International Journal of Engineering Innovation & Research
Volume 5, Issue 5, ISSN: 2277 – 5668

An Analysis of the Performances of Compressive Sensing Algorithms OMP and KLT for Cognitive Radio

Charushila Axay Patel
Department of Electrical Engineering
B. V. Patel Institute of Technology (DS), Umrakh
Gujarat Technological University, Ahmedabad, Gujarat, India
charushila1977@yahoo.in
charushila1977@gmail.com

Dr. C. H. Vithalani
Department of Electronics & Communication Engineering
Government Engineering College, Rajkot
Gujarat Technological University, Ahmedabad, Gujarat, India
chvithalani@gecrajkot.ac.in
chvgec@gmail.com

Dr. Preetida Jani
Department of Electronics & Communication Engineering
Sardar Patel Institute of Technology
Mumbai University, Mumbai, Maharashtra, India
preeti.vinayakray@spit.ac.in

Abstract **– This paper provides a comparison between two widely used algorithms in the field of Compressive Sensing (CS), namely Orthogonal Matching Pursuit (OMP), and, Karhunen-Loève Transforms (KLT). As CS is one of the most essential techniques used by a Cognitive Radio (CR) for efficient usage of spectrum, it is required to be optimally simple, and, still, fast in working. The complexity here refers to the No. of computations a CR is required to make while using such algorithms and, this also, will in turn affect the effective requirement of hardware and power consumption. In this work, by means of simulations, we have tried to get an insight of working both this algorithms, OMP and KLT; and carried out the comparison between the two regarding their performances for the same experimental setup. We have discussed and evaluated their performances in terms of time, exact reconstruction of signal, percentage of error, and, complexity in terms of big-O, and, the probability of missed detection and probability of false alarm. From the simulation results we find that the OMP is quite promising CS tool as compared with the KLT in all these different aspects.**

As the CS is applicable to wideband spectrum sensing for CR and for varying sparsity environments, we are making comparison between the two that how the performance varies with different values of sparsity in frequency domain. We will carry out our further work on the bases of this work for modifying the OMP for CS.

Keywords **– CS, DWT, KLT, Measurement Matrix, Measurement Vector, OMP, Signal Detection, Signal Recovery. Sparsity, Sparsity Order.**

I. INTRODUCTION

A. Background and Related Work

J. Mitola [1] has introduced Cognitive Radio (CR) as one of those possible devices that could be deployed as SU (secondary user) equipments and systems in wireless networks. As originally defined, a CR is a self aware and "intelligent" device that can adapt itself to the wireless environment changes. This device is able to detect the changes in the wireless network to which it is connected and adapt its radio parameters to the new opportunities of spectrum usage that are detected by it. This functionality is called the "spectrum sensing" of a cognitive radio device.

Cognitive radio is an advanced software-defined radio that automatically detects its surrounding RF stimuli and intelligently adapts its operating parameters to network infrastructure while meeting user demands. Since cognitive radios are considered as secondary users for using the licensed spectrum, a crucial requirement of cognitive radio networks is that they must efficiently exploit under-utilized spectrum (denoted as spectral opportunities) without causing harmful interference to the PUs (Primary Users). Furthermore, PUs have no obligation to share and change their operating parameters for sharing spectrum with cognitive radio networks. Hence, cognitive radios should be able to detect independently spectral opportunities without any assistance from PUs; this ability is called spectrum sensing, which we can consider as one of the most critical components in cognitive radio networks. [2]

Cognitive radio systems typically involve primary users of the spectrum, who are incumbent licensees and secondary users who seek to use the spectrum opportunistically when the primary users are idle. The introduction of cognitive radios inevitably creates increased interference and thus can degrade the quality-of-service of the primary system. The impact on the primary system, for example in terms of increased interference, must be kept at a minimal level. Therefore, cognitive radios must sense the spectrum to detect whether it is available or not, and must be able to detect very weak primary user signals. Thus, spectrum sensing is one of the most essential components of cognitive radio. [21]

The problem of spectrum sensing is to decide whether a particular slice of the spectrum is "available" or not. That is, in its simplest form we want to discriminate between the two hypotheses,

$$H_0: y[n] = w[n], n = 1, N \quad (I.1)$$

$$H_1: y[n] = x[n] + w[n], n = 1, \ldots, N. \quad (I.2)$$

Where, x[n] represents a primary user's signal, w[n] is noise and n represents time. The received signal y[n] is vector, of length L. Each element of the vector y[n] could represent, for example, the received signal at a different antenna. [21]

The novel aspect of the spectrum sensing when related to the long-established detection theory literature is that the signal x[n] has a specific structure that stems from the use of modern modulation and coding techniques in contemporary wireless systems. Clearly, since such a structure may not be trivial to represent, this has resulted in substantial research efforts. At the same time, this structure offers the opportunity to design very efficient spectrum sensing algorithms. [21]

As CR networks are required to exploit the spectrum opportunities over wide frequency range for better spectrum utilization and obtaining optimized throughput. We can consider this on bases of Shannon's formula, according to which, under certain conditions, the maximum theoretically achievable throughput or bit rate is directly proportional to the spectral bandwidth. Therefore, the wideband spectrum sensing can help us achieve larger aggregate throughput by exploiting more available spectrum opportunities over a wide frequency range. [2]

Wideband Spectrum Sensing is the technique, which suggests the spectrum that we have to sense will have the frequency bandwidth more than the coherent bandwidth of the channel. The typical narrowband sensing techniques are limited in the way that they make use of single binary decision and cannot detect individual spectrum opportunity available in the wideband spectrum. Looking at the prevailing techniques nowadays, we can classify wideband spectrum sensing into two main classes: Nyquist (Rate based) spectrum sensing, and, Sub-Nyquist (Rate based) spectrum sensing. As the name suggests, the Nyquist (Rate based) spectrum sensing uses the sampling rate for spectral estimation at or more than the Nyquist rate. While the other one uses the rate of sampling below the Nyquist rate. [2]

Recently, compressed sensing/compressive sampling (CS) has been considered as a promising technique to improve and implement cognitive radio (CR) systems. In the area of signal processing, compressed sensing is one of the significant technique for extracting and reconstructing a signal by exploring the solution to underdetermine significant linear systems. The theory of compressive sensing has evolved owing to the issues in Image Processing, Video representation, Spectrum Sensing, etc. [22] As, in wideband radio one may not be able to acquire a signal at the Nyquist sampling rate due to the current limitations in Analog-to-Digital Converter (ADC) technology. Compressive sensing makes it possible to reconstruct a *sparse* signal by taking fewer samples than Nyquist sampling. In general, signals of practical interest may be only nearly sparse and typically the wireless signal in open networks are sparse in the frequency domain since depending on location and at duration the percentage of spectrum occupancy is low due to the idle radios. [3]

It is important that an efficient technique be explored that can minimize the amount of extracted data without any significant impact on quality of a signal. Therefore, under-sampling of k-space, violates the criterion of Nyquist, and there are increasing evidences of artifacts in Fourier reconstruction process. [22]

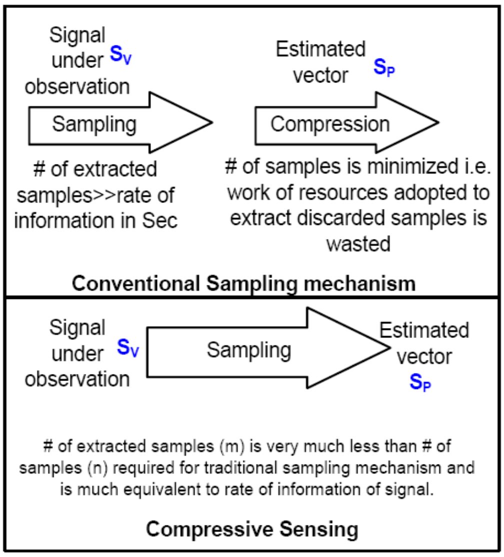


Fig. 1. Conventional Sensing and Compressive Sensing [22]

Fig.1 exhibits the conventional sampling of data as well as compressing sensing. Hence, the theory states that it is possible to extract a signal from minimal samples; however, the extraction of the signal can be 100% successful if the signal is being captured a minimal rate of information. This concept foretells that the signal is originally a sparse or belongs to some other form of transform domain. Hence, it is important to highlight certain definitions to understand compressive sensing as:

1. **Sparsity:** Various conventional forms of signals (image, audio, seismic data etc.) are sited in compressed mode based on suitability or the projection. We have found that after selecting the basis, maximal quantities of the projection coefficients usually become zero or very small values, which we neglect usually. Hence, the theory states that if the signal has n-number of non-zero coefficient, that signal is said to be n-sparse. The theory also states that if maximal quantities of the coefficient of projection are minimal enough that we can neglect then only the signal can be subjected to compression algorithms.
2. **Incoherence:** A statistical quantity evaluates the highest correlation between any two elements from two different matrices. If Ψ is considered to be square matrix of size n with $\psi_1, \psi_2, \ldots \psi_n$ columns and Φ is a non-square matrix of size m x n with $\varphi_1, \varphi_2, \ldots \varphi_m$ as rows, than the mathematical interpretation of coherence σ is:

   $$\mu(\Phi,\Psi) = \sqrt{n} \max | \Phi_k, \Psi_j \quad (I.3)$$

   Where the value of j lies between one to n and value of k lies between one and m. Hence, according to linearity principle, that formulates:

   $$1 \leq \mu(\Phi, \Psi) \leq \sqrt{n} \quad (I.4)$$

   Therefore, from the domain viewpoint of compressive sensing, the focus is much on the matrix incoherence factor adopted in sampled or in sensed signal Φx as well as the matrix that represents the basis where there is a sparse signal of interest Ψ.

International Journal of Engineering Innovation & Research
Volume 5, Issue 5, ISSN: 2277 – 5668

3. **Signal Extraction:** The process of extracting the signal in compressive sensing is equivalent to traditional one. The mathematical interpretation can be laid for the process of sensing SP considering S as signal,

$$x_P = \Phi x \quad (I.5)$$

The signal x and signal process SP are usually represented by real number of dimension n and m respectively. The traditional sensing concepts says that m should be equivalent to n in case certain levels of presence of sparse signals or compressible signals. The minimal value of m is permissible for the sensing matrices that are more incoherent within the original domain (or even in transform domain) where the signal is quite sparse. Hence, traditional sensing concepts uses Dirac delta functions while the problems is resisted by using Compressive sensing that considers random functions to speed up the process of signal extraction.

4. **Signal Reconstruction:** Majority of the existing concepts uses non-linear techniques to reconstruct the original signals in compressive sensing that is dependent on knowledge of basis of representation with a possibility of either compressible or sparse signals. Hence, the basis of representation of signal x is,

$$\Psi x_v = x \quad (I.6)$$

In the above equation, x_v is the sparse vector that represents coefficient of product of x and Ψ. The vector for measurement SP can be now represented as,

$$x_P = \gamma x_v. \quad (I.7)$$

The above equation shows γ as matrix of reconstructed signal which is equivalent to Φ. Ψ and is of size m x n. [22]

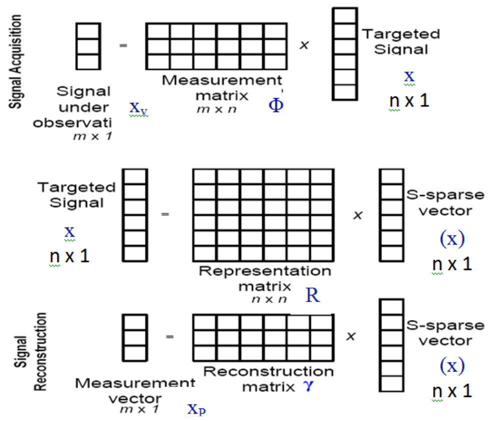


Fig. 2. Signal Extraction and Reconstruction Techniques in CS [22]

In CS, we can recover a signal with a sparse representation in some basis from a small set of non-adaptive linear measurements [4]. A sensing matrix takes few measurements of the signal, and the original signal can be reconstructed from the incomplete and contaminated observations accurately and sometimes exactly by solving a convex optimization problem [3]. The work of Candes, Romberg and Tao [5], [6] and Donoho [7] came as a major breakthrough in that they rigorously demonstrated, for the first time, that, under some very reasonable assumptions, the solution could be found using simple linear programming—thus rendering the solution practically feasible.

Compressive sensing has three important properties. First, the encoding is blind to the content of a signal (or data) and has low computational complexity suitable for fast, real-time usage. Secondly, the number of measurements required for exact recovery is approximately proportional to sparsity of the signal, not its size. Lastly, the decoding is adaptive in the sense that the quality of recovered data can improve under a fixed number measurements—or equivalently, the required number of measurements that achieves the same quality can decrease— when a more effective sparsifying basis becomes available. [9]

A.1 The Fundamental of Signal Detection

In signal detection, the task of interest is to decide whether the observation y was generated under H_0 or H_1. Typically, this is accomplished by first forming a test statistic ε(y) from the received data y, and then comparing ε(y) with a predetermined threshold η:

$$\varepsilon(y) \underset{H_0}{\overset{H_1}{\gtrless}} \eta \quad (I.8)$$

The fundamental problem of detector design is to choose the test statistic ε(y), and to set the decision threshold η in order to achieve good detection performance. [21]

A.2 Existing CS algorithms

Let us have a quick review of the existing techniques and contributory studies discussed by prior literatures. It is very important for investigation that what the existing status in the same domain be. The adoption of compressive sensing is not new and it has been done already in the prior studies. Various researchers have used this technique on various problems domains of signal processing. Still there are comparatively less implementation papers on compressive sensing until date. To get an insight we looked into different survey papers made available in prominent publication like IEEE journals.

Table 1 below gives a brief overview of some prominent survey papers published in IEEE and research done in Compressive Sensing.

Table I: A Survey on some Compression Techniques [22]

Authors	Problem Focused	Informative Factor	Limitation
Berger [23]-2010	Algorithms for sparse channel estimation	Discussion on empirical aspects	No discussion of prior research attempts
Gilbert [24]-2010	Sparse recovery using sparse random matrices	Techniques for-each guarantee	Performance effectiveness at techniques not discussed.
Potter [25]-2010	sparse reconstruction towards radar imaging	Algorithms for sparse reconstruction	Only theoretical illustrations.

