm works for $p_1 = 0.4$ (probability of channel being occupied with normalized energy as 1 is 0.2 and probability of channel being occupied with normalized energy as 2 is 0.2) for rate $\frac{3}{4}$. This represents that when sparsity is 0.4 (half of the occupancy is with signal energy twice than that of remaining half), then hardware equivalent to rate $\frac{3}{4}$ will successfully detect the support of occupied slots blindly using IPE algorithm in given CS framework.

The algorithm can further be extended for a larger discrete set (in-turn for real value bit nodes). Enhancement of the algorithm to achieve performance closer to theoretical bounds to be worked out.

5.5.2 Deterministic Algorithm for Noiseless Case

A deterministic algorithm for sparsity detection was developed for noiseless case with input having Bernoulli distribution. For sparsity detection algorithm, energy present in each band would correspond to bit node value and output of energy detector would correspond to check node value for given measurement (LDPC

like) matrix. This algorithms aims at use cases of source coding and decoding.

Notations

Following the standard convention used in description of matrix multiplication, y_m denotes m^{th} element of column vector or measurement y. Similarly, b_n denotes n^{th} element of column vector b or bit node or variable node vector. C_m denotes a set of indices of variable nodes connected to m^{th} check node, $C_{m,n} = C_{m\backslash n}$ is the set of elements of C_m whose values are determined as equal to 0 or 1. E_{nz} contains non-zero elements of error vector e.

Algorithm Description

The deterministic algorithm is listed below and can be described as:

- Conditions for element (b_n) of the received vector being zero or one is checked and accordingly all the connected bit nodes are set as 0 or 1.
- If the difference of the element of received vector y_m and the sum of elements in $C_{m,n}$ is zero then, substitute b_j with one $\forall j \in C_j - C_{m,n}$.
- Substitute $c_{est} = b$ and then calculate error vector $e = y - Hc_{est}$.
- Make a vector consisting of all bit nodes connected to the non-zero elements of vector e. Randomly pick an element from that vector e and replace it by one.
- Re-calculate error vector e. If the number of non-zero elements in the vector decreases then keep the change done in the previous step otherwise discard it.

Simulation Results

The proposed deterministic algorithm was simulated for different size of measurement matrices (rates) and for a range of probabilities of non-zero elements in the sparse signal $\mathbf{x}$. The $\mathbf{H}$ matrix implemented is same as other BP algorithms.

Algorithm: Deterministic Algorithm

for *all check node indices* $m \in \{1, \ldots, M\}$ **do**
  if $y_m == 0$ **then**
    $b_n \leftarrow 0, \forall n \in C_m$
  else if $y_m == w_r$ **then**
    $b_n \leftarrow 1, \forall n \in C_m$
  else if $(y_m - \sum_{j \in C_{m,n}} b_j) == 0, \forall j \in C_{m,n}$ **then**
    $b_j \leftarrow 1, \forall j \in C_j - C_{m,n}$
With $c_{est} = b$, Calculate $e = y - Hc_{est}$
do
  for *all* $m \in E_{nz}$ **do**
  randomly pick an element $rn \in E_{nz}$ and replace b_{rn} with 1.
  if *Recalculate e. If number of non-zero elements decreases in e* **then**
    $b_{rn} \leftarrow 1$
  else
    $b_{rn} \leftarrow 0$

while *number of iterations is less than a threshold*

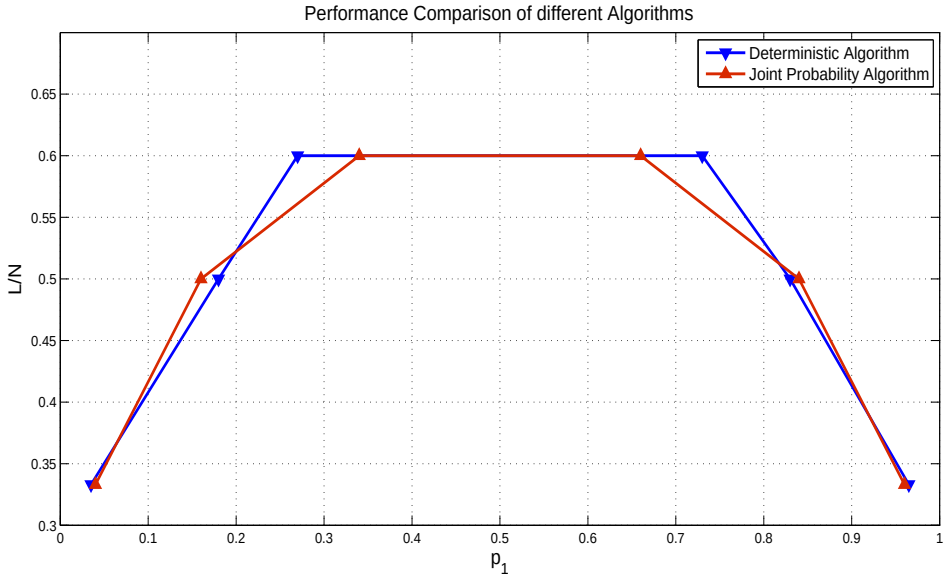


Figure 5.24: Performance comparison of deterministic algorithm with JPE.

Fig. 5.24 shows a performance comparison of the deterministic algorithm with that of the JPE algorithm for different size of measurement matrices or rates. The Deterministic algorithm exhibits comparable or better performance in regions of low sparsity than JPE. The drawback of this algorithm is that it does not account for noise at the input and conceived only for noiseless case.

Summary: A detailed performance analysis of developed BP algorithms, their convergence analysis and performance comparison against theoretical lower bounds has been described in this chapter. The discretized sensing framework enabled by sparsity detection algorithms proved to achieve at least one fourth hardware savings by keeping the rate at $\frac{3}{4}$ for all sparsity levels. This makes the proposed framework very useful for wideband communication applications.

We also developed a deterministic light weight algorithm for sparsity detection for noiseless case. This algorithm offers performance close to JPE algorithm and can be gainfully utilized for source decoding.

The next chapter provides an analytical framework called Low Density Integer Constrained (LDIC) codes for lossless data compression and denoising applications. The proposed LDIC framework uses deterministic as well as message passing algorithms for source decoding.

CHAPTER 6

Low Density Integer Constrained (LDIC) Codes

Implementation of sparse binary LDPC matrix using RF front-end architecture and belief propagation based sparsity detection algorithms provides a means of CS framework realization. In this chapter, a new generalization of the LDPC codes that is termed as *Low Density Integer Constrained (LDIC)* codes is proposed to be used for lossless data compression as well as for joint compression and denoising applications.

6.1. The Concept of LDIC Codes

In LDIC codes, similar to the standard application of LDPC matrices, the proposal is to multiply an $L \times N$ binary-valued LDPC matrix $\mathbf{H}$ with an $N \times 1$ binary-valued vector $\mathbf{c}$. However, unlike the standard LDPC codes, we do not perform modulo operation on the product and maintain the integer-valued vector

$$\mathbf{w} = \mathbf{Hc}. \tag{6.1}$$

Elements of $\mathbf{w}$ take an integer value from $\{0, 1, \ldots, d_c\}$, where d_c denotes the degree of the check nodes. Here, matrix $\mathbf{H}$ is taken as a regular LDPC matrix and d_c is same as w_r (= number of ones in any row) of regular LDPC matrix.

The proposal of the LDIC codes, and the interest in studying them, arose from the CS context. In the CS, LDPC matrices are used with real-valued sparse excitation in the real-number field [36]. The goal is to determine a sparse vector

$\mathbf{x}$ of size $N \times 1$ (which has $n_0 << N$ nonzero elements randomly interspersing $N - n_0$ zeros) given an $L \times 1$ measurement vector $\mathbf{y} = \mathbf{H}\,\mathbf{x}$. Here, $L \ll N$, and $\mathbf{H}$ is known as measurement matrix. One approach of solving this problem identifies the sparsity locations, say, as a binary indicator vector $\mathbf{c}$ of size N. This vector $\mathbf{c}$ has n_0 ones whose locations coincide with n_0 nonzero elements of $\mathbf{x}$. Given $\mathbf{c}$, the locations of zero-valued elements of $\mathbf{x}$ are known. This allows removing $N - n_0$ columns of $\mathbf{H}$ that multiply with zero-valued elements of $\mathbf{x}$. With this, the original CS measurement matrix $\mathbf{H}$ is reduced to a matrix $\mathbf{H}_S$ of size $L \times n_0$. Assuming that these columns of $\mathbf{H}_S$ are linearly independent, the standard Least Squares solution (i.e., the pseudoinverse of $\mathbf{H}_S$) is used to reconstruct the original sparse excitation pattern $\mathbf{x}$ from the measurement vector $\mathbf{y}$ [17].

