Matrix Padding Method for Sparse Signal Reconstruction

Sabna N *sabnan@yahoo.com*
Department of Electronics,
Cochin University of Science and Technology,
Cochin 682 022, India

P. R. Saseendran Pillai *prspillai@cusat.ac.in*
Department of Electronics,
Cochin University of Science and Technology,
Cochin 682 022, India

Abstract

Compressive sensing has been evolved as a very useful technique for sparse reconstruction of signals that are sampled at sub-Nyquist rates. Compressive sensing helps to reconstruct the signals from few linear projections of the sparse signal. This paper presents a technique for the sparse signal reconstruction by padding the compression matrix for solving the underdetermined system of simultaneous linear equations, followed by an iterative least mean square approximation. The performance of this method has been compared with the widely used compressive sensing recovery algorithms such as l1_ls, ℓ_1-magic, YALL1, Orthogonal Matching Pursuit, Compressive Sampling Matching Pursuit, etc.. The sounds generated by 3-blade engine, music, speech, etc. have been used to validate and compare the performance of the proposed technique with the other existing compressive sensing algorithms in ideal and noisy environments. The proposed technique is found to have outperformed the l1_ls, ℓ_1-magic, YALL1, OMP, CoSaMP, etc. as elucidated in the results.

Keywords: Compressive Sensing, Greedy Algorithms, LMS Approximation, Relaxation Methods, Sparse Recovery, Sub-Nyquist Rate.

1. INTRODUCTION

Compressive Sensing [1]-[3] is a new paradigm that gained the attention of researchers in signal processing, communication as well as mathematics. It helps in reconstructing the signal from far less samples than that required by the sampling theorem, which paves the way for saving memory and low data rate requirements in communication applications.

Traditional methods make use of signal representations conforming to the sampling theorem that makes compression a necessity before storage or transmission in situations where the memory space and bandwidth are scarce resources. A signal with only a few non zero coefficients in any transform domain is called a sparse signal and a signal which can be approximated by a few non zero coefficients in any transform domain is called a compressible signal. For sparse or compressible signals, the compressive sensing technology is a paradigm shift.

The recovery of the signal is carried out by using certain optimization techniques. The recovery becomes more difficult when the signal to be compressed is corrupted with noise. For resorting to compressive sensing, it is required that the compression and reconstruction techniques should be capable of transforming the data into a suitable representation domain. Many natural and manmade signals have underlying sparse representations in some basis functions [4]. Basis functions like Discrete Fourier Transform, Discrete Cosine Transform, Wavelets, etc. can be used, depending on the information and type of the signal.

There are many compressive sensing solvers like ℓ_1-magic, l1_ls, YALL1, etc., which come under the category of relaxation methods and Orthogonal Matching Pursuit (OMP), Compressive Sampling Matching Pursuit (CoSaMP), etc., which come under the category of greedy methods for sparse signal recovery. A study of compressive sensing recovery algorithms has been performed in [5] and concluded that although the relaxation algorithms have high computational complexity and slow response time, they work better in terms of relative error between the original and reconstructed signals. However, the authors do not seem to have made any effort to study the noise immunization capability of the algorithms.

Greedy algorithms are faster and have simpler implementation. For many applications OMP does not offer adequate performance, which led to the introduction of CoSaMP. It is faster and more effective for compressive sensing problems but is usually less efficient than algorithms based on convex optimization or relaxation methods. Hence, a method which offers an acceptable signal reconstruction and fast response has been proposed in this paper. Moreover, the robustness of various compressive sensing recovery algorithms are compared with the proposed technique under noisy and ideal environments.

The paper is organized as follows. Section 2 of the paper discusses the fundamental concepts in compressive sensing while section 3 discusses some of the most widely used compressive sensing algorithms. Section 4 gives a method for sparse signal reconstruction based on matrix padding and the iterative least mean squares approximation. Section 5 makes a comparison of the performances of the algorithms by estimating the signal-to-noise ratio, correlation and mean squared error. Section 6 throws light on some of the prospective applications of the proposed technique.