Tropp [26]-2010	Sparse approximations	Algorithms on pursuit techniques	No comparative evaluation being conducted
Patel [27]-2011	Compressive sensing for Pattern recognition	Dictionary methods	Practically reviewed only a few implementation works
Wang [28]-2011	Compressive sensing for medical imaging	Existing studies on compressive sensing	No comparative evaluation being conducted
Dias [29]-2012	Compressive sensing	Usage of transform techniques	Practically reviewed only 4 implementation work
Mammeri [30]-2012	Image compression techniques in sensor networks	Discussed on compression algorithms	No discussion on Research gap, less focus on CS
Hayashi [31]-2013	Compressed sensing in signals	Discussion of algorithms e.g. FISTA (Fast Iterative Shrinkage-Thresholding Algorithms), NESTA (Nesterov's Algorithm)	Only theoretical illustrations
Kaur [32]-2013	Reconstruction techniques	Simplified techniques of compressive sensing	Practically reviewed only 5 implementation works
Ender [33]-2013	Compressed sensing in Radar imaging	Enriched theoretical discussion on domains	Only theoretical illustrations
Pudlewski [34]-2013	Challenges in compressive imaging	Discussed various techniques reconstruction in compressive techniques	No discussion on Research gap
Qaisar [35]-2013	Compressive Sensing and reconstruction algorithms	Reconstruction techniques	Studied complexity of a few implementation studies
Subban [36]-2014	Real time compressive tracking methods	Sparse representation techniques	No discussion on Research gap, Only theoretical illustrations
Zhou [37]-2014	Brief overview of CS	Applications, literature	Performance
Ali [38]-2015	Discussion on different present techniques for localization of user through CS	Literature review, comparison between CS and DS, graphs. Different techniques with issue and parameters	Less Significant discussion on effectiveness of compressive sensing

In the year 2010, Berger et.al [23] have published a survey paper towards compressive sensing exclusively focusing on the sparse channel estimation. Along with the theory, the author has discussed the conventional algorithms e.g. convex and greedy type towards sparse multipath channels. Gilbert and Indyk [24] are also doing similar type of study in same year. A unique survey study was found in same year by Potter et.al [25] who have investigated the techniques of sparse reconstruction towards radar imaging. Tropp and Wright [26] have also investigated the sparse approximation techniques. The authors have discussed the conventional algorithms e.g. convex relaxation, greedy pursuits, Bayesian, brute force etc., and discussed various algorithms of pursuits. Patel and Chellappa [27] have presented a discussion paper towards compressive sensing and spare representation. In 2011, Wang [28] have presented an editorial for compressive sensing with an exclusive focus on medical image processing. In 2012, Dias and Bandewar [29] have published a survey paper on compressive sensing and discussed the existing trends in it with respect to signal processing. In the same year Mammeriet. al. [30] have presented a review paper on image compression techniques exclusively considering sensor networks. The authors have discussed various compression schemes and finally discussed on effective principles on compression for sensor networks. In 2013, Hayashi et al. [31] have presented a survey paper with focus on design and development of sensing matrix and sparsity aspects in compressed sensing. In the same year, Kaur et. Al [32] have presented a review paper on reconstruction techniques. However, the study did not significantly yield any potential findings towards compressive sensing. Ender [33] has performed a study, which is almost similar to review work done by Potter et. al. [25]. Pudlewski and Melodia [34] have discussed on various impediments towards multimedia transmission with respect to compressive sensing. Qaisar et. al. [35] have presented a discussion on pathway of compressive sensing from hypothetical approach to practical approach. Subban et. al. [36] have investigated the algorithms for sparse representation and compressed tracking. In 2014, Zhou and Zhou [37] have presented an article on compressive sensing that are adaptable in multimedia coding. Same year, Ali [38] have surveyed some of the techniques of compressive sensing pertaining to localization. [22]

Standard CS algorithms have been used to reconstruct the original spectrum, such as basis pursuit (BP) [8] and others. In BP, it may take a long time to solve the linear program, even for signals of moderate length. Furthermore, when off-the-shelf optimization software is not available, the implementation of optimization algorithms may demand serious effort. We may consider alternate methods for reconstructing sparse signals from random measurements [10]. The DCT (Discrete Cosine Transform) and DFT (Discrete Fourier Transform) are also the candidates to apply for Compressive Sensing. Discrete Fourier Transform (DFT) is based on a fixed support set. The recovery with DFT as the sparsifying basis is suboptimal for a given signal, but it is used popularly in compressive sensing because it does not require any data-dependent adaptation. [9] The Discrete Cosine Transform, even though suboptimal, has been extremely popular in video coding. The principal reasons for the heavy usage of

DCT are 1) it is signal independent 2) it has fast algorithms resulting in efficient implementation and 3) its performance approaches that of KLT for a Markov-1 signal with large adjacent correlation coefficient.

The other family, which is called iterative greedy algorithms, received significant attention due to their low complexity and simple geometric interpretation. Apart from the Matching Pursuit (MP)[12], they include, Compressive Sampling MP (CoSaMP) [13], Subspace Pursuit (SP) [14], Iterative Hard Thresholding (IHT) [15]. Out of all these and among many algorithms designed to recover the sparse signal, orthogonal matching pursuit (OMP) algorithm has received much attention for its competitive performance as well as practical benefits, such as implementation simplicity and low computational complexity. Over the years, the OMP algorithm has long been considered as a heuristic algorithm hard to be analyzed. Recently, however, many efforts have been made to discover the condition of OMP ensuring the exact recovery of sparse signals. In one direction, studies to identify the recovery condition using probabilistic analyses have been proposed. Tropp and Gilbert showed that when the measurement matrix Φ is generated at random and the measurement size is about K log n, OMP ensures the accurate recovery of every fixed K-sparse signal with overwhelming probability. [10]

On the other hand, KLT, the Karhunen-Loève transform separates the input (= noise + signal) into ***UNCORRELATED*** components. The KLT is most widely used in applications such as multi-spectral analysis of satellite-gathered images through the spectral signature of imaged regions or, **for compression purposes**. The KLT is a unitary transform that diagonalizes the covariance of a discrete random sequence. As dispensation (quantization, coding etc) of any one coefficient in the KLT domain has no direct demeanor on the others, this decorrelation property is desirable. [11] It is also considered as an optimal transform among all discrete transforms based on a number of criteria. It is, used infrequently, however, as it is dependent on the statistics of the sequence i.e. when the statistics change so also the KLT. Because of this signal dependence, generally it has no fast algorithm. KLT has been used as a benchmark in evaluating the performance of other transforms. It has also provided an incentive for the researchers to develop signal independent (fixed) transforms that not only have fast algorithms, but also approach KLT in terms of performance.

A. 3 Research Gap

We can see that existing studies towards implementing compressive sensing on signal processing do exists with advantages as well as limitations too. However, a closer look into the studies being performed until the date was found with an obvious research gap. [22]

Less Effective Survey

We came across is the survey papers till date mentioned above as having less discussion of prior research contribution and an attempt to excavate its effectiveness by exploring either comparative analysis or by exploring research gap.

Less focus on Reconstruction

All the experimental based research papers have emphasized on implementing compressive sensing and quite less focus on its outcome with respect to complexities associated with reconstructed signals. Although reconstruction phenomenon is well defined in image signals, but importance of it is found in few video and speech signals. A closer look into the tabulated information will show that frequently used algorithms are projection-based, orthogonal matching pursuits, least absolute shrinkage and selection operator etc. However, the researchers have overlooked that although such techniques sometimes yield faster processing, but none of the above-discussed technique can be wisely adopted for reconstruction of a signal.

Ambiguity in implementing Sparsity matrix

Majority of the studies until date have considered sparsity as the image size, which will mean that when the image is divided into smaller sizes (like sub-images); the quantity of the samples will be required to be higher in size for the purpose of performing reconstruction of an image. However, adoption of such techniques drastically minimizes the probability of adopting compressive sensing with present definition of sparsity matrix in real-time.

B. Motivation for this work

There are many CS algorithms being researched and developed. Moreover, OMP and KLT are among the most promising techniques for that. There have been many works and papers available on different CS techniques; but according to the study conducted by us and as per the knowledge and understanding of the authors, there is not any other work actually that offers the comparative study of these two algorithms.

In addition, the OMP needs the information on sparsity level beforehand and the KLT is data dependent. It would be interesting to check which of these two will work satisfactorily for different sparsity levels in frequency domain. Therefore, we decided to work over comparing the two and undertook the experimental studies required for that.

In addition, the sparsity in frequency domain will be the main possible concern in CS research. When we estimate a signal through a CS system, we assume that the sparsity level is already given. This approach can fail when the sparsity assumptions given are invalid due to sparsity varying environment.

With advances in CR technology implemented, the secondary usage of the spectrum will also increase, making the spectrum denser. At this time, we will need the algorithm that can work well with varying sparsity levels.

We should be able to estimate the sparsity because it helps address a wide range of issues:

- Modeling assumptions
- The Number of measurements
- The Measurement matrix
- Recovery algorithms

Therefore, the algorithm, that is sparsity-robust and provides a satisfactory signal reconstruction is one, much anticipated development we can look for.

International Journal of Engineering Innovation & Research
Volume 5, Issue 5, ISSN: 2277 – 5668

C. Contribution of this work

In this work, considering the research gap mentioned above, we are comparing the performances of OMP and KLT for the signal recovery for the same input spectrum with the same experimental scenario. We will be presenting that to define a better suitable algorithm for the CS for the CR systems of future, which will be working in the scenario with varying sparsity levels.

We are proposing a sparsity-robust, greedy, OMP-based algorithm that may prove to be promising for CS in the follow-up paper of this.

D. Organization of the paper

In this paper, in section II, we will discuss about the OMP and its basic spectrum-sensing algorithm. In section III, we will discuss the KLT in detail. An insight into the algorithmic complexity of the two will be presented in the section IV. In addition, the experimental set up the results for both will be discussed in section V. In section VI, the performance metrics and results of the simulation for the OMP and KLT will be discussed. The section VII contains the discussion over the new proposed sparsity robust OMP algorithm. The table 2 here provides us with different parametric notations used throughout the paper.

Table II: Notations

Parameters	Description
x	Input signal in frequency domain
y	Output signal in frequency domain
w	Noise in measurements (Measurement Error)
$\hat{x}$	Estimated signal in R^d
v	N x 1 dimensional Data Vector for OMP
K	Sparsity
Φ	N x d Sparse Measurement Matrix
N	Measurement Vector length
d	Signal input length
φ_d	Columns of the measurement matrix
x_n	Rows of the measurement matrix or measurement vectors for OMP
a_m	N-dimensional approximation of v for OMP
r_m	N-dimensional residual for OMP
Λ_0	index set for OMP
λ_t	Index for OMP
η	Error tolerance for OMP
s	Karhunen-Loève transform of x
Q	KLT basis of x
Q_X^T	Karhunen-Loève transform of x^2
R_x	Correlation matrix for KLT
Λ	Diagonal matrix of eigenvalues for KLT
τ^c	The component of in the set τ for KLT
D	Length of dictionary for KLT
σ	Noise Variance
V_t	Threshold Voltage

II. ORTHOGONAL MATCHING PURSUIT (OMP)

Signal recovery can be considered as a problem dual to sparse approximation. Since x has only K nonzero components, the data vector v = Φx is a linear combination of m columns from Φ. In the language of sparse approximation, we say that v has an K-term representation over the dictionary Φ. Therefore, for recovering sparse signals, we can make use of sparse approximation algorithms. To identify the ideal signal s, we need to determine *which* columns of Φ participate in the data vector v. The idea behind the algorithm is to pick columns in a greedy fashion. At each iteration, we choose the column of Φ that is most strongly correlated with the remaining part of v. Then we subtract off its contribution to v and iterate on the residual. One hopes that, after m iterations, the algorithm will have identified the correct set of columns.

As we have considered, x $\in R^d$is a sparse vector, meaning its number of nonzero components Kis smaller than d. The support of x is the locations of the nonzero entries and is sometimes called its *sparsity pattern*. A common sparse estimation problem is to infer the sparsity pattern of x from linear measurements of the form

$$v = \Phi x + w, \quad \text{(II.1)}$$

where Φ ∈ R^{Nxd} is a known measurement matrix, v$\in R^N$ represents a vector of measurements and w $\in R^N$is a vector of measurements errors (noise). [39]

Sparsity pattern detection and related sparse estimation problems are classical problems in nonlinear signal processing and arise in a variety of applications including wavelet-based image processing and statistical model selection in linear regression. There has also been considerable recent interest in sparsity pattern detection in the context of *compressed sensing*, which focuses on large random measurement matrices A. We will analyze that scenario with random measurements.

Optimal subset recovery is NP-hard and it usually involves searches over all the $\binom{d}{K}$possiblesupport sets of x. Thus, most attention has focused on approximate methods for reconstruction.OMP is a simple greedy method that identifies the location of one nonzero component of xat a time. The best-knownanalysis of the performance of OMP for large random matrices is due to Tropp and Gilbert [10,40].

Among other results, Tropp and Gilbert show that when the number of measurements scales as

$$N \geq (1+\delta)4K\log(d) \quad \text{(II.2)}$$

for some δ >0, A has i.i.d. Gaussian entries, and the measurements are noise-free (w = 0), theOMP method will recover the correct sparse pattern of x with a probability that approaches one asd and K→ ∞. [39]

However, numerical experiments reported in [10] suggest that a smaller number of measurements than the above equation may be sufficient for asymptotic recovery with OMP. Specifically, the experiments suggest that the constant 4 can be reduced to 2. [39]

The theorem below proves this conjecture. Specifically, it is seen that the scaling in measurements

$$N \geq (1+\delta)2K\log(d-K) \quad \text{(II.3)}$$

is also sufficient for asymptotic reliable recovery with OMP provided both (n –k)and k → ∞. The result goes further by allowing uncertainty in the sparsity level k.