The main idea of LDIC codes is to measure $\mathbf{w} = \mathbf{H}\mathbf{c}$ in addition to $\mathbf{y} = \mathbf{H}\,\mathbf{x}$. Whenever, this measurement of $\mathbf{w}$ in equation (6.1) is possible, i.e., the binary sparsity pattern $\mathbf{c}$ inherent in the non-binary real-valued $\mathbf{x}$ can itself be measured through the measurement matrix, the CS problem reduces to that of lossless data compression, i.e., how to maximally compress a sparse binary vector $\mathbf{c}$ using the measurements $\mathbf{w}$ ($L \ll N$). This proposed approach does not perform the modulo operation, and restrict to GF(q), it is better suited to the task of data compression. This is analysed in next section.

6.1.1 Source Encoding

As in (6.1), we apply the binary-valued LDPC matrix $\mathbf{H}$ at the source encoder. The input to the source encoder (i.e. input to $\mathbf{H}$) is a discrete-valued[1] vector. The output of the source encoder, i.e., the result of multiplication of the input vector with LDPC matrix $\mathbf{H}$, is non-binary; modulo operation is not performed on the output of this multiplication. The LDIC code characterization presented in [42] provides motivation for not restricting the output of (6.1) to GF(2).

Analysis: The discrete-valued vector of symbols at the input to the source encoder is denoted as $\mathbf{c}$. The output of the source encoder is $\mathbf{w} = \mathbf{H}\,\mathbf{c}$. Unlike the application of LDPC matrices in the FEC, the output $\mathbf{w}$ takes values from the set

[1]For the sake of simplicity in the analysis and exposition, binary-valued case is focused.

$\{0, 1, \ldots, d_c\}$, where d_c denotes the degree of the check nodes. This is assumed to be identical (i.e., matrix $\mathbf{H}$ is a regular LDPC matrix). Thus, matrix $\mathbf{H}$ transforms a length N vector $\mathbf{c}$ into a length $L < N$ discrete-valued vector $\mathbf{w}$.

We consider the compression problem for binary random variable C, with probability of one denoted as p. For sake of simplicity, we denote an $N \times 1$ vector of iid variables C as $\mathbf{c}$.

Specifically, we use the set of nonbinary constraints obtained as $\mathbf{w} = \mathbf{Hc}$ as the compressed version of the binary data. Unlike channel coding application there is no modulo-two operation taken here.

Our proposal for not taking modulo-two operation (GF(2)) is motivated as follows.

We define the set of binary sequences each of which produce the integer-valued output $\mathbf{w}$ when applied as the input to a system $\mathbf{H}$ of parity check equations as $\mathcal{C}_{(\mathbf{H},\mathbf{w})}$. For each $\mathbf{c} \in \mathcal{C}_{(\mathbf{H},\mathbf{w})}$, $\mathbf{H}\,\mathbf{c} = \mathbf{w}$.

Consider a ratio Q_p defined as follows:

$$Q_p = \frac{H(C)}{H(W)} \tag{6.2}$$

where $H(C)$ and $H(W)$ are the Shannon Entropy values of the input variable C (elements of the input vector $\mathbf{c}$ are drawn from C where probability of 1 is p), and output variable W (elements of the output vector $\mathbf{w}$ are variables W).

From information theory, the ratio L/N of the length L of the compressed nonbinary string $\mathbf{w}$ to the length of the binary string $\mathbf{c}$ is lower bounded by Q_p, i.e.,

$$L/N \geq Q_p. \tag{6.3}$$

As an example, consider a case when the input variable C is a non-compressible Bernoulli RV and $\log_2(d_c + 1)$ is an integer. Thus, the probability p that C takes the value 1 is 0.5, and $H(C) = 1$ bit. In this case $Q_{p=0.5} = \frac{1}{H(W)} \geq \frac{1}{\log_2(d_c+1)}$, with the equality occurring if and only if W is uniformly distributed in the range $\{0, \ldots, d_c\}$. Effectively, $Q_{p=0.5}$ denotes the dimensionality reduction L/N possible in this non-compressible case because the binary input C is trans-

formed into a nonbinary variable W.

In general, the variable W can be efficiently compressed (e.g., by an Entropy Coding scheme such as Huffman Coding) without any encoding loss if all the probabilities $p_W(W)$ are negative powers of two [107]. In practice, an encoding loss is typically incurred since this condition does not hold. Let encoding of k^{th} symbol ($k = 1, \ldots, d_c + 1$) with probability p_W^k result in n_k bits, and let $p_{\hat{W}}^k = 2^{-n_k}$ denote the post-encoding pseudo-probability of k^{th} symbol of W.

With this, the average number of bits R_W per symbol of W can be written as follows:

$$\begin{aligned} R_W &= \sum_{k=1}^{d_c+1} p_W^k n_k \\ &= -\sum_{k=1}^{d_c+1} p_W^k \log_2\left(p_{\hat{W}}^k\right) \\ &= -\sum_{k=1}^{d_c+1} p_W^k \log_2\left(p_W^k\right) - \sum_{k=1}^{d_c+1} p_W^k \log_2\left(\frac{p_{\hat{W}}^k}{p_W^k}\right) \\ &= H(W) + D(p_W \parallel p_{\hat{W}}). \end{aligned} \tag{6.4}$$

Here, $D(p_W \parallel p_{\hat{W}})$ denotes the Kullback-Leibler Divergence between the distribution of the output variable W and the encoded variable $\hat{W}$.

The average number of bits required to encode $L \times 1$ output vector $\mathbf{w}$ is given as $L \times R_W$, which is lower bounded by $N \times H(C)$. Therefore, the efficiency $0 \leq \eta \leq 1$ of data compression is given as the ratio $\eta = \dfrac{N \times H(C)}{L \times R_W}$. An upper bound on this ratio is stated in the following Lemma.

Lemma 6.1.1 *Efficiency of the data compression η is upper bounded by* $\left(1 + Q_p \times \dfrac{D(p_W \parallel p_{\hat{W}})}{H(C)}\right)^{-1}$.

$$\begin{aligned}
\eta &= \frac{N \times H(C)}{L \times R_W} \\
&= \frac{N \times H(C)}{L \times \left(H(W) + D(p_W \parallel p_{\hat{W}})\right)} && (6.5) \\
&= \frac{1}{\frac{L}{N} \times \left(\frac{H(W)}{H(C)} + \frac{D(p_W \parallel p_{\hat{W}})}{H(C)}\right)} \\
&\leq \frac{1}{Q_p \times \left(\frac{H(W)}{H(C)} + \frac{D(p_W \parallel p_{\hat{W}})}{H(C)}\right)} && (6.6) \\
&= \frac{1}{1 + Q_p \times \frac{D(p_W \parallel p_{\hat{W}})}{H(C)}}. && (6.7)
\end{aligned}$$

Here, (6.5) is obtained from (6.4) and the inequality in (6.6) follows from (6.3).

We next consider the following two schemes for compression of the binary source word.

- Scheme 1 (Serial to Parallel Approach): In this scheme, a total of $\log_2(d_c + 1)$ bits are taken together (in a block-by-block manner, where the variable d_c is chosen to be a power of two minus 1) to create a variable $W^{(1)}$, which takes one value out of a total of $d_c + 1$ different symbols. We require a total of $L^{(1)}$ such blocks for N bits, where $L^{(1)} = N/\log_2(d_c + 1)$. Thus, the ratio $L^{(1)}/N$ equals $1/\log_2(d_c + 1)$. Assuming that N bits are independent, $H(W^{(1)}) = \log_2(d_c + 1)H(C)$, and we obtain

$$Q_p^1 = H(C)/H(W^{(1)}) = 1/\log_2(d_c + 1) = L^{(1)}/N. \tag{6.8}$$

- Scheme 2 (with LDPC Matrix): In this scheme, we linearly transform vector $\mathbf{c}$ to $\mathbf{w}$ using LDPC matrix $\mathbf{H}$, i.e., $\mathbf{w} = \mathbf{Hc}$. Denoting the resultant RV as $W^{(2)}$, when number of ones in rows of $\mathbf{H}$ is d_c, $W^{(2)}$ has the same number $(d_c + 1)$ of different symbols as in Scheme 1. However, instead of $\log_2(d_c + 1)$, a greater number $d_c + 1$ of bits are summed by each row of $\mathbf{H}$ to generate $W^{(2)}$. The ratio $L^{(2)}/N$ is lower bounded by $Q_p^2 = H(C)/H(W^{(2)})$. Here, the denominator $H(W^{(2)})$ is the Entropy of Binomial Distributed $W^{(2)}$,which

can be approximated as

$$H(W^{(2)}) = 0.5 \times (\log_2(d_c) + \log_2(p(1-p)) + \log_2(2\pi e)) + \mathcal{O}(1/d_c)$$

Lemma 6.1.2 *In the asymptotic case of $d_c \to \infty$, ratio Q_p^1/Q_p^2 is greater than 1 for $p \leq 0.11$.*

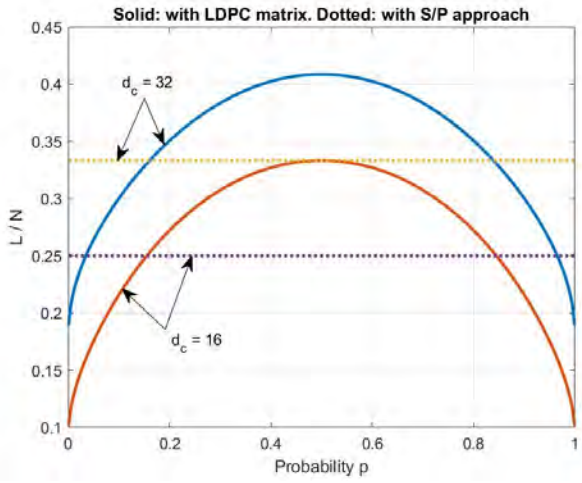
Given the expression of Q_p^1 in (6.8) the ratio Q_p^2/Q_p^1 can be written as follows:

$$\begin{aligned} Q_p^1/Q_p^2 &= \frac{1}{\log_2(d_c+1)} \times \frac{H(W^{(2)})}{H(C)} \\ &= \frac{0.5}{H(C)} \times \underbrace{\frac{(\log_2(d_c) + \log_2(p(1-p)) + \log_2(2\pi e)) + \mathcal{O}(1/d_c)}{\log_2(d_c+1)}}_{\text{This term approaches 1 as } d_c \text{ increases.}} \end{aligned}$$

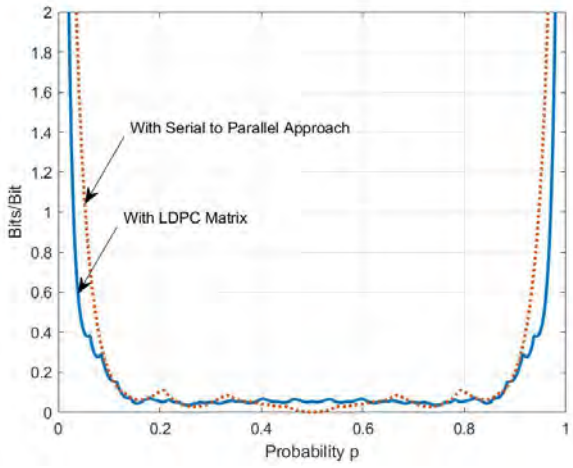
Thus, in the limit $d_c \to \infty$, $Q_p^1/Q_p^2 \to \frac{0.5}{H(C)}$. This ratio equals 1 when $H(C) = 0.5$, i.e., when $p \approx 0.11$. For $p < 0.11$, $H(C) < 0.5$ and the ratio Q_p^1/Q_p^2 becomes greater than unity (vice versa when $p > 0.11$).

A numerical demonstration of Lemma 6.1.2 is provided in Fig. 6.1a, which shows that $Q_p^1 > Q_p^2$ for $p \leq 0.11$. Result of a numerical evaluation of the ratio $\frac{D(p_W \parallel p_{\hat{W}})}{H(C)}$ for both Scheme 1 and Scheme 2 is provided in Fig. 6.1b. It is seen that both the terms, Q_p and $\frac{D(p_W(W) \parallel p_{\hat{W}}(\hat{W}))}{H(C)}$ in the denominator of (6.7) are greater for Scheme 1 compared to Scheme 2 when $p \leq 0.11$. Accordingly, the data compression efficiency of the LDPC scheme is greater than that of the conventional Serial to Parallel (S/P) approach for low values of probabilities. This is shown in Fig. 6.2.

The analysis presented above shows the efficacy of LDPC matrices in achieving a highly efficient data compression for low values of p and sparsely populated source vectors. This also shows that the LDPC matrices are well-suited to the Compressive Sensing domain where a sparse source is simultaneously measured/sampled and compressed.



(a) Ratios Q_p^1 in dotted and Q_p^2 in solid.



(b) $\dfrac{D(p_W \parallel p_{\hat{W}})}{H(C)}$.

Figure 6.1: Evaluation of the individual terms of (6.7).

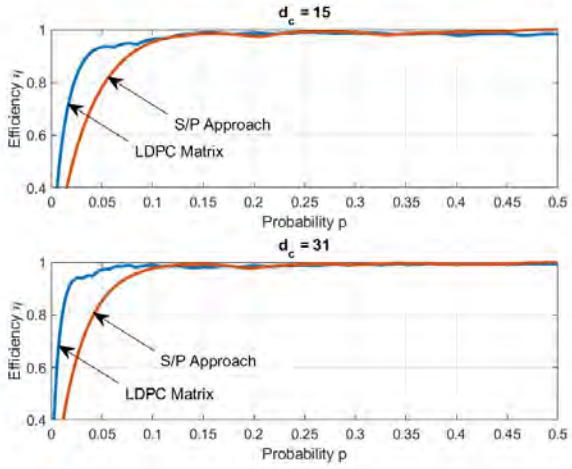


Figure 6.2: Efficiency η.

6.1.2 Source Decoding

In noiseless data compression, the source encoder has the same information available to the decoder. Thus, not only it can check whether the original source word will be correctly decompressed but it can also duplicate the decoder iterations.

The fixed-to-fixed (N to L) compression algorithm can be adapted to zero-error variable-to-fixed or fixed-to-variable operation by tuning the input/output length in the compression to guarantee decoding success, for example, by successively generating the parity-check equations until successful decoding is achieved. Al-

ternatively, the encoder can try several parity-check matrices (with equal or different rates) from a library known at the decoder until success is achieved. A header identifying the actual matrix sent is part of the compressed sequence.

Therefore, source decoding methods for noiseless data compression is complementary to encoding scheme. For noiseless fixed-to-fixed (N to L) compression case, deterministic algorithm as described in Section 5.5.2 in Chapter 5 is used for source decoding. The deterministic algorithm has proven performance for information sequences with high sparsity compared to other known similar algorithms.

Source decoding and denoising: For the non-binary, non-discrete, cases, the analog-valued vector $\mathbf{c}$ can be represented as $\mathbf{x} = \mathbf{c} + \mathbf{u}$, where $\mathbf{u} = \mathbf{x} - \mathbf{c}$ is the error relative to the binary (or, in general, discrete-valued) model. Thus, our interest is in compressed representation of binary $\mathbf{c}_b$ when it is perturbed by modeling error or noise $\mathbf{u}$. The goal, therefore, is joint compression as well as denoising.

In such cases, the developed message passing algorithms with non-zero sigma value is used at source decoder. The analysis and simulation results presented in Section 5.2.2 of Chapter 5, shows that the developed BP algorithm within LDIC framework is suitable for compression and performs denoising.