It also improves upon the Tropp–Gilbert analysis by accounting for the effect of the noises. While the Tropp–Gilbert analysis requires that the measurements are noise-free, we show that the scaling given with constant reduced to 2 is also sufficient when there is noise w, provided the signal-to-noise ratio (SNR) goes to infinity. [39]

The MP (Matching Pursuit)is one of the basic greedy algorithms that find one atom at a time. In OMP, following steps of the algorithm we find the one atom that best matches the signal given the previously found atoms. While in the next step, it finds the next one, to best fit the residual. [10]

- The algorithm stops when the error is smaller than the destination threshold.
- An enhanced version of the algorithm is the Orthogonal MP (OMP) that re-evaluates the coefficients by Least Squares after each round.
- It can reliably recover a signal with m nonzero entries in dimension d given O (mlnd) random linear measurements of that signal. Suppose that x is an arbitrary K-sparse signal in R^d, and let $\{x_1, x_2, \ldots, x_n\}$ be a family of N measurement vectors. Form an N x d matrix Φ whose *rows* are the measurement vectors, and observe that the N measurements of the signal can be collected in an N-dimensional data vector v=Φx. Here Φ is the *measurement matrix* and its columns are denoted by $\varphi_1, \ldots, \varphi_d$.
- Since x has only K nonzero components, the data vector v=Φx is a linear combination of columns m from Φ. In the language of sparse approximation, we say v is a data vector, that has an m-term representation over the dictionary Φ.
- Therefore, sparse approximation algorithms can be used for recovering sparse signals. To identify the ideal signal x, we need to determine *which* columns of Φ participate in the data vector v.
- The algorithm in a voracious or greedy fashion picks the columns. At each iteration, we choose the column of Φ that is most strongly correlated with the remaining part of v. Then we subtract off its contribution to v and iterate on the residual. One hopes that, after m iterations, the algorithm will have identified the correct set of columns.

Algorithm 1: OMP for Compressive Sensing

Presentation Matrix and Sensing Matrix Application

Input:
- x=wbs
- RIC=δ=delta=0.36

Output:
- The N x d Measurement Matrix, Θ=Φ
- The N x 1 dimensional data vector *b*

Procedure:

Initialize:
- Wavelet Decomposition Level, K=1

Decompose:
- Wavelet Decomposition of x, with db1, for generating Sparse Representation Matrix, Ψ

Calculate:
- Measure the sparsity level, sprlvl, of input using Gini Index method
- **If** Calculate the Size of the Dictionary, N=length(C), C=final decomposed signal
- Take tn=sprlvl
- Calculate the No. of Measurements, M= (1 + delta)*4*tn*log (N)

Design:
- Design M x N Sensing Matrix Φ
- Find N x 1 dimensional data vector *b*= Φ*C

Basic OMP Algorithm

Input:
- The N x d measurement matrix Θ=Φ
- The N x 1 dimensional data vector *b*
- Thesparsity level K of the ideal signal
- Maximum no. of iterations m
- Error tolerance,η.

Output:
- An estimate $\hat{x}$ in R^d for the ideal signal
- A index set Λ_m containing m elements from {1,….,d}
- An N-dimensional approximation a_m of the data b
- An N-dimensional residual $r_m = b - a_m$

Procedure:

Initialize:
- The index set I = ∅ and the residual r = b
- The set of non-zero elements as empty,
- The index set Λ_0=Ø, and,
- Iteration count t=1.

Repeat:
- The following, 'K' times:
- **Identify**
- Find the index λ_t that solves the easy optimization problem,
  $\lambda_t = \text{argmax}_{j=1,\ldots,d} |\langle r_{t-1}, \varphi_j \rangle|$.
- If the maximum occurs for multiple indices, break the tie deterministically.
- **Update**
- Add to the index set and the matrix of chosen atoms:
  $\Lambda_t \leftarrow \Lambda_{t-1}, \lambda_t$ and $\Phi_t \leftarrow [\Phi_{t-1}, \varphi\ \lambda_t]$.
  We here consider that Φ_0 is an empty matrix.
- A least square problem is solved to obtain a new signal estimate:
  $x_t \leftarrow \text{argmax}_x \|b - \Phi_t x\|_2$.
- Calculate the new approximation of the data and the new residual
  - $a_t \leftarrow \Phi_t x_t$
  - $r_t \leftarrow b - a_t$.
- $t \leftarrow t+1$, and find the new index λ_t, if t < m .
- The estimate for the ideal signal has nonzero indices at the components listed in Λ_m. The value of the estimate in component λ_j equals the j_{th} component of x_t .
- **Return** if t≥m.

III. Karhunen-Loève Transform (KLT)

The Kahrunen-Lo`eve Transform (KLT) is a classical transform for a signal. KLT is optimal in the sense that it completely decorrelates the signal and maximally makes the information contained in the signal compact. Given its optimality in revealing the sparsity of a signal, it should be natural to explore the use of the KLT basis in compressive sensing decoding. It separates the input (= noise + signal) into uncorrelated components. The KLT is most widely used in applications such as multi-spectral analysis of

International Journal of Engineering Innovation & Research
Volume 5, Issue 5, ISSN: 2277 – 5668

satellite-gathered images through the spectral signature of imaged regions or, **for compression purposes**. [9]

Consider $x \in C^N$, a complex-valued, wide-sense stationary signal with mean zero (for simplicity). The correlation matrix of x can be computed numerically: $R_x = E[xx^H]$, where the superscripted H denotes Hermitian transpose (i.e., $x^H = X^{*T}$). R_x is real and symmetric, and the eigen-decomposition, $R_x = Q\Lambda Q^H$, gives columns of Q the eigenvectors of R_x and Λ a diagonal matrix of the eigenvalues. Q is an orthogonal matrix, thus $Q^{-1} = Q^H$.

The representation $s = Q^H x$ is known as Karhunen-Lo`eve Transform (KLT) of x, and we call Q the KLT basis or an uncorrelated representation s, whose correlation matrix has zero cross-correlation terms. In other words, s fully describes x without any statistical redundancy.

The KLT matrix Q is computed from the correlation matrix of input signal x, $R_x = E[xx^H]$. Similarly, the correlation matrix of the compressive measurements y is $R_y = E[yy^H]$. By compressive encoding $y = \Phi x$, we know: $R_y = E[\Phi xx^H \Phi^T] = \Phi E[xx^H] \Phi^T$. So, $R_y = \Phi R_x \Phi^T$.

Note that Φ is not a square matrix. Using the pseudo-inverse $(\Phi^T)y$, we can have the following expression:

$$R_y(\Phi^T)y = \Phi R_x \qquad \text{(III.1)}$$

Here, we find that we have been compressively measuring R_x in $R_y(\Phi^T)y$, which can be approximated from $y = \Phi_x$ that we used to encode our data x. Thus, compressive measurement has sufficient information to recover R_x from above equation. Below is a procedure to estimate KLT basis Q with compressive measurements in four steps:

Algorithm 2: KLT for Compressive Sensing

Presentation Matrix and Sensing Matrix Application

Input:
- x=wbs
- RIC=δ=delta=0.36

Output:
- The N x d Measurement Matrix, Θ=Φ
- The N x 1 dimensional data vector *x*

Procedure:

Initialize:
- Wavelet Decomposition Level, K=1

Decompose:
- Wavelet Decomposition of x, with db1, for generating Sparse Representation Matrix, Ψ

Calculate:
- Measure the sparsity level, sprlvl, of input using Gini Index method
- **If** Calculate the Size of the Dictionary, N=length(C), C=final decomposed signal
- Take tn=sprlvl
- Calculate the No. of Measurements, M= (1 + delta)*4*tn*log (N)

Design:
- Design M x N Sensing Matrix Φ
- Find N x 1 dimensional data vector *x*= Φ*C

KLT-based CS

Input:
- The N x d measurement matrix Θ=Φ
- The N x 1 dimensional data vector *x*

Output:
- An estimate $\hat{x}$ in R^d for the ideal signal
- A index set Λ_m containing m elements from {1,….,d}
- An N-dimensional approximation a_m of the data b

Procedure:

Compute:
- Correlation Matrix,

  $R_x = E[xx^H]$,

  or, Define the Covariance Matrix,

  $[\hat{C}]_x = \frac{1}{n}\sum_{i=1}^{n} x_i x_i^T$,

  where x_i is the data vector.
- Find the PSD of X, $S_X(\omega)$.
- Define the Diagonal Matrix Λ with non-increasing entries,

  $\Lambda = [Q][R]_x[Q]^T$,

  where, the matrix Λ contains the eigenvalues on the diagonals,

  $(\lambda_1, \lambda_2, \ldots, \lambda_n) = \text{diag}(=[Q][R]_x[Q]^T) = Sx(2\pi k/n)$.
- Find the KLT from Λ, as the columns of [Q] are the basis vectors of the KLT
- Find eigen values for the matrix
- Sort the eigen values to remove least eigen vector for compression
- Obtain Q by eigen-decomposition of $Rx = Q \Lambda Q^H$
- Reconstruct by $\hat{x} = Q\hat{s}$, or $s = Q^{-1}x$

We denote the transformed version of x as,

$$y = Q_X^T x. \qquad \text{(III.2)}$$

Since Q_x is unitary, it follows that,

$$E[||x-\hat{x}||^2] = E[||y-\hat{y}||^2] = \sum_{m=1}^{N} E[|y_m - \hat{y}_m|^2], \qquad \text{(III.3)}$$

where $\hat{y} = Q_X^T \hat{x}$. The key point is that the components of y are uncorrelated. Therefore, in terms of the components $(y_1, y_2, \ldots, y_N)$ a simple answer can be given. First, if the component y_m is retained, then clearly its corresponding estimate is $\hat{y}_m = y_m$. However, if y_m is *not* retained, then its corresponding estimate is $\hat{y}_m = 0$; none of the other components of the vector y contain anything relevant about y_m. The best k-dimensional approximation space is therefore easily found in terms of y. Denote the set of the k-indices corresponding to the retained components of y by τ . Then, the incurred distortion can be expressed as

$$E[||x-\hat{x}||^2] = \sum_{m\in\tau^c} \lambda N. \qquad \text{(III.4)}$$

Where τ^c denotes the complement of in the set τ = {1, 2,….., N}. Hence, the best k-dimensional approximation is given by the eigenvectors corresponding to the k-largest eigenvalues.

After the sparse representation consideration for reconstruction we have to, signal vector $x \in \mathbb{R}^N$ can be expanded in an orthonormal basis $Q \in \mathbb{R}^{d\times d}$ in the form of x = Qs. If the coefficients $s \in \mathbb{R}N$ have at most *k* non-zero components, we are calling x a *k*-sparse signal with respect to Q, andQ is the correlation matrix of KLT. Many natural signals, we can represent, the same way, as a sparse signal in an appropriate basis. With a linear measurement matrix $\Phi^{N\times d}$, N≪d, CS measurements of a *k*-sparse signal x are collected in the form of y = Φx = ΦQs. If A≙ ΦQ satisfies the Restricted Isometry Property (RIP), then the sparse coefficient vector s can be recovered accurately (with very high probability) via the following linear program,

$$\hat{s} = \arg\min_{\tilde{s}} ||\tilde{s}||_{l_1}, \text{ subject to } y = \Phi Q\tilde{s}. \qquad \text{(III.5)}$$

Afterwards the signal of interest can be reconstructed by, $\hat{x} = Q\hat{s}$.

IV. Comparison between OMP and KLT

Here, both the OMP and KLT use the sensing matrices for compressive sensing.

In OMP, we define the length of measurement vector by defining the no. of measurements in our simulation that determines the no. of columns or no. of iterations to take place. The algorithm picks the columns in a greedy fashion here. At each iteration, we choose the column of Φ that is most strongly correlated with the remaining part of v. Then we subtract off its contribution to v and iterate on the residual. One hopes that, after m iterations, the algorithm will have identified the correct set of columns. Here if a K sparse signal is there, the no. of iterations are defined to be equal to K. Therefore, we can define relation between the no. of measurements and the no. of iterations from the relation between the no. of measurements and sparsity level.

For KLT, we are decomposing the signal by removing the eigenvectors that are columnsof the correlation matrix. The matrix is sized normally N x N representing the atoms of the signal in form of the eigenvectors. The bases of the KLT are the eigenvectors of the auto. Assuming that R_x is the auto of the signal X, when the KLT is calculated over the vector X, the R_x can be estimated by: $R_x = \langle XX^T \rangle$. Let Λ be a matrix whose columns constitute a set of orthonormal eigenvectors of R_x, so that $QQ^T = I$ and; T $R_x = Q\Lambda Q^T$ Where Λ is the diagonal matrix of non-null eigenvalues: $\Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_k)$. For every vector X, through the matrix Q^T , we can obtain the sparse coefficients vector Θ as: $\Theta = Q^T X$.

We decompose X's correlation matrixR_x into $Q^{-1} * \Lambda * Q$. Then we create the N x (N-r) diagonal matrix Λ that has the effect of removing the columns of Q that correspond to the smallest diagonal entries in D. These smallest entries in D should make up (1-err) % of the total value of D. Now, $Q^{-1}*x$ will transform the highly correlated signal x, into one with no correlation. Then $\Lambda^T*Q^{-1}*x$ will throw away the least significant entries in $Q^{-1}*x$. This (Λ^T*Q^{-1}) will be our compression matrix. The corresponding decompression matrix will be Q (or equivalently, Q* Λ). So in a real system, we would predetermine $T= \Lambda^T*Q^{-1}$, the compression matrix, and $T^T=Q* \Lambda$, the decompression matrix, based the assumption that our future signal has a known correlation matrix. We right multiply the signal x by T, send the signal (which has N-r samples instead of N samples), receive it, right multiply it by T^T, and we have our reconstructed signal. All of this compression-decompression can me modeled by T^T*T*x, or equivalently $Q* \Lambda* \Lambda^T*Q^{-1} = Q* \Lambda`*Q^{-1}$, where $\Lambda' = \Lambda* \Lambda^T$. Λ` is the diagonal matrix with zeros in the diagonal entries corresponding to the thrown away columns of Q, and ones in the diagonal entries corresponding to the retained columns of Q.

Here, the KLT kernel is a unitary matrix, Q, whose columns, vectors q_k (arranged in descending order of eigenvalue amplitude), are used to transform each zero-meaned vector:

$$y_k = V^T(x_k-m_k). \quad \text{(IV.1)}$$

V. The Experimental Setup

We carried out simulations for checking the performances of KLT and OMP for signal recovery for spectrum sensing purpose. Our main aim presently is to provide comparison between the two basic algorithms. Based on the obtained results and the related works suggesting modifications in basic OMP algorithm, we will propose a new modified algorithm for OMP.

The simulation was carried out on Matlab 2011b version for both the algorithms. The system was having the processor Intel Core i5, M460, with a 64-bit OS, and 2.53 GHz Clock Speed.