Summary: The mathematical analysis of the proposed LDIC codes have demonstrated that the sparse binary LDPC matrix is a suitable candidate for the task of lossless data compression than conventional serial to parallel approach. Also, the deterministic algorithm and other message passing algorithms developed in this work are suitable for source decoding and denoising applications.

CHAPTER 7

Summary and Conclusion

7.1 Work Summary and Conclusion

In this work, the operation and efficacy of the proposed novel hardware architecture with newly developed sparsity detection algorithms for wideband spectrum sensing and communication applications has been demonstrated through analysis and simulation. The proposed hardware architecture implements a sparse binary regular LDPC matrix (without cycles of 4 and 6) as analog front-stage of compressive sampling (CS) receiver for sparse multiband signals. The hardware architecture has w_c (=number of rows) number of parallel branches which down-convert the received signal using a sum of a small number ($= w_r$) of complex-valued sinusoids with random assignment of frequencies in the sensed spectral width. The down-converted output of each branch is low-pass filtered and then digitised. The sampled output of the digitiser is passed through energy estimator block which provides observations as function of input with Bernoulli distribution. These observations along with the measurement matrix are used by belief propagation based sparsity detection algorithms to determine the spectral locations of active sub-bands in the sparsely populated wide multiband input signal. The active sub-band location is then used to form a modified measurement matrix thereby, ensuring the linear independence of all columns in modified matrix. The updated measurement matrix with sampled output of all branches is then used for signal reconstruction.

The proposed architecture is different from the already known MWC architecture in two distinct ways. The first being the use of a linear combination of w_r

complex sinusoids as mixing sequence in place of rectangular random periodic mixing sequence. The use of complex sinusoid ensures uniform amplitude of all spectral components of mixing sequence thereby, removing sync type amplitude modulation present in the case of rectangular random periodic sequence. The proposed change ensures uniform treatment of all input sub-bands which was not the case for MWC architecture. The other advantage of using w_r complex sinusoid (regular LDPC matrix) is reduced noise foldover due to the presence of only w_r spectral component in the mixing sequence. The quantum of benefit has been derived analytically and demonstrated by performing Bit Error Rate analysis of PSK system with the proposed hardware architecture and MWC architecture. The proposed system shows better BER performance compared to MWC. The second difference is use of energy estimator block which enables discretized compressive sensing model and use of message passing based robust sparsity detection algorithms. For given size of measurement matrix or hardware architecture, the performance of the developed sparsity detection algorithms is found to be closer to the lower bounds derived using information theoretic approach.

The two new algorithms developed for sparsity detection are message passing algorithms with Independent Probability Estimates (IPE) and Joint Probability Estimates (JPE). The third algorithm is deterministic method of sparsity detection. The probability (belief) calculation in case of IPE algorithm is based on the assumption that the occupancy of each sub-band is independent of the other sub-band. In case of JPE algorithm the probability is calculated based on observations and joint probability of all sub-bands. For these algorithms, a Monte-Carlo simulation was run for different sizes of measurement matrix and all sparsity levels for performance verification. Additionally, a semi-analytical convergence analysis using Density Evolution and EXIT chart method was performed for different sizes of measurement matrices and discrete sparsity levels. The outcome of simulation was compared against derived information theoretic performance bounds of different sizes of measurement matrices and sparsity levels. The comparison shows that the performance of JPE is closer to information theoretic bounds and offers a minimum one fourth hardware savings compared to Nyquist processing for all

sparsity levels. The performance of JPE and IPE is found to be better than the algorithms motivated by Sudocode and Gallager based LDPC decoding techniques. The IPE algorithm with non-binary (3-level) sparse signal vector input suggests that the algorithms of sparsity detection can also be generalised for $M-$ary sparse signal vectors.

A LDIC framework is proposed and analysed theoretically for joint compression and denoising application. The simulated results prove that the developed framework can offer better compression efficiency. Wherever, it is possible to detect the binary pattern hidden inside the real measurements, the framework can achieve compression as well as denoising.

As an application of known theoretical CS framework, a simulation based performance verification of CS receiver for narrowband jamming detection and filtering application has been performed. The proposed framework demonstrates low processing load advantage over conventional Nyquist based receivers. The receiver performed blind interference detection and filtering and is suitable for spectrum sensing in cognitive communication applications. Another unique application of CS for realisation of Scalable Video Codec (SVC) is proposed. In this SVC framework, at encoder, sparse wavelet coefficients are compressed using CS matrix and they are retrieved using AMP algorithm at decoder.

The analytical framework of this thesis encourages the development of a hardware prototype of the proposed analog front-end for wideband signal, which are sparse in frequency domain. The characterisation and performance optimisation of the developed algorithms will establish the proposed CS architecture as an efficient sub-Nyquist sampling scheme. The CS-based receivers are best suited for blind spectrum sensing and wideband communication receivers for cognitive applications.

7.2 Future Work

The CS framework presented in this research work is very promising and calls for extended research. The two sparsity detection algorithms developed assume Bernoulli distribution marked by the presence or absence of signal within a sub-band of wide multiband input. This is valid only when all sub-bands have similar flux density and occupied bandwidth. However, in many practical applications the waveform of all sub-bands may be different resulting in discrete $M-ary$ inputs to measurement matrices. Therefore, a generalisation of the message passing algorithms for discrete $M-ary$ inputs will enhance the applicability of the proposed framework. Signal reconstruction methods of input waveforms spread across two or more bands need to be developed. Also, a suitable method of normalisation of energy detector output of the proposed CS receiver will help improve the efficacy of the system. In the present work, the convergence criteria for developed MP algorithms are based on fixed number of iterations which could be optimised and made dynamic for faster processing. The CS architecture is a realisation of sparse binary matrix without cycles of 4 and 6 at given rates. Development of deterministic construction methods of sparse binary matrix for all rates will further aid the realisation of the proposed CS architecture.

References

[1] A. J. Jerri. The shannon sampling theorem- its various extensions and applications: A tutorial review. *Proceedings of the IEEE*, 65(11):1565–1596, Nov 1977.

[2] P. L. Butzer. A survey of the whittaker-shannon sampling theorem and some of its extensions. *J Math. Res. Expo.,*, 1983(1):185–212, Jan 1983.

[3] H. J. Landau. Sampling, data transmission, and the nyquist rate. *Proceedings of the IEEE*, 55(10):1701–1706, Oct 1967.

[4] Athanasios Papoulis. Generalized Sampling Expansion. *IEEE transactions on circuits and systems*, 24(11):652–654, 1977.

[5] Rodney G Vaughan, Neil L Scott, and D Rod White. The Theory of Bandpass Sampling. *IEEE Transactions on signal processing*, 39(9):1973–1984, 1991.

[6] J. Brown. Multi-channel sampling of low-pass signals. *IEEE Transactions on Circuits and Systems*, 28(2):101–106, February 1981.

[7] Yuan-Pei Lin and P. P. Vaidyanathan. Periodically nonuniform sampling of bandpass signals. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 45(3):340–351, Mar 1998.

[8] Yuan-Pei Lin and P. P. Vaidyanathan. Periodically nonuniform sampling of bandpass signals. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 45(3):340–351, Mar 1998.

[9] William C Black and David A Hodges. Time Interleaved Converter Arrays. *IEEE Journal of Solid-state circuits*, 15(6):1022–1029, 1980.

[10] C. R. Anderson, S. Venkatesh, J. E. Ibrahim, R. M. Buehrer, and J. H. Reed. Analysis and implementation of a time-interleaved adc array for a software-defined uwb receiver. *IEEE Transactions on Vehicular Technology*, 58(8):4046–4063, Oct 2009.

[11] B. Razavi. Problem of timing mismatch in interleaved adcs. In *Proceedings of the IEEE 2012 Custom Integrated Circuits Conference*, pages 1–8, Sept 2012.

[12] Yonina C. Eldar. Sampling theory: Beyond bandlimited systems. *Book: Cambridge University Press*, 2015.

[13] Shree Krishna Sharma, Eva Lagunas, Symeon Chatzinotas, and Björn Ottersten. Application of compressive sensing in cognitive radio communications: A survey. *IEEE Communication Surveys & Tutorials*, 2016.

[14] Fatima Salahdine, Naima Kaabouch, and Hassan El Ghazi. A survey on compressive sensing techniques for cognitive radio networks. *Physical Communication*, 20:61–73, 2016.

[15] Moshe Mishali, Yonina C Eldar, and Joel A Tropp. Efficient sampling of sparse wideband analog signals. In *Electrical and Electronics Engineers in Israel, 2008. IEEEI 2008. IEEE 25th Convention of*, pages 290–294. IEEE, 2008.

[16] Sami Kirolos, Jason Laska, Michael Wakin, Marco Duarte, Dror Baron, Tamer Ragheb, Yehia Massoud, and Richard Baraniuk. Analog-to-information conversion via random demodulation. In *Design, Applications, Integration and Software, 2006 IEEE Dallas/CAS Workshop on*, pages 71–74. IEEE, 2006.

[17] Moshe Mishali and Yonina C Eldar. From Theory to Practice: Sub-Nyquist Sampling of Sparse Wideband Analog Signals. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):375–391, 2010.

[18] Yongqing Qian and Weizhen Chen. Wideband spectrum sensing in cognitive radio with structured bayesian compressed sensing by variational

bayesian approach. In *Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), International Congress on*, pages 1128–1132. IEEE, 2016.

[19] Davood Mardani Najafabadi, Ali A Tadaion, and Masoud Reza Aghabozorgi Sahaf. Wideband spectrum sensing using compressive sampling based energy reconstruction. In *Telecommunications and Signal Processing (TSP), 2012 35th International Conference on*, pages 667–670. IEEE, 2012.

[20] Pankaz Das and Sudharman K Jayaweera. Robust wideband spectrum sensing with compressive sampling in cognitive radios. In *Vehicular Technology Conference (VTC Fall), 2015 IEEE 82nd*, pages 1–5. IEEE, 2015.

[21] Mohammad-Ali Damavandi and Said Nader-Esfahani. Compressive wideband spectrum sensing in cognitive radio systems based on cyclostationary feature detection. In *Next Generation Mobile Applications, Services and Technologies, 2015 9th International Conference on*, pages 282–287. IEEE, 2015.

[22] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, April 2006.

[23] J. Romberg Emmanuel J. Candes and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Comm.Pure Appl. Math.*, 59(8):1207–1223, Aug 2006.

[24] Emmanuel J. Candes. The restricted isometry property and its implications for compressed sensing. *Academie des sciences*, feb 2008.

[25] E. J. Candes and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, Dec 2005.

[26] S. G. Mallat and Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, Dec 1993.

[27] Joel A. Tropp and Anna C. Gilbert. Signal recovery from partial information via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 2005.

[28] Yonina C Eldar and Gitta Kutyniok. *Compressed Sensing: Theory and Applications*. Cambridge University Press, 2012.

[29] David L Donoho, Arian Maleki, and Andrea Montanari. Message Passing Algorithms For Compressed Sensing. *Proceedings of the National Academy of Sciences*, 106(45):18914–18919, 2009.

[30] E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, Feb 2006.

[31] Jianwei Ma, Gerlind Plonka, and M Yousuff Hussaini. Compressive video sampling with approximate message passing decoding. *IEEE Transactions on circuits and systems for video technology*, 22(9):1354–1364, 2012.

[32] S. N. Karishma, B. K. N. Srinivasarao, Indrajit Chakrabarti, and Chandra Prakash. Hardware/software co-simulation of entropy coder for compressive sensing based scalable video codec system. *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pages 727–731, 2017.

[33] Indrajit Chakrabarti Chandra Prakash S. N. Karishma, B. K. N. Srinivasarao. Hardware/software co-simulation of entropy coder for compressive sensing based scalable video codec system. *IEEE Wireless Communications, Signal Processing and Networking (WISPNET)*, 2017.

[34] Indrajit Chakrabarti Chandra Prakash S. N. Karishma, B. K. N. Srinivasarao. Scalable video coding based method for compressing satellite and spaceborne data. *Space Development Network Conference Asia (SDN), IISc Bangalore*, 2016.

[35] D. Baron, S. Sarvotham, and R. G. Baraniuk. Bayesian Compressive Sensing Via Belief Propagation. *IEEE Transactions on Signal Processing*, 58(1):269–280, Jan 2010.

[36] A. G. Dimakis, R. Smarandache, and P. O. Vontobel. LDPC Codes for Compressed Sensing. *IEEE Transactions on Information Theory*, 58(5):3093–3114, May 2012.

[37] Elaine Crespo Marques, Nilson Maciel, Lírida Naviner, Hao Cai, and Jun Yang. A review of sparse recovery algorithms. *IEEE Access*, 7:1300–1322, 2018.

[38] Moshe Mishali and Yonina C Eldar. Blind Multiband Signal Reconstruction: Compressed Sensing For Analog Signals. *IEEE Transactions on Signal Processing*, 57(3):993–1009, 2009.

[39] Thu LN Nguyen and Yoan Shin. Deterministic sensing matrices in compressive sensing: a survey. *The Scientific World Journal*, 2013, 2013.

[40] Yash Vasavada and Chandra Prakash. An Iterative Message Passing Approach for Compressive Spectrum Sensing . *IEEE International Conference on Advances in Computing, Communications and Informatics*, 2017.

[41] Yash Vasavada and Chandra Prakash. Sub-nyquist spectrum sensing of sparse wideband signals using low density measurement matrix. *IEEE Transactions on Signal Processing: Under Review*, 2018.

[42] Yash Vasavada and Chandra Prakash. Low desnity integer constrained codes. *IEEE Transactions on Communication : Under Submission*, 2018.

[43] Chandra Prakash and Vijay Kumar Chakka. Sub-band filtering in compressive domain. *IEEE 3rd International Conference on Advances in Computing, Communications and Informatics*, September 2014.

[44] Chandra Prakash, Deep Bhimani, and Vijay Kumar Chakka. Interference detection and filtering in satellite transponder. *IEEE International Conference on Communication and Signal Processing*, April 2014.

[45] Yash Vasavada and Chandra Prakash. Narrow Band Jamming Detection and Filtering in Satellite Transponder. *5th IEEE International Conference on Signal Processing and Integrated Networks-Under Review*, 2018.

[46] Chandra Prakash, Vijay Kumar, S. N. Satashia, and K. S. Parikh. Digital receiver based ka band beacon receiver for improved beacon power estimation. *IEEE International Conference on Communication and Signal Processing*, April 3-5 2013.

[47] Emmanuel J Candès and Michael B Wakin. An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2):21–30, 2008.

[48] Emmanuel J Candes and Terence Tao. Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215, 2005.

[49] Andjela Draganic, Irena Orovic, and Srdjan Stankovic. On some common compressive sensing recovery algorithms and applications-review paper. *arXiv preprint arXiv:1705.05216*, 2017.

[50] R. Venkataramani and Y. Bresler. Optimal sub-nyquist nonuniform sampling and reconstruction for multiband signals. *IEEE Transactions on Signal Processing*, 49(10):2301–2313, Oct 2001.

[51] M. E. Domínguez-Jitnénez and N. González-Prelcic. Analysis and design of multirate synchronous sampling schemes for sparse multiband signals. In *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pages 1184–1188, Aug 2012.

[52] T. Moon, N. Tzou, Xian Wang, Hyun Choi, and A. Chatterjee. Low-cost high-speed pseudo-random bit sequence characterization using nonuniform periodic sampling in the presence of noise. In *2012 IEEE 30th VLSI Test Symposium (VTS)*, pages 146–151, April 2012.

[53] Samba Traore, Babar Aziz, Daniel Le Guennec, et al. Adaptive multi-coset sampler. In *4th Workshop of COST Action IC0902*, 2013.

[54] J. N. Laska, S. Kirolos, M. F. Duarte, T. S. Ragheb, R. G. Baraniuk, and Y. Massoud. Theory and implementation of an analog-to-information converter using random demodulation. In *2007 IEEE International Symposium on Circuits and Systems*, pages 1959–1962, May 2007.

[55] Benjamin Scott Boggess. Compressive sensing using random demodulation. *Masters Theses*, page 21, 2009.

[56] Joel A Tropp, Jason N Laska, Marco F Duarte, Justin K Romberg, and Richard G Baraniuk. Beyond Nyquist: Efficient Sampling of Sparse Bandlimited Signals. *IEEE Transactions on Information Theory*, 56(1):520–544, 2010.