We shall consider, at baseband, a wideband spectrum range [0 MHz –60 MHz] containing 30 channels of 2 MHz each and encode it as c = c1, c2, … , cn; where n = 30. Every channel may be possibly occupied by a Primary User (PU) using digital modulation scheme either 16-PSK or 16-QAM. Therefore, the symbol rate will be 2 MHz, number of samples per symbol will be 16, and number of symbols in a frame can be chosen to be 512. Here, we shall consider the Nyquist sampling frequency, fs= 128 MHz and the sampling number, N = 8192.

We took the readings over a span of 60 MHz having 30 channel components having the BW of 2 MHz each. The simulation parameters are tabulated in Table 3. We first demonstrate the sub-Nyquist rate reconstruction performance using Fig. 5. Let number of samples per symbol be M = 16 and let N= Sampling No. = 8192for Gaussian random matrix for an AWGN channel. Fig. 5 shows the results from OMP and KLT for different sparsity levels and the time required for simulation for each and signal recovery in terms of correlation between the input and output.

Table III: Simulation Parameters for OMP and KLT Comparison

Parameter	Value
The Wide Band Spectrum Band	60 MHz
No. of Channels	30
Band Width of a single channel	2 MHz
Modulation Scheme	16-QAM
No. of Samples per Symbol, M	16
No. of Symbols per Frame	512
Sampling No., N	8192
Sampling Frequency, Fs	128 MHz
SNR	5 dB
Sensing Parameters	
Wavelet Transform (as Sparsifying	Daubechies
Channel Type	AWGN
Sensing Algorithm	Basic OMP, KLT

VI. The Performance Metrics and Results

The results are depicted here in form of graphical plots for both OMP and KLT, for different sparsity levels. These are expressed in terms of the Time taken for simulation (sec) and the Correlation between the input and output of the algorithms.

Here the time taken for the simulation by its name itself defines the speed of the algorithm on an average. This shows how quickly the algorithm succeeds in recovering the unknown input spectral component when implemented in a cognitive radio working as a secondary user and it indicates how fast the particular algorithm will be out of the two we have considered.

The correlation between the input and output on the other hand, is used here to show how much the algorithm succeeds in recovering the unknown input spectrum.

The correlation is derived for both the algorithms using the xcorr and mscohere commands in Matlab. It is derived by taking the root, mean and square of the two outputs obtained by the two commands. The xcorr gives the correlation between the two signals (here the input and output spectrums) in terms of cross-correlation between the two random processes. On the other hand, the mscohere finds the magnitude squared estimate of the input signals x and y (here the input and output spectrums) indicating how well the two signals correspond to each other at each frequencies.

The sparsity level is measured here using the Gini Index. As the Gini index is one of the most reliable measures for sparsity, we have opted to use it here.

Gini index is used to express the percentage of sparsity that gives its value in true sense. It is one of the most reliable measures for sparsity,

$$GI(x) = 1-2\sum_{k=1}^{N}\frac{|x_k|}{||x||_{l_1}}\left(\frac{N-k+1/2}{N}\right) \quad (V.1)$$

Here the vector $\underline{f}$ = [f(1), f(2), ……., f(N)] is given with its elements re-ordered and represented by $f_{[k]}$ for k = 1, 2, ….., N, where $|f[1]| \le |f[2]|, \ldots, \le |f[N]|$, and $||\underline{f}||1$ is the l_1 norm of the function $\underline{f}$.

The Gini index possesses the values between 0 to 1. So, percentage representation required multiplication with 100.For us, Sparsity K, is,

Table IV: Sparsity Conversion corresponding to GINI Index representation

Gini Index	No. of Active Spectrum
100	1
79.3103	7
62.069	12
41.3793	18
20.6897	24
0	30

Performance Analysis:

To evaluate efficiency of the algorithm, it can be taken into account properties of the algorithm (complexity, velocity or speed, memory consumption), the amount of compression, and that how closely the reconstruction resembles the original signal. In this work, we will focus on the complexity of the algorithms, speed, and, a quantification of the difference/ similarity between the original signal and its reconstruction after compression. Moreover, the probability of missed detection P_M, and, the probability of false alarm P_F, or their average across channels, are among the prominent performance metrics used to characterize the wideband sensing performance of these algorithms.

THE PARAMETERS AFFECTING THE SENSING PERFORMANCE:

Before we can analyze the performances of these two algorithms and compare them, we must get the insight of the experimental parameter considerations used here for performing the sensing.

As we have applied the wideband signal to the algorithms, for compressed sensing, it is required to sparsified and then sensed by measurement matrix. These two actions affect the sensing performance of the two algorithms. Therefore, we will have to be introduced to the basics of the two actions we perform. Let us have a quick look into these two:

<u>The Sparsifying Basis (DWT)</u>:

For both this algorithms, we sparsify the signal before application to the algorithms. This is done to make it sure that the signal under consideration is sparse in true sense. We have used the Discrete Wavelet Transforms (DWTs) as sparsifying basis. It has its own excellent space frequency localization property. Application of DWT in 1D signal corresponds to 1D filter in each dimension. The use of DWT as sparsifying basis enables the removal of blocking artifacts.

Dilations and translations of the “Mother function," or “analyzing wavelet" $\psi(x)$; define an orthogonal basis, our wavelet basis:

$$\psi_{(s,l)}(x) = 2^{-s/2}\psi ¡2^{-s}x - l). \quad (V.2)$$

The variables *s* and *l* are integers that scale and dilate the mother function ψ to generate wavelets, such as a Daubechies wavelet family. The scale index *s* indicates the wavelet's width, and the location index *l* gives its position. Notice that the mother functions are rescaled, or “dilated" by powers of two, and translated by integers. What makes wavelet bases especially interesting is the self-similarity caused by the scales and dilations. Once we know about the mother functions, we know everything about the basis. [42]

To span our data domain at different resolutions, the analyzing wavelet is used in a scaling equation:

$$W(x) = \sum_{k=-1}^{N-2}(-1)^k h_{k+1}\psi(2x+k) \quad (V.3)$$

where $W(x)$ is the scaling function for the mother function ψ, and h_k are the *wavelet coefficients*. The wavelet coefficients must satisfy linear and quadratic constraints of the form

$$\sum_{k=0}^{N-1} h_k = 2, \sum_{k=0}^{N-1} h_k h_{k+2l} = 2\delta_{l,0} \quad (V.4)$$

where $\delta$is the delta function and l is the location index. One of the most useful features of wavelets is the ease with which a scientist can choose the defining coefficients for a given wavelet system to be adapted for a given problem. The Haar wavelet is even simpler, and it is often used for educational purposes. [42]

It is helpful to think of the coefficients $\{h_0,\ldots h_n\}$ as a filter. The filter or coefficients are placed in a transformation matrix, which is applied to a raw data vector. The coefficients are ordered using two dominant patterns, one that works as a smoothing filter (like a moving average), and one pattern that works to bring out the data's "detail" information. These two orderings of the coefficients are called a *quadrature mirror* filter pair in signal processing parlance.

Now, let's look at how the wavelet coefficient matrix is applied to the data vector. The matrix is applied in a hierarchical algorithm, sometimes called a *pyramidal algorithm*. The wavelet coefficients are arranged so that odd rows contain an ordering of wavelet coefficients that act as the smoothing filter, and the even rows contain an ordering of wavelet coefficient with different signs that act to bring out the data's detail. The matrix is first applied to the original, full-length vector. Then the vector is smoothed and decimated by half and the matrix is applied again. Then the smoothed, halved vector is smoothed, and halved again, and the matrix applied once more. This process continues until a trivial number of "smooth-smooth- smooth..." data remain. That is, each matrix application brings out a higher resolution of the data while at the same time smoothing the remaining data. The output of the DWT consists of the remaining "smooth (etc.)" components, and all of the accumulated "detail" components. [42]

We have used db1-db4 of Daubechies family of wavelet transforms as sparsification basis for our experiments. We have observed the effects of using these all on the performances of both the OMP and KLT.

Application of DWT in 1D signal corresponds to 1D filter in each dimension. The Daubechies wavelet transforms are defined in the same way as the Haar wavelet transform by computing the running averages and differences via scalar products with scaling signals and wavelets. The only difference between them consists in how these scaling signals and wavelets are defined.

The wavelet function (mother wavelet) is orthogonal to all functions which are obtained by shifting the mother right or left by an integer amount and the mother wavelet is orthogonal to all functions which are obtained by dilating (stretching) the mother by a factor of 2_j (2 to the jth power) and shifting by multiples of 2_j units. The orthogonality property means that the inner product of the mother wavelet with itself is unity, and the inner products between the mother wavelet and the aforementioned shifts and dilates of the mother are zero. The collection of shifted and dilated wavelet functions is called a wavelet basis. The grid in shift-scale space on which the wavelet basis functions are defined is called the dyadic grid. The orthonormality of the Daubechies wavelets has a very important mathematical and engineering consequence: any continuous function may be uniquely projected onto the wavelet basis functions and expressed as a linear combination of the basis functions. The collection of coefficients which weight the wavelet basis functions when representing an arbitrary continuous function are referred to as the Wavelet Transform of the given function.

For the Daubechies wavelet transforms, the scaling signals and wavelets have slightly longer supports, i.e., they produce averages and differences using just a few more values from the signal. This slight change, however, provides a tremendous improvement in the capabilities.

The input Daubechies Wavelet as mother wavelet is divided into 8 non-overlapping multi-resolution sub-bands by the filters, namely db1, db2, db3up to db8, where db is acronym for Daubechies. The sub-band is processed further to obtain the next coarser scale of wavelet coefficients, until some final scale "N" is reached. When a signal is decomposed into 8 levels, the db6 sub-band signal best reflects the original signal, since according to the wavelet theory, the approximation signal at level n is the aggregation of the approximation at level n-1 plus the detail at level n-1. [41]

- This wavelet type has balanced frequency responses but non-linear phase responses.
- They use overlapping windows, so the high frequency coefficient spectrum reflects all high frequency changes.
- These wavelets are useful in compression and noise removal of audio signal processing.

For N ∈N, a Daubechies wavelet of class D-2N is a function $\psi = {}_N\psi \in L^2$ (R) defined by,

$$\Psi(x) := \sqrt{2}\sum_{k=0}^{2N-1}(-1)^k h_{2N-1-k}\,\varphi(2x-k). \quad (V.5)$$

Where $h_0, \ldots, h_{2N-1} \in R$ are the constant filter coefficients satisfying the conditions N−1.

$$\sum_{k=0}^{N-1} h_{2k} = \frac{1}{\sqrt{2}} = \sum_{k=0}^{N-1} h_{2k+1}. \quad (V.6)$$

As well as for $l = 0, 1, \ldots, N-1$,

$$\sum_{k=2l}^{2N-1+2l} h_k h_{k-2l} = \begin{cases}1, if\ l=0\\ 0, if\ l\neq 0\end{cases}. \quad (V.7)$$

and where $\phi = N\phi : R \to R$ is the (Daubechies) scaling function (sometimes also "scalet" or "father wavelet"), given by the recursion equation,

$$\varphi(x) = \sqrt{2}\sum_{k=0}^{2N-1} h_k(2x-k). \quad (V.8)$$

And obeying,

$$\varphi(x) = 0, \text{ for } x \in R \setminus [0, 2N-1]. \quad (V.9)$$

As well as

$$\int_R \varphi(2x-k)(2x-l)dx = 0, \text{ for } k\neq l \quad (V.10)$$

There are N equations given by the orthonormality conditions (V.7). Together with (V.6) this gives in total N + 2 equations for the 2N filter coefficients h_k. Hence, for N = 1, they are over-determined, for N = 2 they are unique (if they exist), and for N > 2 they are underdetermined. However, once the filter coefficients are given, (V.6) demonstrates the existence and uniqueness of a function ϕ satisfying (V.8) and the normalization condition R Rϕ = 1, for a given sequence of $h_0, \ldots, h_{2N-1}$. [43]

The computational complexity of applying DWT basis for sparsification one time is equivalent to O(N), where N is the no. of time samples.

Restricted Isometry Property and the No. of measurements:

As an alternative to coherence and to probabilistic analysis, a large number of algorithms within the broader field of CS have been studied using the restricted isometry property (RIP) for the matrix Φ[16]. A matrix Φ satisfies the RIP of order K if there exists a constant $\delta \in (0,1)$ such that,

$$(1-\delta)||x||_2^2 \le ||\Phi x||_2^2 \le (1+\delta)||x||_2^2, \qquad \text{(V.11)}$$

holds for all x such that $||x||_0 \le K$ [17]. In other words, Φ acts as an approximate isometry on the set of vectors that are K -sparse. Much is known about finding the matrices satisfying the RIP. For example, if we draw a random N x d matrix Φ whose entries ϕ_{ij} are independent and identically distributed sub-Gaussian random variables, then provided that

$$N = O(K\log(d/K)/\delta^2) \qquad \text{(V.12)}$$

with high probability,Φ will satisfy the RIP of order K [17]. When it is satisfied, the RIP for a matrix provides a sufficient condition to guarantee successful sparse recovery using a wide variety of algorithms [16].

The No. of measurements is decided based on the criteria of noise present in the measurement matrix. If the noise is absent, i.e., the measurements are error-free, the no. of measurements can be considered to be

$$N \ge (1+\delta)4K\log(d). \qquad \text{(V.13)}$$

While we expect the noise to be present, the No. of measurements can be,

$$N \ge (1+\delta)2K\log(d-K). \qquad \text{(V.14)}$$

The RIC (the Restricted Isometry Constant), δ here, was taken to be 0.36.

THE METRICS FOR EVALUATING THE SENSING PERFORMANCE OF THE OMP AND KLT:

Here we have focused on the computational complexity of the algorithms, speed, and, a quantification of the difference/ similarity between the original signal and its reconstruction after the application of compressive sensing. Also, the probability of missed detection, P_M, and, the probability of false alarm, P_F; or, their average across channels, are some of the prominent performance metrics used to characterize the wideband sensing performance of these algorithms.

We will now have a brief analysis of these performance metrics for the algorithms under consideration.

Computational Complexity:

Normally the computational complexity is equivalent to the no. of steps an algorithm takes to solve the problem as a function of the input size.

However, we can classify the complexity as Time Complexity and Space Complexity.

Time complexity – The number of steps required by an algorithm varies with the size of the problem it is solving in the way. Time complexity is normally expressed as an order of magnitude, e.g. O(N^2) means that if the size of the problem (N) doubles then the algorithm will take four times as many steps to complete.

Space complexity – The amount of storage space required by an algorithm varies with the size of the problem it is solving in the way. Space complexity is normally expressed as an order of magnitude, e.g. O(N^2) means that if the size of the problem (N) doubles then four times as much working storage will be needed.

OMP:

At each stage, OMP computes residual correlations and solves a least-squares problem for the new solution estimate. OMP builds up the active set one element at a time. Hence, an efficient implementation would necessarily maintain a Cholesky factorization of the active set matrix and update it at each stage, thereby reducing the cost of solving the least-squares system. In total, k steps of OMP would take at most $4k^3/3+knN+O(N)$ flops. Without any sparsity assumptions on the data, OMP takes at most n steps, thus, its worst-case performance is bounded by $4n^3/3+n^2N+O(N)$ operations.

Three main points are there: (1) that in each step of the algorithm, the residual vector can be written as a matrix times a sparse signal, (2) that this matrix satisfies the RIP, and (3) that consequently a sharp bound can be established for the vector h^l of inner products. The RIP of order K+1 (with $\delta < 1(3\sqrt{K})$) is sufficient for OMP to recover exactly any K-sparse signal in exactly K iterations. [17]

However, for restricted classes of K-sparse signals (those with sufficiently strong decay in the nonzero coefficients), a relaxed bound on the isometry constant can be used. If we wish to use the RIP of order K+1 as a sufficient condition for exact recovery of all K-sparse signals via OMP (as we have), then little improvement is possible in relaxing the isometry constant δ above $1/(3\sqrt{K})$. In particular, there exists a matrix satisfying the RIP of order K+1 with $\delta \le 1/\sqrt{K}$ for which there exists a K-sparse signal that $x \in R^N$ cannot be recovered exactly via K iterations of OMP. [17]

As we know, the no. of measurements is $N = O(K\log(d/K)/\delta^2)$ based on RIP condition to recover the K-sparse signal, we see that finding a matrix satisfying the RIP of order K+1 with an isometry constant $\delta < 1/(3\sqrt{K})$ will likely require $N = O(K^2\log(d/K))$ random measurements. [17]

However, if one wishes to guarantee exact recovery of all K-sparse signals via OMP, then there is little room for further reducing N. When $N \le K^{3/2}$, for most random N x d matrices Φ there will exist some K-sparse signal $x \in R^N$ that cannot be recovered exactly via K iterations of OMP. [17]

Tropp and Gilbert have shown that when the number of measurements scales as $N \ge (1+\delta)4K\log(d)$ for some $\delta > 0$, A has i.i.d. Gaussian entries, and the measurements are noise-free (w = 0), the OMP method will recover the correct sparse pattern of x with a probability that approaches one as d and $K \to \infty$. [39]

KLT:

The KLT has no structure since it depends on the autocorrelation matrix of the input signal. The implementation of KLT involves the estimation of the auto-correlation matrix of the data sequence, its diagonalization, and the construction of the basis vectors. Therefore, the basis vectors are depended on the signal, which cannot be predetermined, and must be completely repeated whenever any new data is added. The KLT requires much of computation time for the eigenvector decomposition; some approximated approaches to overcome this problem are required to be developed. The coefficients in the KLT domain are sparse by nature. In the KLT domain, we can use a heuristic choice of K=k+1 when the improvement in the accuracy of the reconstruction is achieved. [18]

The correlation between the input and output obtained with KLT is much better than that achieved with the OMP. However, as the results show, that decreases with increasing density of spectrum components.

Given a typical signal x such as the signal received from a wideband channel, if Q is the KLT basis of x, then Q_x is guaranteed to be highly sparse for any x. Decoding with the KLT basis in general can use fewer measurements to achieve the recovery accuracy, and more importantly it can work with any signal. However, the KLT basis is data-dependent. This means that it needs to be updated from time to time if the signal is non-stationary. To reduce required measurements in recovering an updated KLT basis, we shall still exploit whatever sparsity the signal may possess in some domains. Fortunately, we can update the optimal basis relatively infrequently, that is, we only do this after the signal has changed substantially. We can make effective use of optimal KLT basis in compressive sensing decoding, in spite of the fact that KLT is signal dependent and needs to be updated from time to time. [19]

In the KLT we are applying a measurement matrix of the size Nxd and then the N x N correlation matrix is applied which represents the N eigenvalues and corresponding N eigenvectors for decorrelating the signal components and decomposition. The basis vectors of the KLT are the eigenvectors of the correlation matrix.

If we take the smallest r eigenvalues and zero them out, leaving err% of the sum of eigenvalues, then s, a realization of S will be compressed by a factor of N/(N-r), and contain err% of the energy of the original signal.

The KLT comprises three distinct processing stages: 1) covariance formation, 2) eigenvector calculation, and 3) eigenspace transform. [20]

Numerical computation of eigenvalues:

As we already know, for using KLT as the Compressive Sensing tool, we have to apply the eigenvalue decomposition first and eigenvector removal. We have used the eigenvector removal method here. For that, the computation of eigenvalues and eigenvector is to be carried out.

Suppose that we want to compute the eigenvalues of a given matrix. If the matrix is small, we can compute them symbolically using the characteristic polynomial. However, this is often impossible for larger matrices, in which case we must use a numerical method.

In practice, eigenvalues of large matrices are not computed using the characteristic polynomial. Computing the polynomial becomes expensive in itself, and exact (symbolic) roots of a high-degree polynomial can be difficult to compute and express: the Abel–Ruffini theorem implies that the roots of high-degree (5 or above) polynomials cannot in general be expressed simply using *n*th roots. Therefore, general algorithms to find eigenvectors and eigenvalues are iterative.

Iterative numerical algorithms for approximating roots of polynomials exist, such as Newton's method, but in general, it is impractical to compute the characteristic polynomial and then apply these methods. One reason is that small round-off errors in the coefficients of the characteristic polynomial can lead to large errors in the eigenvalues and eigenvectors: the roots are an extremely ill conditioned function of the coefficients.

Numerical computation of eigenvectors

Once the eigenvalues are computed, the eigenvectors could be calculated by solving the equation

$$(\boldsymbol{A}-\lambda_i \boldsymbol{I})\, \boldsymbol{v}_{i,j}=0 \qquad \text{(V.15)}$$

using Gaussian elimination or any other method for solving matrix equations. However, in practical large-scale eigenvalue methods, the eigenvectors are usually computed in other ways, as a byproduct of the eigenvalue computation.

What is remarkable here is that for effectively applying the KLT as the tool for CS, we have to implement eigenvector removal and eigenvalue decomposition; and for doing that we have to apply iterative methods for computing them.

This is the main point here, that even KLT is also requiring the iterative methods for computations. That makes it more complex as well as more time consuming.

It has slow speed in seeking the transform from the correlation matrix constructed by given training data.The larger the scale or dimension of the correlation matrix is, the slower the speed of computing the eigenvectors and hence transform is, and then so is performing compressing or encoding transform.

The time complexity of the operations is analyzed as follows, where no distinction is made between a multiplication and an add operation:

- Form the mean vector with O(MND) element-wise operations. Calculate the set of outer products and sum, $\sum_{k=0}^{MN-1} x_k x_k^T$, in O ($MND^2$).
- Form $m_x m_x^T$; subtract matrices to find C_x ; and find the eigenvectors of C_x. The eigenvector calculation is $O(D^3)$. Convert the x_k to zero-mean form in O (MND).
- Form the y_k by $O(MND^2)$ operations.

  Here, M is the no. of rows, N is the no. of columns and D is the rank of the matrix.

  The complexity, therefore, can be considered to be $O(MND) + O(MND^2)$.

The lack of a general fast algorithm, because the covariance matrix eigenvectors must be found in every case, makes it pressing to find a suitable parallel decomposition, though some iterative algorithms also exist. It, therefore, suggests that the KLT also requires iterative methods for fast implementation. This makes it in one sense similar to that of the greedy algorithms.

Since it is data dependent and requiring the iterative methods, for the same level of complexity of work, the KLT requires more computations as shown above.

There is no unique KLT for all random processes, and it is, in general, not possible to find a fast (FFT-type) algorithm to compute the transform coefficients.

International Journal of Engineering Innovation & Research
Volume 5, Issue 5, ISSN: 2277 – 5668

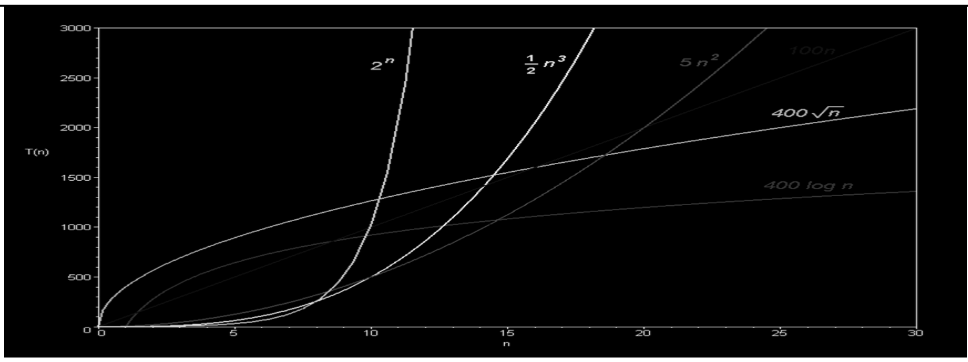


Fig. 5. The curves depicting comparative complexities of various algorithms with respect to no. of measurements

Table V: Run-times of OMP and KLT with Measurement Matrix with and without noise

Sparsity (GI) (%)	No. of Active Spectrum Components (Out of 30)	No. of Measurements, N	Time (OMP-Noiseless MM) (sec)	Time (KLT-Noiseless MM) (sec)	Time (OMP-MM with AWGN) (sec)	Time (KLT-MM with AWGN) (sec)
100	**1**	**8**	**0.0063**	**0.008115**	**0.028627**	**0.008219**
79.3103	**7**	**55**	**0.006385**	**0.014684**	**0.008195**	**0.025053**
62.069	**12**	**97**	**0.005808**	**0.022116**	**0.008413**	**0.041528**
41.3793	**18**	**145**	**0.005885**	**0.026397**	**0.00825**	**0.049071**
20.6897	**24**	**193**	**0.006466**	**0.034693**	**0.008402**	**0.076821**
0	**30**	**242**	**0.006208**	**0.042193**	**0.008253**	**0.071453**

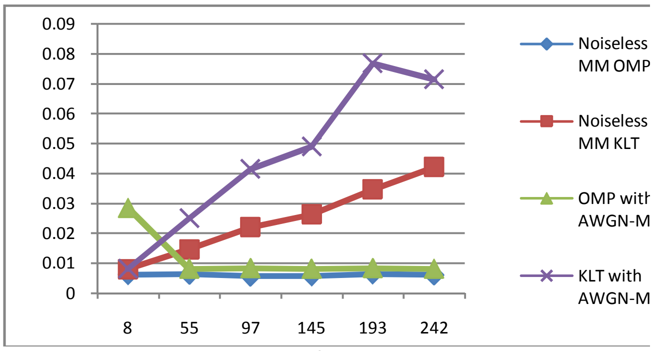


Fig. 6. The Run-times of OMP and KLT with Measurement Matrix with and without noise

The Execution Time or The Run Time of the algorithms:

The Run time for the algorithms was observed for two types of criteria:

1. The wavelet basis and level of decomposition applied for the sparsification process, and
2. The no. of measurements that were calculated based on the sparsity level.

This parameter defines the speed of the algorithm for solving the given problem. Here, we consider the sensing of the spectrum and detecting the spectral holes as our main tasks; as the device, using these algorithms for spectrum sensing will be a Cognitive Radio device and a non-licensed secondary user (SU).

In the dynamically changing environment of spectrum usage and allotments, it will be necessary to sense the spectrum and detect the opportunities for spectrum reuse, and make decisions quickly, within the least possible time duration. Hence, we consider the Run Time of the algorithm or Execution Time is one of the most important performance metrics.

Tables VI (a-b): The time required for execution of OMP (a-Left) and KLT (b-Right) with Measurement Matrix without Noise for different DWTs applied as Sparsifying Basis

Sparsity (%)	Time (sec) (db1)	Time (sec) (db2)	Time (sec) (db3)	Time (sec) (db4)
100	0.0069065	0.0067543	0.0061568	0.006599
79.31	0.0063794	0.0055374	0.0064882	0.006609
62.07	0.0060973	0.0061418	0.0071015	0.00672

Sparsity (%)	Time (sec) (db1)	Time (sec) (db2)	Time (sec) (db3)	Time (sec) (db4)
100	0.006868	0.007634	0.009491	0.0084661
79.31	0.014681	0.014447	0.015368	0.0142399
62.07	0.022149	0.019695	0.020844	0.0257737

41.38	0.006794	0.00638	0.0064275	0.007045
20.6897	**0.0066212**	**0.0067539**	**0.0063435**	**0.006659**
0	**0.0054575**	**0.0064847**	**0.0057184**	**0.006596**

41.38	0.026274	0.028314	0.023897	0.0271049
20.6897	0.030036	0.031567	0.039273	0.037895
0	**0.044272**	**0.039862**	**0.037502**	**0.047135**