[57] M. Mishali, Y.C.Eldar, and Asaf J Elron. Xampling: Signal acquisition and processing in union of subspaces. *IEEE Transactions on Signal processing*, Oct.l 2011.

[58] Shree Krishna Sharma, Symeon Chatzinotas, and Bjorn Ottersten. Cognitive radio techniques for satellite communication systems. In *Vehicular Technology Conference (VTC Fall), 2013 IEEE 78th*, pages 1–5. IEEE, 2013.

[59] Simon Haykin et al. Cognitive radio: brain-empowered wireless communications. *IEEE journal on selected areas in communications*, 23(2):201–220, 2005.

[60] Feilong Li, Guangxia Li, Zhiqiang Li, Yulin Wang, and Gengxin Zhang. A novel approach to wideband spectrum compressive sensing based on dst for frequency availability in leo mobile satellite systems. *Mathematical Problems in Engineering*, 2016, 2016.

[61] Shree Krishna Sharma, Symeon Chatzinotas, and Björn Ottersten. Satellite cognitive communications: Interference modeling and techniques selection. In *Advanced Satellite Multimedia Systems Conference (ASMS) and 12th Signal Processing for Space Communications Workshop (SPSC), 2012 6th*, pages 111–118. IEEE, 2012.

[62] Shilian Zheng and Xiaoniu Yang. Wideband spectrum sensing in modulated wideband converter based cognitive radio system. In *Communications and Information Technologies (ISCIT), 2011 11th International Symposium on*, pages 114–119. IEEE, 2011.

[63] Peihan Qi, Zan Li, Hongbin Li, and Tianyi Xiong. Blind sub-nyquist spectrum sensing with modulated wideband converter. *IEEE Transactions on Vehicular Technology*, 67(5):4278–4288, 2018.

[64] Mort Naraghi-Pour and Takeshi Ikuma. Autocorrelation based spectrum sensing for cognitive radios. *IEEE transactions on Vehicular Technology*, 59(2):718–733, 2010.

[65] Geert Leus Dyonisius Dony Arianandа. Compressive wideband power spectrum estimation. *IEEE Transactions on Signal processing*, 60, 2012.

[66] Hector Reyes, Sriram Subramaniam, Naima Kaabouch, and Wen Chen Hu. A spectrum sensing technique based on autocorrelation and euclidean distance and its comparison with energy detection for cognitive radio networks. *Comput. Electr. Eng.*, 52(C):319–327, May 2016.

[67] S. Subramaniam, H. Reyes, and N. Kaabouch. Spectrum occupancy measurement: An autocorrelation based scanning technique using usrp. In *2015 IEEE 16th Annual Wireless and Microwave Technology Conference (WAMICON)*, pages 1–5, April 2015.

[68] Y. Kondareddy and P. Agrawal. Collaborative spectrum sensing in cognitive radio networks. volume 59, pages 1–6, 2011.

[69] Z. Tian and G. B. Giannakis. A wavelet approach to wideband spectrum sensing for cognitive radios. In *2006 1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, pages 1–5, June 2006.

[70] Zhang Zhang, Qingqing Yang, Lingkai Wang, and Xiaofang Zhou. A novel hybrid matched filter structure for ieee 802.22 standard. In *Circuits and Sys-*

tems (APCCAS), 2010 IEEE Asia Pacific Conference on, pages 652–655. IEEE, 2010.

[71] F Weidling, D Datla, V Petty, P Krishnan, and GJ Minden. A framework for rf spectrum measurements and analysis. In *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, pages 573–576. IEEE, 2005.

[72] Moshe Mishali and Yonina C Eldar. Wideband spectrum sensing at sub-nyquist rates [applications corner]. *IEEE Signal Processing Magazine*, 28(4):102–135, 2011.

[73] Moshe Mishali and Yonina C Eldar. Reduce and boost: Recovering arbitrary sets of jointly sparse vectors. *IEEE Transactions on Signal Processing*, 56(10):4692–4702, 2008.

[74] C. H. Lin, S. H. Tsai, and G. C. H. Chuang. A novel sub-nyquist sampling of sparse wideband signals. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4628–4632, May 2013.

[75] Lu Gan and Huali Wang. Deterministic binary sequences for modulated wideband converter. In *Proceeding of 10th International Conference on Sampling Theory and Applications*, pages 264–267, May 2013.

[76] Steven Barraclough. Satellite television in iran: Prohibition,imitation and reform. *Middle East Studies*, 37, 2005.

[77] Steven Barraclough. Eutelsat condems jamming of broadcasts from iran and renews appeals for decisive action to international regulators. *Eutelsat Communications*, October 2012.

[78] Mark A. Davenport, Petros T Boufounos, Michael B. Wakin, and Richard G. Baraniuk. *Rec. ITU-R SF.1004*.

[79] Z. Ben-Haim, Y. C. Eldar, and M. Elad. Coherence-based performance guarantees for estimating a sparse vector under random noise. *IEEE Transactions on Signal Processing*, 58(10):5030–5043, Oct 2010.

[80] Ery Arias-Castro and Yonina C Eldar. Noise folding in compressed sensing. *IEEE Signal Processing Letters*, 18(8):478–481, 2011.

[81] Ery Arias-Castro and Yonina C. Eldar. Noise folding in compressed sensing. *CoRR*, abs/1104.3833, 2011.

[82] Ery Arias-Castro and Yonina C Eldar. Noise folding in compressed sensing. *IEEE Signal Processing Letters*, 18(8):478–481, 2011.

[83] Arash Amini and Farokh Marvasti. Deterministic construction of binary, bipolar, and ternary compressed sensing matrices. *IEEE Transactions on Information Theory*, 57(4):2360–2370, 2011.

[84] Gesen Zhang, Shuhong Jiao, Xiaoli Xu, and Lan Wang. Compressed sensing and reconstruction with bernoulli matrices. In *Information and Automation (ICIA), 2010 IEEE International Conference on*, pages 455–460. IEEE, 2010.

[85] Shu-Tao Xia, Xin-Ji Liu, Yong Jiang, and Hai-Tao Zheng. Deterministic constructions of binary measurement matrices from finite geometry. *IEEE Transactions on Signal Processing*, 63(4):1017–1029, 2015.

[86] Alexandros G Dimakis, Roxana Smarandache, and Pascal O Vontobel. Ldpc codes for compressed sensing. *IEEE Transactions on Information Theory*, 58(5):3093–3114, 2012.

[87] Jason N Laska, Sami Kirolos, Marco F Duarte, Tamer S Ragheb, Richard G Baraniuk, and Yehia Massoud. Theory and Implementation of an Analog-to-Information Converter Using Random Demodulation. In *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, pages 1959–1962. IEEE, 2007.

[88] J. A. Tropp, M. B. Wakin, M. F. Duarte, D. Baron, and R. G. Baraniuk. Random filters for compressive sampling and reconstruction. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 3, pages III–III, May 2006.

[89] Justin Romberg. Compressive sensing by random convolution. *SIAM Journal on Imaging Sciences*, 2(4):1098–1128, 2009.

[90] W. Lu, K. Kpalma, and J. Ronsin. Sparse Binary Matrices of LDPC Codes for Compressed Sensing. In *2012 Data Compression Conference*, pages 405–405, April 2012.

[91] Shriram Sarvotham, Dror Baron, and Richard G Baraniuk. Sudocodes fast measurement and reconstruction of sparse signals. In *Information Theory, 2006 IEEE International Symposium on*, pages 2804–2808. IEEE, 2006.

[92] E Prapuchanay, D Sakulhirirak, H Zenkner, and S Chalermwisutkul. Design and implementation of a low cost bjts comb generator. In *Intelligent Signal Processing and Communications Systems (ISPACS), 2011 International Symposium on*, pages 1–4. IEEE, 2011.

[93] Kenneth E Kolodziej. Simple comb generator design for swap constrained applications. Technical report, MIT Lincoln Laboratory Lexington United States, 2016.

[94] Puja Srivastava, Punam P Kumar, and Ch VN Rao. Design and development of multi-frequency comb generator for space borne microwave active sensors. In *Recent Advances in Microwave Theory and Applications, 2008. MICROWAVE 2008. International Conference on*, pages 445–447. IEEE, 2008.

[95] Feng Lin and Mina Rais-Zadeh. Continuously tunable 0.55–1.9-ghz bandpass filter with a constant bandwidth using switchable varactor-tuned resonators. *IEEE Transactions on Microwave Theory and Techniques*, 65(3):792–803, 2017.

[96] Alexander Jaschke, Mengistu Tessema, Mario Schühler, and Rainer Wansch. Digitally tunable bandpass filter for cognitive radio applications. In *Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2012 IEEE 17th International Workshop on*, pages 338–342. IEEE, 2012.

[97] Hyun-Keun Kim, Jong-Hyun Lee, and Sang Won Yun. Design of digitally tunable bandpass filter for spectrum sensing application in the tvws. In *Microwave Conference (APMC), 2014 Asia-Pacific*, pages 938–940. IEEE, 2014.

[98] R. T. Yazicigil, T. Haque, M. Kumar, J. Yuan, J. Wright, and P. R. Kinget. How to make analog-to-information converters work in dynamic spectrum environments with changing sparsity conditions. *IEEE Transactions on Circuits and Systems I: Regular Papers*, PP(99):1–10, 2017.

[99] Jun Fan, Yang Xiao, Kiseon Kim. Design LDPC Codes without Cycles of Length 4 and 6. *Research Letters in Communications*, 2008(354137), 2008.

[100] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[101] R. Gallager. Low Density Parity Check Codes. *IRE Transactions on Information Theory*, 8(1):21–28, January 1962.

[102] Bernhard M. J. Leiner. LDPC Codes - A Brief Tutorial. http://www.bernh.net/media/download/papers/ldpc.pdf, April 2005.

[103] S. Sarvotham, D. Baron, and R. G. Baraniuk. Sudocodes Fast Measurement and Reconstruction of Sparse Signals. In *2006 IEEE International Symposium on Information Theory*, pages 2804–2808, July 2006.

[104] Fan Zhang and Henry D Pfister. On the Iterative Decoding of High Rate LDPC Codes with Applications in Compressed Sensing. *arXiv preprint arXiv:0903.2232*, 2009.

[105] Thomas J Richardson, Mohammad Amin Shokrollahi, and Rüdiger L Urbanke. Design of capacity-approaching irregular low-density parity-check codes. *IEEE transactions on information theory*, 47(2):619–637, 2001.

[106] B. L. Sturm and M. G. Christensen. Comparison of orthogonal matching pursuit implementations. In *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pages 220–224, Aug 2012.

[107] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[108] Siyuan Xiang and Lin Cai. Scalable video coding with compressive sensing for wireless videocast. In *Communications (ICC), 2011 IEEE International Conference on*, pages 1–5. IEEE, 2011.

[109] Iain EG Richardson and Yafan Zhao. Adaptive algorithms for variable-complexity video coding. In *Image Processing, 2001. Proceedings. 2001 International Conference on*, volume 1, pages 457–460. IEEE, 2001.

Appendix A

CS Based Scalable Video Codec (SVC)

In modern age of communication, multimedia communication occupies highest traffic bandwidth and due to the variable size of user gadgets and variable data carrying capacity of communication networks, scalable video codec is in high demand. Scalable video codec allows a video to be encoded in scalable mode which could be decoded using a layered approach. Researchers have already proposed a CS based SVC in literature [108]. CS based scalable video encoder that exploits the sparsity of the video frames in the wavelet basis is used for source coding at the transmitter end [33].

Wavelet based video compression schemes inherently provide frame rate, resolution and quality scalabilities in spatial and temporal domains. Wavelet based scalable video codec exhibit excellent coding efficiency without sacrificing the compression ratio in comparison to non-scalable MPEG-4 coding technology. The block diagram for proposed CS based Scalable Video Codec (SVC) is shown in Fig-A.1.

In the proposed CS based SVC approach, the input video frames are decomposed in several sub-bands using 3-D DWT and Compressed sensing (CS) is applied to all the sub-bands except lowest frequency sub-bands [34]. Three dimensional Discrete wavelet transform (3D-DWT) inherently offers resolution, frame-rate and quality scalability. Wavelet based compression algorithms provide excellent coding efficiency without sacrificing Signal to Noise Ratio (SNR). The signal processing steps for the proposed scalable video encoder is shown in Figure-A.2.

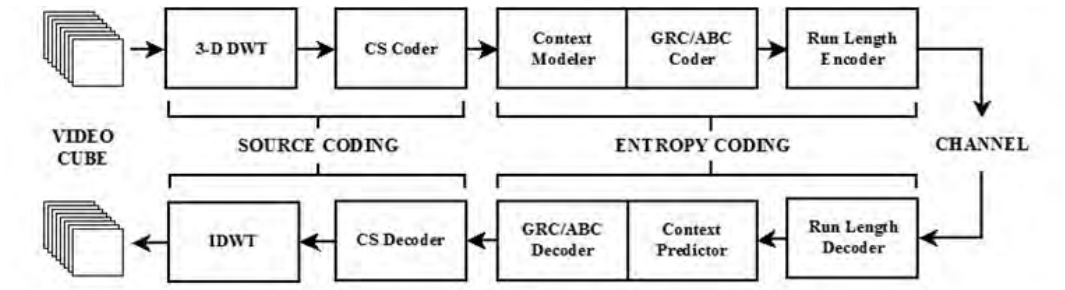


Figure A.1: Block Diagram of CS based SVC

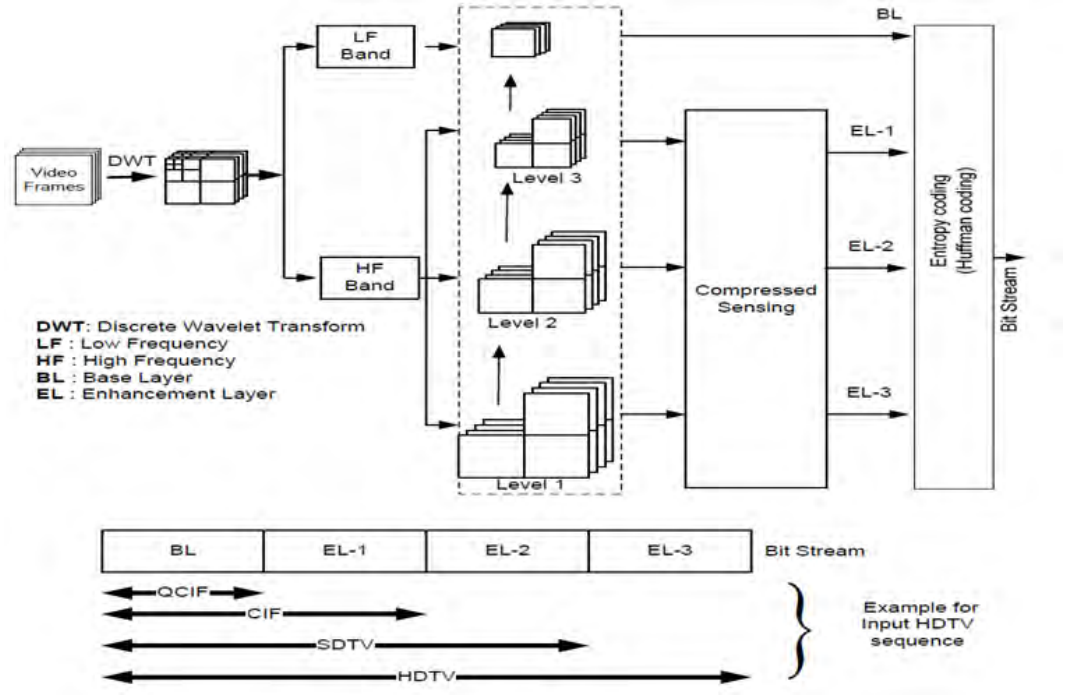


Figure A.2: Processing steps of Scalable Video Encoder

The process of 3D-DWT implementation to exploit spatial as well as temporal redundancy is shown in figure-A.3. The lowest frequency sub-band (LLL) has most of the information preserved, and hence is fed directly to entropy encoder. The seven other bands have sparsity, and hence are fed to Compressive Sensing (CS) block through Adaptive Measurement Scheme for further compression. To apply CS, multiple measurement vector (MMV) is formed by combining periodic columns of the wavelet coefficients of sub-band matrices (except for LLL sub-band). The MMV is converted into single measurement vector (SMV) by forming vectors of N1 dimension (N = 1024 or 2048) for application of CS as shown in Figure A.3. The number of columns for each vector is equal to (N/Sub-band Height).