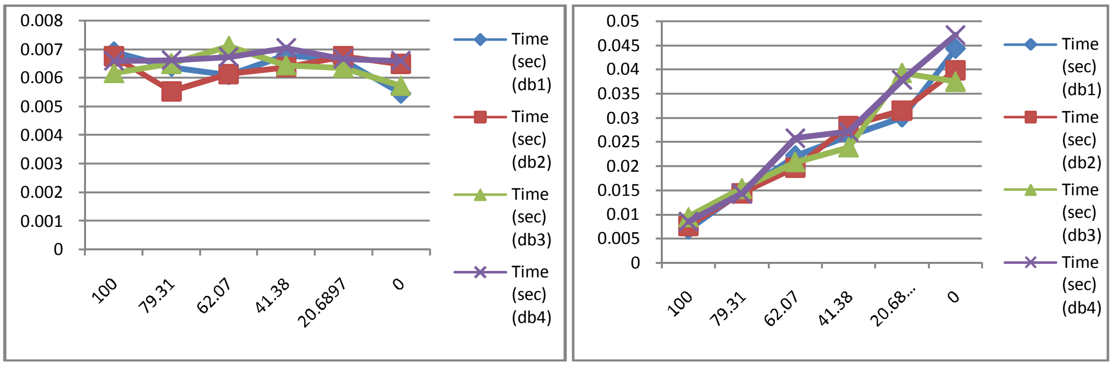
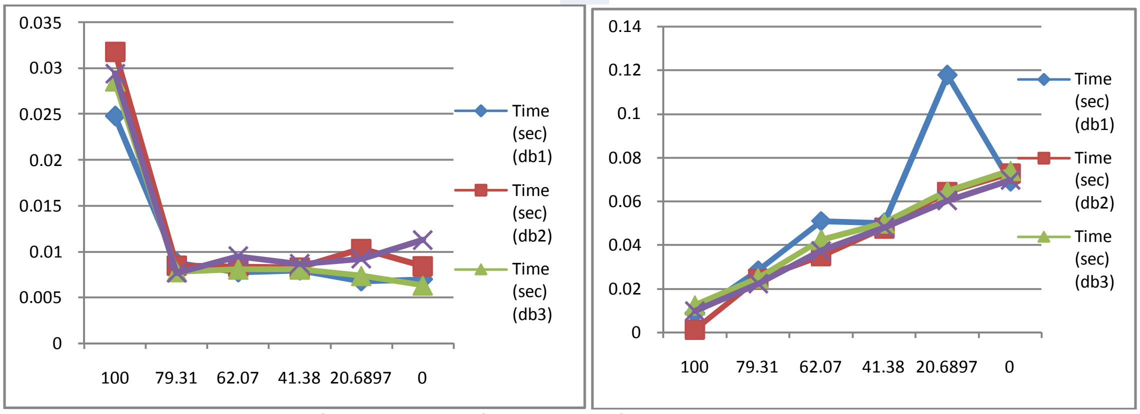


Fig. 7. (a-b): The time required for execution of OMP (a-Left) and KLT b-(Right) with Measurement Matrix without Noise for different DWTs applied as Sparsifying Basis

Tables VII (a-b): The time required for execution of OMP (a-Left) and KLT (b-Right) with Measurement Matrix with AWGN for different DWTs applied as Sparsifying Basis

Sparsity (%)	Time (sec) (db1)	Time (sec) (db2)	Time (sec) (db3)	Time (sec) (db4)
100	**0.0247835**	**0.0317876**	**0.0285245**	**0.029414**
79.31	0.0088185	0.0085029	0.0077845	0.007673
62.07	0.0077475	0.0083158	0.0080823	0.009507
41.38	0.0079527	0.008278	0.0081015	0.00867
20.6897	**0.0067748**	**0.0102749**	**0.0073684**	**0.00919**
0	**0.0069868**	**0.0084305**	**0.0063271**	**0.011266**

Sparsity (%)	Time (sec) (db1)	Time (sec) (db2)	Time (sec) (db3)	Time (sec) (db4)
100	**0.0089002**	**0.0014397**	**0.0125407**	**0.009996**
79.31	0.0284509	0.0245528	0.0247724	0.022437
62.07	0.0509852	0.0351901	0.0425076	0.037429
41.38	0.050227	0.0478296	0.0500714	0.048156
20.6897	**0.1178214**	**0.0642845**	**0.0645933**	**0.060584**
0	**0.069132**	**0.072829**	**0.07402**	**0.06983**



Figures 8(a-b): The time required for execution of OMP (a-Left) and KLT (b-Right) with Measurement Matrix with AWGN for different DWTs applied as Sparsifying Basis

a) The DWT variations, except for slight changes in execution time, that is proportional to O(KNd),show little effect on performance of OMP as a sparsity basis.
b) The execution time for the KLT shows that the Run time varies in proportion with O(MND) + O(MND^2). It varies in much wider range with Noisy Measurement Matrix compared to Noiseless case.
c) If we consider the measurement of time taken for execution, OMP is much faster in any case. The effective variations in time for OMP are quite less.

With increasing sparsity and decreasing sparsity index (Gini index), we can see that the OMP and KLT are behaving the opposite ways. The OMP has its execution time decreased, while KLT takes more time with increasing sparsity levels and decreasing Gini Index values. The time taken by KLT is also almost 10 times greater than that taken by OMP. However, it is also observable that the OMP takes rather a larger time when the signal sparsity is more and no. of active spectrum components is too small, i.e., one only.

d) The sparsification by different Daubechies wavelets is showing slighter only, effects on the required execution time for both OMP and KLT.

The Correlation between input and output or Reconstruction Accuracy of the Algorithms:

The correlation between the input and output,or, the Reconstruction Accuracy here, is used to show how much the algorithm succeeds in recovering the unknown input spectrum. This in turn can give the idea of spectrum detection ability of the algorithm.

The correlation is derived for both the algorithms using the xcorr and mscohere commands in Matlab. It is derived by taking the root, mean and square of the two outputs obtained by the two commands. The xcorr gives the correlation between the two signals (here the input and output spectrums) in terms of cross-correlation between the two random processes. On the other hand, the mscohere finds the magnitude squared estimate of the input signals x and y (here the input and output spectrums) indicating how well the two signals correspond to each other at each frequencies.

Tables VIII (a-b): The Correlation between Input and Output, or, the Reconstruction Accuracy of OMP (a-Left) and KLT (b-Right) with Measurement Matrix without Noise for different DWTs applied as Sparsifying Basis

Sparsity (%)	Correlation (db1)	Correlation (db2)	Correlation (db3)	Correlation (db4)
100	0.722362	0.722336	0.722323	0.722171
79.3103	0.70646	0.707156	0.707662	0.707408
62.069	0.702475	0.702228	0.70307	0.70272
41.3793	0.699084	0.699653	0.700259	0.699871
20.6897	0.69924	0.700298	0.700744	0.700216
0	0.6981	0.7	0.69689	0.69715

Sparsity (%)	Correlation (db1)	Correlation (db2)	Correlation (db3)	Correlation (db4)
100	0.853899	0.3592918	0.302245	0.300532
79.3103	0.5966033	0.2814665	0.187051	0.174835
62.069	0.516531	0.2713444	0.138732	0.185723
41.3793	0.5699064	0.2843684	0.172286	0.153008
20.6897	0.4827888	0.3356263	0.206208	0.142309
0	0.44698	0.33278	0.23189	0.15004

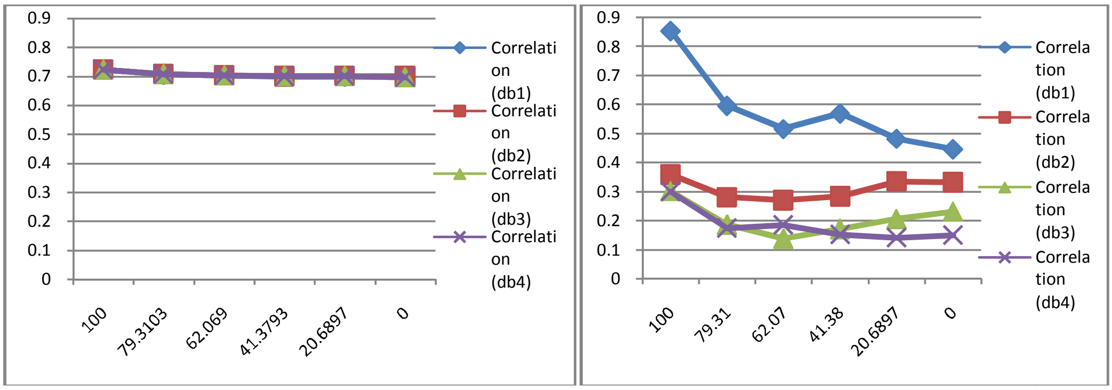


Fig. 9. (a-b): The Correlation between Input and Output, or, the Reconstruction Accuracy of OMP (a-Left) and KLT (b-Right) with Measurement Matrix without Noise for different DWTs applied as Sparsifying Basis

Tables IX (a-b): The Correlation between Input and Output, or, the Reconstruction Accuracy of OMP (a-Left) and KLT (b-Right) with Measurement Matrix with AWGN for different DWTs applied as Sparsifying Basis

Sparsity (%)	Correlation (db1)	Correlation (db2)	Correlation (db3)	Correlation (db4)
100	**0.706595**	**0.702676**	**0.706798**	**0.705193**
79.3103	0.7065933	0.7073629	0.70792	0.707329
62.069	0.702476	0.7022055	0.702307	0.70256
41.3793	0.6991179	0.6956571	0.700192	0.699835
20.6897	0.699235	0.7002688	0.700821	0.700201
0	0.6981	0.7	0.6969	0.69715

Sparsity (%)	Correlation (db1)	Correlation (db2)	Correlation (db3)	Correlation (db4)
100	**0.8526277**	**0.3815782**	**0.302953**	**0.294874**
79.3103	0.6028192	0.2807598	0.187581	0.177451
62.069	0.5173895	0.2729902	0.1392	0.185659
41.3793	0.5693043	0.2850822	0.172994	0.153233
20.6897	0.48321	0.3356888	0.206933	0.141704
0	0.44698	0.33276	0.23188	0.15005

International Journal of Engineering Innovation & Research
Volume 5, Issue 5, ISSN: 2277 – 5668

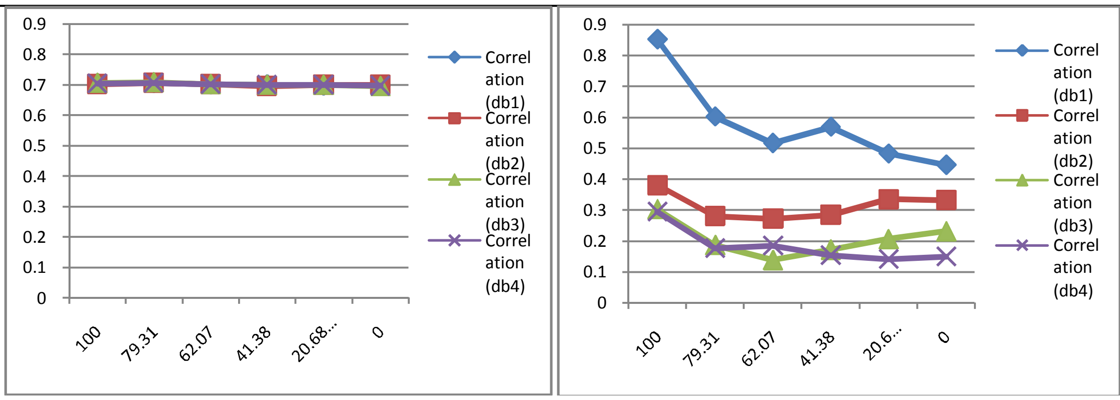


Fig. 10 (a-b): The Correlation between Input and Output, or, the Reconstruction Accuracy of OMP (a-Left) and KLT (b-Right) with Measurement Matrix with AWGN for different DWTs applied as Sparsifying Basis

1. We can see that for both Noiseless and Noisy Measurement Matrix, OMP exhibits Very slight change in Correlation. It shows the variation in this parameter with increasing sparsity level in very small range and exhibits the tendency for decreasing Correlation with increasing sparsity. The value remains within the proximity of 0.70 most of the time with OMP.
2. While, very wide range of change is observed in Correlation with KLT. It shows the variation in this parameter with increasing sparsity level significantly and exhibits the tendency for decreasing Correlation with increasing sparsity, from 0.85 for the lowest no. of active spectral components present, i.e., one, to the highest sparsity, around 0.15 for the highest no. of active spectral components present.
3. In addition, with Noiseless and Noisy Measurement Matrix application KLT shows a notable change in correlation.
4. The DWTs are having prominent effects on performance of KLT algorithm as sparsifying basis, hence indicating the data dependent nature of KLT. The OMP on the other hand, exhibits very small variations in correlation values with different DWT basis applications.

The Error (%):

Here the error represents the dissimilarities between the input and output spectrums of the algorithms. It shows that how much these algorithms fall short of giving out the accurate spectrum reconstruction, and, consequently, recovery. It is expressed in percentage.

It will be natural to observe the variation pattern of this parameter be opposite to that of the Correlation between the input and output, or, the Reconstruction Accuracy.

Tables X(a-b): The Error, or, Dissimilarities between Inputs and Outputs of OMP (a-Left) and KLT (b-Right) with Measurement Matrix without Noise for different DWTs applied as Sparsifying Basis

Sparsity (%)	**Error (%) (db1)**	**Error (%) (db2)**	**Error (%) (db3)**	**Error (%) (db4)**
100	27.76378	27.76637	27.76767	27.7829
79.3103	29.35396	29.28438	29.23379	29.25925
62.069	29.75255	29.7772	29.69305	29.72805
41.3793	30.09164	30.03471	29.97407	30.14286
20.6897	30.076	29.97025	29.92563	29.97838
0	30.19	30	30.311	30.285

Sparsity (%)	**Error (%) (db1)**	**Error (%) (db2)**	**Error (%) (db3)**	**Error (%) (db4)**
100	14.6101	64.070817	69.77552	69.94675
79.3103	40.339667	71.853354	81.29492	82.5162
62.069	48.3469	72.86556	86.12683	81.42768
41.3793	43.009357	71.563157	82.77138	84.69921
20.6897	51.721125	66.437325	79.37925	85.76914
0	55.302	66.722	76.811	84.996

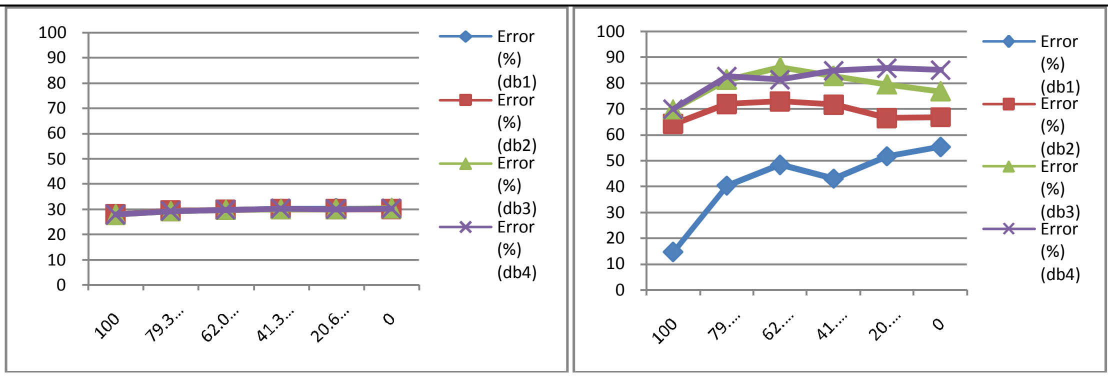


Fig. 11 (a-b): The Error, or, Dissimilarities between Inputs and Outputs of OMP (a-Left) and KLT (b-Right) with Measurement Matrix without Noise for different DWTs applied as Sparsifying Basis

Tables XI (a-b): The Error, or, Dissimilarities between Inputs and Outputs of OMP (a-Left) and KLT (b-Right) with Measurement Matrix with AWGN for different DWTs applied as Sparsifying Basis

Sparsity (%)	**Error (%) (db1)**	**Error (%) (db2)**	**Error (%) (db3)**	**Error (%) (db4)**
100	29.3405	29.7324	29.32017	29.4807
79.3103	29.340667	29.263708	29.20796	29.26708
62.069	29.7524	29.77945	29.7693	29.74405
41.3793	30.088214	30.043429	29.98079	30.0165
20.6897	30.0765	29.973125	29.91788	29.97988
0	30.19	30	30.31	30.285

Sparsity (%)	**Error (%) (db1)**	**Error (%) (db2)**	**Error (%) (db3)**	**Error (%) (db4)**
100	14.737233	61.84218	69.70468	70.51261
79.3103	39.718083	71.924021	81.2419	82.25492
62.069	48.26105	72.700985	86.08	81.43412
41.3793	43.069571	71.491779	82.70065	84.67667
20.6897	51.679	66.431125	79.30675	85.82961
0	55.302	66.724	76.812	84.995

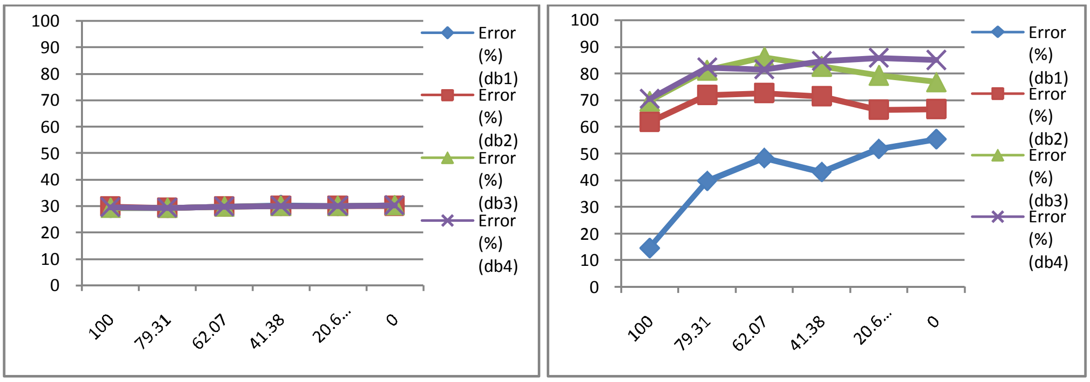


Fig. 12 (a-b): The Error or, Dissimilarities between Inputs and Outputs of OMP (a-Left) and KLT (b-Right) with Measurement Matrix with AWGN for different DWTs applied as Sparsifying Basis

For various DWTs applied as sparsifying basis, the error in the output shows, that how much the algorithms do fail to recover the input spectrum exactly.

a) Very slight change in Error is observed for OMP, as compared to the KLT. It shows the variation in Error with increasing sparsity level lies in very small range and exhibits the tendency increasing with increasing sparsity, for OMP, i.e., 29.0 % to 30.2 %.
b) While KLT, has a wide variation in Error also, as we had observed in Correlation. However, the pattern shows the opposite or increasing direction of variations with increasing Sparsity Levels, or, decreasing Sparsity Index (Gini Index). It varies widely between 14.61 % to 84.995 % for different sparsity levels for various scenarios with KLT.
c) The OMP does not tend to have much effect of DWT basis on error. While the KLT shows the effect of them on its performance in terms of the Error.

The Probability of Missed Detection (P_M) and The Probability of False Alarm (P_F):

The Probability of Missed Detection (P_M) is actually the chance of missing the detection of any existing spectrum component actively present. While the Probability of False Alarm (P_{FA}) is the chance where in the sensing process the CR will get the detection of an active spectrum component even if the said component is not active.

In simplest form, spectrum sensing of a single channel is a binary hypothesis testing problem. Specifically,

$$H_0: y[n] = w[n], n = 1, N \qquad \text{(V.16)}$$

$$H_1: y[n] = x[n] + w[n], n = 1, . . . , N. \qquad \text{(V.17)}$$

Where, x[n] represents a primary user's signal, w[n] is noise and n represents time. The received signal y[n] is vector, of length L; and n is the sample index.

International Journal of Engineering Innovation & Research
Volume 5, Issue 5, ISSN: 2277 – 5668

For simplicity, let 0 and 1 denote the two hypotheses, let the random variable H denote the state of the signal, and let the random variable $\widehat{H}$ denote the sensing decision. Thus, the probability of missed detection and the probability of false alarm are defined as,

$$P_M \triangleq P_r\{\widehat{H} = 0 | H = 1\} \quad (V.18)$$

$$P_F \triangleq P_r\{\widehat{H} = 1 | H = 0\}. \quad (V.19)$$

Small $P_F$is necessary in order to provide possible high throughput in dynamic spectrum access networks, since a false alarm wastes a spectrum opportunity. On the other hand, small $P_M$is necessary in order to limit the interference to PUs. A detection algorithm can seek tradeoffs between P_M and $P_F$by varying the detection threshold. [44]

Other performance metrics proposed in the literature include, but are not limited to, the detection probability of wideband. It is defined as the probability that all ON channels are correctly detected, and the false alarm probability of wideband, which is defined as the probability that any of the OFF channels are falsely detected as ON, the (empirical) probability of detecting a given number of ON channels and so forth. [44]

During the simulations, which we carried out for both basic OMP and KLT, the input and output spectrums were matched. With help of the level of matching between the two, different probabilities related to the detecting performance of the algorithms were calculated. The more the matching, the better the detection we found. Therefore giving a higher probability of detection.

For finding out these probabilities, let us make a simple consideration. Suppose, we have these events being observed during a spectrum sensing/ detection process:

A = {a Signal is ON} (V.20)

B= {a Signal is Detected} (V.21)

Therefore, the complementary events will be,

A^c = {a Signal is OFF} (V.22)

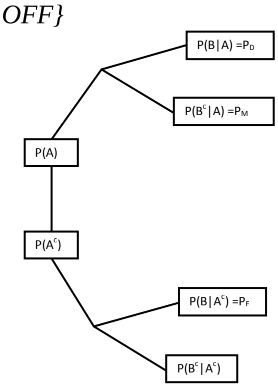


Fig. 13. Binary Hypothesis Representation for Probabilities of Detection, Missed Detection and False Alarm

B^c= {a Signal is NOT Detected} (V.23)

Here,

- P_D = P(B|A)=Probability of Detection = Probability of Signal being Detected when Signal is ON.
- P_M = P(B^c|A) = Probability of Missed Detection = Probability of Signal NOT being Detected when Signal is ON.
- P_F = (P|A^c) = Probability of False Alarm = Probability of Signal being Detected even when the Signal is OFF.
- P(B^c|A^c) = Probability of Signal NOT being Detected when the Signal is OFF.

Therefore, we get:

$$P_D = P(A \cap B) = P(A)P(B|A), \quad (V.24)$$

$$P_F = P(A^c \cap B) = P(A^c)P(B|A^c), \text{ and,} \quad (V.25)$$

$$P_M = P(B^c \cap A) = P(A)P(B^c|A). \quad (V.26)$$

Based on these assumptions, the Probability of False alarm, P_F, and Probability of Missed detection, P_M, were calculated for both the algorithms.

For getting the idea of behavior of a compressive detector and probabilities, we first discuss a classical detector; and then describe how a compressed detector can be derived using the same approach. As already, we have assumed that there are two hypotheses concerning the signal; that it is present in the measurements or it is not. The classical Neymon-Pearson (NP) detector involves a likelihood ratio test where the sufficient statistics t ≡‹y, x› is compared against a threshold γ. Here y are the measurements, x is the signal of interest and γ is set to achieve certain probability of false alarm rate $P_F \leq \alpha$ for some $0 \geq \alpha \leq 1$. It is easy to show that

$$P_D(\alpha) \approx Q(Q^{-1}(\alpha) - \sqrt{SNR}), \quad (V.27)$$

where Q(·) is the flipped version of standard Gaussian cumulative distribution function. [45]

This theory can be easily extended to the case when the measurements are made using a compressed sampler in a compressive detector. Considering the binary hypothesis given above, having M measurements taken by the compressive sampler, we can find, [45]

$$P_D(\alpha) \approx Q(Q^{-1}(\alpha) - \sqrt{(M/N)}\sqrt{SNR}). \quad (V.28)$$

The Probability of Detection can give us the Probability of Missed Detection, P_M. The Probability of False Alarm, P_{FA}, can be set to determine the required threshold of SNR for that specific P_{FA}. Or else, It was calculated for different SNR specifications using the following commands from MATLAB.

```
noisepow = 1.38e-23*293*db2pow(1)*33e6;
Ntrial=1000;
snrthreshold=5;
noise =
sqrt(noisepow/2)*(randn(1000,1)+1j*randn(1000,1));
threshold=sqrt(noisepow*db2pow(snrthreshold));
```

We can also find the snr threshold setting the P_F values heuristically and take the help from the following MATLAB command,

```
snr thresh=npwgnthresh(pfa).
```

It calculates the SNR threshold in decibels for detecting a deterministic signal in white Gaussian noise. The detection uses the Neyman-Pearson decision rule to determine the specified Probability of False Alarm, P_{FA}. This function of Matlab uses a square-law detector.

Based upon above concepts, we have considered various cases for different values of P_{FA} for obtaining the corresponding Probabilities of Detection. From this it is easy to obtain the specific corresponding Probability of Missed Detection, P_M.

Table XII: Theoretical values of Probability of Detection and Probability of Missed Detection based on specified values of Probability of False Alarm

Input Sparsity (%)	**PD (Case 1)**	**PD (Case 2)**	**PD (Case 3)**	**PM (Case 1)**	**PM (Case 2)**	**PM (Case 3)**
100	0.4718	0.2893	0.2688	0.5282	0.7107	0.7312
79.3103	0.8809	0.4636	0.4524	0.1191	0.5364	0.5476
62.069	0.9673	0.5614	0.5572	0.0327	0.4386	0.4428
41.3793	0.9927	0.6467	0.6485	0.0073	0.3533	0.3515
20.6897	0.9984	0.7135	0.7197	0.0016	0.2865	0.2803
0	**0.9997**	**0.7679**	**0.777**	**0.0003**	**0.2321**	**0.223**

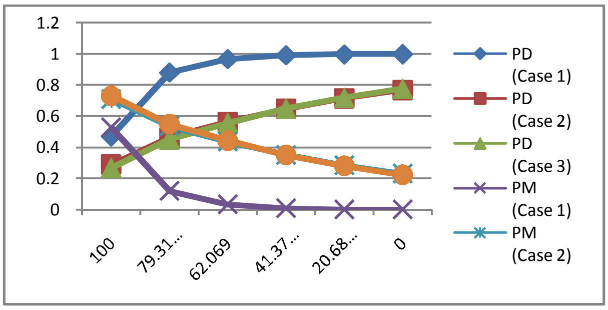


Fig. 14. Variations obtained theoretically for Probability of Detection, P_D, and, Probability of Missed Detection, P_M, for different Sparsity Levels for Compressive Sensing Algorithms, defined for corresponding no. of measurements

The various values taken for calculating the aforesaid probabilities corresponding to the parameters used for simulation of sensing algorithms OMP and KLT:

y=PD=qfunc((qfuncinv(α))-(sqrt(M/N)*sqrt(SNR)))

$PD(\alpha) \approx Q(Q^{-1}(\alpha) - \sqrt{(M/N)}\sqrt{SNR})$

α=PFA (predefined heuristically for obtaining the P_D) N=202, M=No. of Measurements (M=based on formula given here)

N= Length of signal

And, different considerations for calculating the probabilities theoretically:

Case 1: PFA=0.2, and , SNR = 15 dB

Case2: PFA=0.2 and for this,**snrthresh=npwgnthresh(0.2)**; and y=qfunc((qfuncinv(0.2))-(sqrt(8/202)*sqrt(2.0667)));

Case 3: PFA=0.1781and for this, **snrthresh=npwgnthresh(0.1781)**; and y=qfunc((qfuncinv(0.1781))-(sqrt(8/202)*sqrt(2.3689)));

The results we obtained with simulation provided us with P_D, P_M, and, P_F values as per the probability theory considerations discussed above.

Table XIII: The Probability of False Alarm, Probability of Missed Detection, and, Probability of Detection for OMP and KLT with Noiseless Measurement Matrix and AWGN-Measurement Matrix for various sparsity levels according to the simulation results

Input Signal Sparsity (%)	$P_{F(OMP)}$	$P_{F(KLT)}$	$P_{M(OMP)}$	$P_{M(KLT)}$	$P_{D(OMP)}$	$P_{D(KLT)}$
100	0.01	0.03	0	0	1	1
79.31	0.08	0.26	0	0	1	1
62.07	0.1	0.46	0	0	1	1
41.38	0.14	0.49	0	0.03	1	0.97
20.6897	0.23	0.5	0	0	1	1

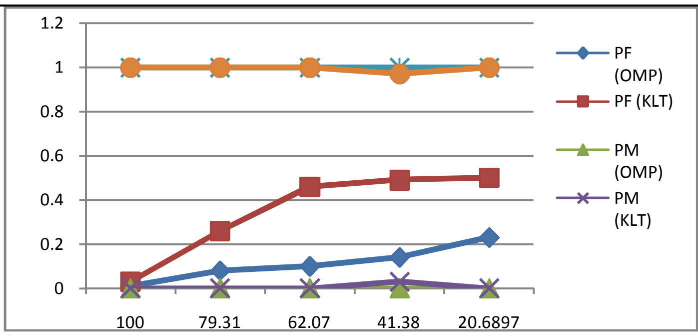


Fig. 15. The Probability of False Alarm, Probability of Missed Detection, and, Probability of Detection for OMP and KLT with Noiseless Measurement Matrix and AWGN-Measurement Matrix for various sparsity levels according to the simulation results

The Probability of False Alarm, P_F, is the one, considered here, as the probability for detecting an OFF channel as ON. Both the OMP and KLT show similar behavior with the noiseless measurement matrix and AWGN-added measurement matrix.