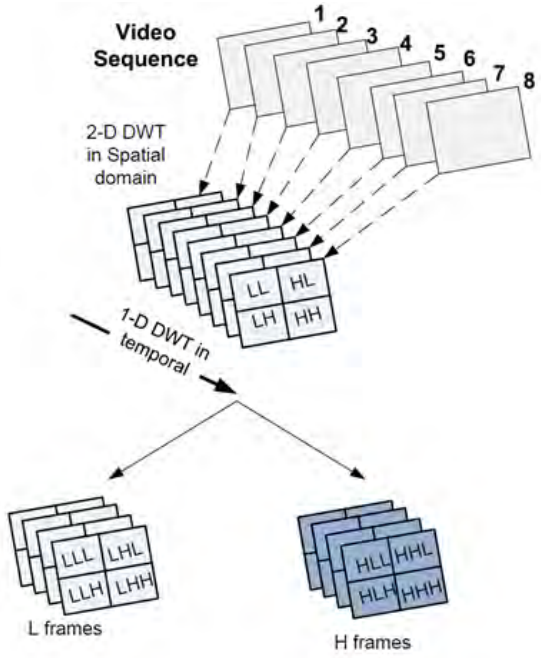


Figure A.3: 3D-DWT Process at SVC Encoder

The CS samples are generated from SMV by multiplying with randomly gener-

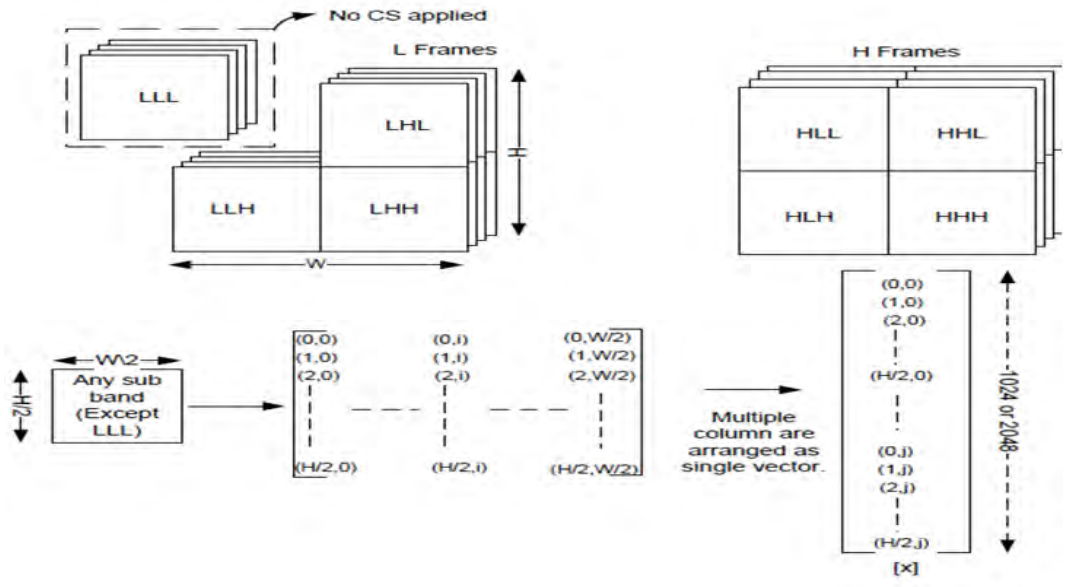


Figure A.4: Conversion from MMV to SMV at Encoder

ated adaptive Bernoulli Matrix (ϕ). The size of the Bernoulli matrix (ϕ) is $M \times N$, where M is greater than twice the non-zero elements in the vector X. Based on the value of M to be chosen, ϕ is selected from a codebook of eight Bernoulli matrices

that are already generated. The measured matrix (Y) is calculated from the equation $Y = \phi X$. The CS samples are then quantized and entropy coded by context based compression for variable length coding (VLC) to increase the compression efficiency [33, 34].

At SVC decoder, post entropy decoding, the Approximate Message Passing algorithm is used for recovery of coefficient of all sub-bands. The recovered coefficients are then given to inverse DWT block and video is reconstructed.

A simulation based performance verification of proposed CS based SVC was done by taking eight frames in each autonomous video cube, and a maximum of three levels of wavelet decomposition in spatial and temporal domains. The third level low frequency frame (LLL band) is considered as a base layer. Enhancement layers are comprised of the wavelet coefficients of all the other frames i.e., LLH, LHL, LHH, HLL, HLH, HHL and HHH bands. After applying the CS on 3D-DWT coefficients, the measurements are passed through the entropy coder. The complete base layer (LLL band) is entropy coded without compressive sensing as sparse information is unavailable. Loss-less compression is achieved when the video frames are recovered by utilizing the base layer, however the other layers undergoes lossy compression due to CS encoding and decoding. A constructed image from a video frame at different resolution is shown in figure-A.5

The performance comparison in terms of objective metrics like CR and PSNR is

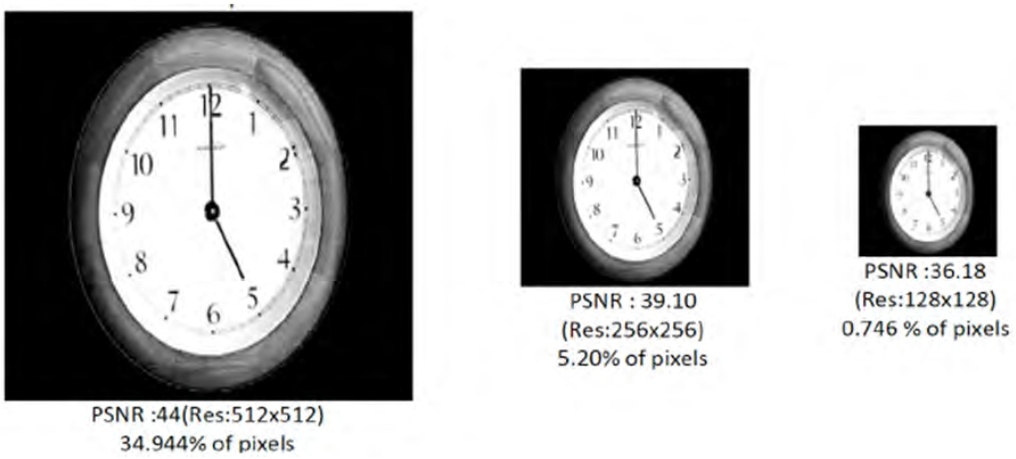


Figure A.5: Reconstructed frame from video at different resolutions

done for the proposed architecture with H.264 standards [109] for different video

sequences and presented in Table A.1.

Table A.1: Performance comparison of CS based SVC with standard H.264 model

CODEC	Video	CR	PSNR (dB)
Proposed CS Framework at Level-1	Clock	15.73	**43.32**
	Viplane	12.53	**40.48**
	Cyclone	8.34	**39.03**
Proposed CS Framework at Level-2	Clock	30.56	37.05
	Viplane	21.16	34.16
	Cyclone	14.85	31.69
Proposed CS Framework at Level-3	Clock	**55.16**	32.6
	Viplane	**56.26**	28.62
	Cyclone	**40.53**	26.88
Standard H.264 Model	Clock	62.3	42.65
	Viplane	37.8	40.57
	Cyclone	22.1	38.4

The PSNR realized by the proposed approach is comparable with H.264 model. The performance of the proposed architecture indicates that CS based SVC offers comparable performance with much simpler encoder configuration and is suitable for hardware implementation. The hardware implementation of CS based SVC has been initiated and some initial results are also reported in literature [32].

As an extension of this work, the sparsity detection algorithms developed under this dissertation could also be extended for CS based SVC application.