In addition, it is observed that this probability is minimum or near to zero at the highest sparsity level in Gini-index, means when only a single signal component is active. We have observed that the algorithms detect only one spectral components that is inactive out of total 29 inactive spectral components for a specific range of 60 MHz spectrum. The same way, it is minimum when the entire spectrum is occupied with active spectral components; as there are all the components in ON state, there is not a single inactive component. Then with increasing no. of active spectral components and decreasing Gini-index values; along with the increasing no. of measurements, the OMP gives slight increase in P_F, while KLT shows considerable increase in it. The OMP and KLT both have its highest value at 20.69 % of sparsity level in terms of Gini-index. At different sparsity levels of the input, the OMP detected two frequency components that were inactive actually. While KLT detected and gave out as output six to twelve frequency components that were inactive at different sparsity levels. This is reflected in the graph also depicting the PFA of the algorithms.

The Probability of Missed Detection, P_M, is on the other hand, the chance of detecting an ON channel as OFF. The OMP, gives a performance, which makes this probability, zero, for both noiseless Measurement matrix and AWGN-added measurement matrix cases. There is no active component of the said, or, concerned spectrum, which is not detected by the OMP. The KLT also performs well in the matter of detection. It has only a small P_M for the sparsity level of 41.38 % in Gini-index up to about 0.03. With all other sparsity levels, ranging from the highest to the lowest, except from this one, it also has P_M equaling zero. Correspondingly, the Probability for Detection P_D, is found to be very good, almost around one for both the algorithms.

The Implications:

In this work, simulation time and correlation which are acquired from around a total of 1000 trials for typical signals with range of 60 MHz having 30 spectrum components of 2 MHz bandwidth each. All simulations are performed on a laptop with an Intel(R) core(TM) i5 2.40 GHz processor and the CPU time was calculated by the Matlab (version 2011b). Tables 2, 3, and 4 illustrate the simulation outputs for both OMP and KLT algorithms.

Our results indicate here that for OMP, the correlation behavior can be defined as this,

$$r \rightarrow 1, \text{ as } K \rightarrow 0. \qquad (V.29)$$

Where, K is the sparsity order. On the other hand, KLT has different behavior, as,

$$r \rightarrow 0, \text{ as } K \rightarrow 0. \qquad (V.30)$$

It is also observed here, that with increasing no. of measurements, M in OMP, the correlation, r, tends to increase and simulation time seems to increase slightly – not very remarkably. For the KLT, with increasing no. of measurements, M, the correlation, r, tends to decrease and simulation time increases significantly.

The algorithms we have discussed here are applied for compressive sensing. To enable efficient spectrum usage, cognitive radio devices are being considered to be used. Moreover, for better spectrum reuse, a wider range of it is to be sensed. We know that for wide band sensing the compressive sensing is a promising technique on the horizon.

The OMP has the measurement matrix of size Nxd for taking sparse measurements and hence for data compression. While on the other hand, the OMP and other greedy algorithms are simple by implementation ways and even if they required iterative methods, due to simplicity in their basic nature they are preferable candidates to work upon for CS applications.

This motivated us to focus on OMP, because it can be instrumental for the CS as it is simpler in application despite being a greedy iterative method. The results above show that the performance of the OMP is improving with the increasing density of the spectrum, in terms of spectral reconstruction or in terms of time consumed, compared with that for KLT; that gives a hopeful vision for the CS based systems as the no. of secondary users is going to increase in the future definitely.

It is apparent that the algorithms we discussed above are the enabling tools for efficient spectrum sensing for the CR. Improvisation of these may help development of a sophisticated spectrum sensing CR that will lead towards the better spectrum resource sharing and utilization.

VII. CONCLUSION AND FUTURE WORK

The OMP and KLT both are working quite efficiently for recovering the spectrum for compressive spectrum sensing. However, the implications of the results obtained are that the KLT is much data dependent and the computational time of the algorithm is high for eigenvector decompositions.

The OMP is also an efficient and impressive algorithm, which is a fast greedy algorithm that iteratively builds up a signal representation by selecting the atom that maximally improves the representation at each iteration. The OMP is easily implemented and it converges quickly. That makes it an attractive choice to work on. To improvise the signal recovery performance of the OMP, we would like to work on it in future with a varying sparsity environment.

The results obtained above show that the OMP can be successfully applicable for sparsity robust environments for compressive spectrum sensing and detection.

REFERENCES

[1] J. Mitola and G. Maguire, "Cognitive radio: making software radios more personal," Personal Communications, IEEE, vol. 6, no. 4, pp. 13–18, 1999.

[2] Patel, C A, and, Vithalani, C H, "Compressive Spectrum Sensing: An Overview", IJIREC, Volume 1, Issue 6, September 2014, PP 1-10 ISSN 2349-4042 (Print) & ISSN 2349-4050 (Online)

[3] E. J. Candes, "Compressive sampling," in Proceedings of the International Congress of Mathematicians, (Madrid, Spain), 2006.

[4] M. A. Davenport, M. B. Wakin, and R. G. Baraniuk, "Detection and estimation with compressive measurements," technical, Rice University, 2007.

[5] E. Candes, J. Romberg and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information", IEEE Transactions on Information Theory, 52(2) pp. 489 - 509, Feb. 2006.

[6] E. Candes and T .Tao, "Near Optimal Signal Recovery From Random Projections: Universal Encoding Strategies?", IEEE Trans. on Information Theory, 52(12) pp. 5406 - 5425, Dec. 2006.

[7] D. Donoho, "Compressed Sensing", IEEE Transactions on Information Theory, 52(4), pp. 1289 - 1306, April 2006

[8] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," SIAM review, vol. 43, no. 1, pp. 129–159, 2001.

[9] Gwon, Y., Kung, H. T., and, Vlah, D., "Compressive Sensing with Optimal Sparsifying Basis and Applications in Spectrum Sensing", Globecom, 2012

[10] Tropp, J A, and, Gilbert, A C, "Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit", IEEE Transactions, 2005.

[11] A.K. Jain, "Fundamentals of digital image processing," Chapters 2, 5 and 11, Englewood Cliffs, NJ, Prentice Hall, 1989.

[12] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. IEEE Trans. Signal Process, 41(12):33973415, 1993.

[13] Joel A. Tropp and Deanna Needell. Cosamp: Iterative signal recovery from incomplete and inaccurate samples. CoRR, abs/0803.2392, 2008.

[14] Wei Dai and OlgicaMilenkovic. Subspace pursuit for compressive sensing signal reconstruction. IEEE Trans. Inf. Theor., 55(5):2230–2249, May 2009.

[15] Thomas Blumensath and Mike E. Davies. Iterative hard thresholding for compressed sensing. CoRR, abs/0805.0510, 2008

[16] E. Candès and T. Tao, "Decoding by linear programming," IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[17] Davenport, M A, and, Wakin,M B, "Analysis of Orthogonal Matching Pursuit Using the Restricted Isometry Property", IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 56, NO. 9, SEPTEMBER 2010

[18] Xiao, Q, Chen, L, and, Wang, Y, "An Efficient Dimension Reduction Quantization Scheme for Speech Vocal Parameters", IJITCS, 2011, 1, 18-25, MECS Press.

[19] Gwon, Y., Kung, H. T., and, Vlah, D., "Compressive Sensing with Directly Recoverable Optimal Basis and Applications in Spectrum Sensing", TR-08-11, Computer Science Group Harvard University Cambridge, Massachusetts.

[20] Fleury,M., Self, R.P., and, Downton, A. C., "Development of a Fine-grained Parallel Karhunen-Loève Transform", April 22-26, 2003, Parallel and Distributed Processing Symposium, 2003. Proceedings. International

[21] Axell, E., Leus,G., Larsson, E. G., and Vincent Poor, H., "State-of-the-art and recent advances Spectrum Sensing for Cognitive Radio: State-of-the-art and recent advances", 2012, IEEE signal processing magazine (Print), (29), 3, 101-116.

[22] Lakshminarayana. M., and, Sarvagya, M., "Scaling the Effectiveness of Existing Compressive Sensing in Multimedia Contents", International Journal of Computer Applications (0975 – 8887), Volume 115 – No. 9, April 2015.

[23] C. R. Berger, Z. Wang, J. Huang, and S. Zhou, "Application of Compressive Sensing to Sparse Channel Estimation" IEEE Communication Magazine, pp.164-174, 2010.

[24] A. Gilbert, P. Indyk, "Sparse Recovery Using Sparse Matrices", Proceedings of IEEE, pp.937-947, Vol.98, Iss.6, 2010.

[25] L.C. Potter, E. Ertin, J. T. Parker, M. Cetin, "Sparsity and Compressed Sensing in Radar Imaging", Proceedings of the IEEE, Vol. 98, No. 6, June 2010.

[26] J.A. Tropp, S. J. Wright, "Computational Methods for Sparse Solution of Linear Inverse Problems", Proceedings of the IEEE, Vol. 98, No. 6, June 2010.

[27] V. M. Patel, R. Chellappa, "Sparse Representations, Compressive Sensing and Dictionaries for Pattern Recognition", IEEE-First Asian Conference on Pattern Recognition, pp.325-329, 2011.

[28] G. Wang, "Compressive Sensing for Biomedical Imaging", IEEE Transactions on Medical Imaging, vol. 30, no. 5, May 2011.

[29] U. Dias, M. Rane, S. R. Bandewar, "Survey of Compressive Sensing", International Journal of Scientific & Engineering Research, Vol.3, Iss.2, February-2012.

[30] A. Mammeri, B. Hadjou, and A. Khoumsi, "A Survey of Image Compression Algorithms for Visual Sensor Networks", International Scholarly Research Network, Article ID 760320, 19 pages, 2012.

[31] K. Hayashi, M. Nagahara, T. Tanaka, "A user's guide to Compressed Sensing for Communication Systems", IEICE Transactions of Communications, Vol.96, No.3, 2013.

[32] J. Kaur, K. Kaur, M. Bharti, P. Sharma and J. Kaur, "Reconstruction Using Compressive Sensing: A Review", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Iss.9, September 2013.

[33] J. Ender, "A Brief Review of Compressive Sensing Applied to Radar", 14th International radar Symposium, 2013.

[34] S. Pudlewski and T. Melodia, "A Tutorial on Encoding and Wireless Transmission of Compressively Sampled Videos", IEEE Communications Surveys & Tutorials, Vol. 15, No. 2, Second Quarter 2013.

[35] S. Qaisar, R. M. Bilal, W. Iqbal, M. Naureen and S. Lee, "Compressive Sensing: From Theory to Applications, A Survey", IEEE-Journal of Communication and Network, vol.15, Iss.5, pp.443-456, 2013.

[36] R. Subban, S. Guria, P.Pasupathi, S.Muthukumar, "Real-time Compressive Tracking - A Study and Review", International Journal of Emerging Technologies in Computational and Applied Sciences, 2014.

[37] Q. Zhou and L. Zhou, "Compressive Sensing for Video Coding: A Brief Overview", IEEE COMSOC MMTC E-Letter, Vol.9, No.2, March 2014.

[38] A. Ali, "Localization through compressive sensing: A survey", International Journal of Wireless Communications and Mobile Computing, 2015.

[39] Fletcher, A. K., and Rangans S., "Orthogonal Matching Pursuit fromNoisy Measurements: A New Analysis", Advances in Neural Information Processing Systems 22 (NIPS 2009).

[40] J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit: The Gaussian case. Appl. Comput.Math. 2007-01, California Inst. of Tech.,August 2007.

[41] Roy, A.,B., Dey,D., Benarjee, D., Mohanti, B., "Comparison of FFT, DCT, DWT, WHT Compression Techniques on Electrocardiogram & Photo plethysmography Signals", Special Issue of International Journal of Computer Applications (0975 – 8887), International Conference on Computing, Communication and Sensor Network (CCSN) 2012.

[42] Graps, A., "An Introduction to Wavelets", Andreas de Vries, "Wavelets", Version: September 4, 2006.

[43] Sun, Z., "PERFORMANCE METRICS, SAMPLING SCHEMES, AND DETECTION ALGORITHMS FOR WIDEBAND SPECTRUM SENSING", A Doctor of Philosophy Dissertation, Graduate Program in Electrical Engineering Notre Dame, Indiana December 2013.

[44] Khan, M.E., "Compressed Sensing, Compressed Classification and Joint Signal Recovery".

AUTHORS' PROFILES

Charushila Axay Patel did complete her graduation in Ele. & Comm. engineering in 1998, from Bhavnagar University. She worked as a Teaching Faculty in various institutes in Gujarat State, India; and, currently she is working as a Sr. Lecturer at BVPIT (DS), GTU, at Surat, Gujarat. She did obtain her ME degree in Microwave Engg from BIT, Mesra in 2010. She is currently pursuing her PhD at GTU, Gujarat.

Dr. C. H. Vithalani is a prominent researcher in field of Wavelet Transforms. He obtained his PhD from Gujarat University in 2006. He has a wide range of working experience in the field of Electronics & Communication engineering in industries as well as academia. He has guided over 200 students at Graduate level and over 20 students at Post Graduate level. He has presented his research papers at various National and International level conferences. Currently he is working as Department Head of Ele. & Comm. Engg Dept, GEC, Rajkot, Gujarat, India.

Dr. Preetida Jani has a broad range of experience. She obtained her degree of B.E. (Electronics & Communication) from Gujarat University, Ahmadabad, India, M.Eng.Sc (Digital Communication) from University of New South Wales, Sydney, Australia, and, PhD (Electrical Engineering) from University of Technology, Sydney, Australia.

She has worked:

1. As a Senior Research Engineer in Industry Nokia Research Center (Finland) for 10 years
2. As Professor in Sardar Vallabhbhai National Institute of Technology, Surat(For 7 months)
3. As Associate Professor in Dhirubhai Ambani Institute of Information and Communication Technology Gandhinagar, Gujarat, India.(For 2 Years)

   Currently she is working as Professor in Department of Electronics & Telecommunication Engineering at SVIT, Mumbai University. Her area of Research Interest ranges from Wireless Communication to Algorithms, and, Computer Networking to Cognitive Radio.