2. BACKGROUND

A discrete time signal $x(n)$ with N elements, can be viewed as an $N \times 1$ vector with $n=1, 2, \ldots N$. Consider a basis function ψ, which provides K sparse representations (i.e., $||\mathbf{x}||_0 \leq K$) of $x(n)$, with $K < N$. $x(n)$ can be represented in the matrix form as $\mathbf{x=\psi f}$, where ψ is the basis matrix of order $N \times N$ and $\mathbf{f}$ is the weighting coefficient vector of order $N \times 1$. The vector $\mathbf{y=\phi x}$, where ϕ is the measurement matrix of order $M \times N$ with $M < N$, is the linear projection of the signal $x(n)$. Recovering the original signal $\mathbf{x}$ requires solving an underdetermined system of simultaneous linear equations. Given the knowledge that $\mathbf{x}$ is sparse, the system regenerates the actual signal from the acquired small number of non-adaptive linear projections of the signal.

2.1 Data Recovery

To recover the sparse signal, the condition that should be satisfied is

$$\text{minimize } ||\mathbf{x}||_0 \text{ subject to } \mathbf{y=\phi x} \tag{1}$$

where $||\mathbf{x}||_0$ is the number of non-zero elements of $\mathbf{x}$, which is also called ℓ_0 norm. Computing ℓ_0 norm is an NP-hard problem [6] which led to making use of the basis pursuit relaxation or convex optimization [7]. Such approaches have led to the establishment of an ℓ_1 - ℓ_0 equivalence [8] and hence, (1) can be represented as,

$$\text{minimize } ||\mathbf{x}||_1 \text{ subject to } \mathbf{y=\phi x} \tag{2}$$

where $||\mathbf{x}||_1$ is the sum of the absolute values of the elements in $\mathbf{x}$, which is also being referred to as the ℓ_1-norm.

In the case of signals, which are contaminated with noise, the equality constraint is relaxed to allow some error tolerance $\epsilon \geq 0$ [9], such that

$$\text{minimize } ||\mathbf{x}||_1 \text{ subject to } ||\boldsymbol{\phi}\mathbf{x} - \mathbf{y}||_2 \leq \epsilon \tag{3}$$

will help to reconstruct $\mathbf{x}$ with insignificant error [10],[11].

2.2 Foundation of Compressive Sensing

Compressive sensing is based on sparsity and incoherence. Incoherence expresses the idea that the signals that are spread out in the domain in which they are acquired, may have a sparse representation in another domain. For effective reconstruction, it is mandatory that ϕ has to be incoherent with ψ. *Incoherence* implies that the mutual coherence or the maximum magnitude of entries of the product matrix $\phi\psi$ is relatively small. Coherence is measured according to

$$\mu(\phi,\psi) = \sqrt{N}\, max\left|\langle \phi_k, \psi_j \rangle\right| \quad 1 \leq k,j \ \leq N \tag{4}$$

If ϕ and ψ contain correlated elements, the coherence is large [1]. Coherence takes the values in the range [1, $\sqrt{N}$]. The random measurement matrices are largely incoherent with any fixed basis function ψ. Hence the sparsity basis function need not even be known, when designing the measurement system.

The necessary and sufficient condition for the sparse signals to be uniquely determined is that the matrix ϕ should satisfy the Restricted Isometry Property of order K [9], such that

$$(1-\delta_K)\|\mathbf{x}\|_2^2 \leq \|\boldsymbol{\phi}\mathbf{x}\|_2^2 \leq (1+\delta_K)\|\mathbf{x}\|_2^2 \tag{5}$$

where δ_K is the isometry constant of the matrix ϕ and K is the sparsity of the signal. Evaluating RIP for a given matrix being computationally complex, this property is normally verified by computing the coherence of the matrix ϕ. Certain random matrices such as Gaussian and Bernoulli matrices are known to obey RIP [12].

A signal **x** is compressible if its sorted coefficient magnitudes α_n in the transform domain ψ observe a power law decay [11], according to which,

$$|\alpha_n| \leq Rn^{-q},\ n=1, 2, \dots \tag{6}$$

where R and q are constants with $q \geq 1$.

3. COMPRESSIVE SENSING ALGORITHMS

The general block diagram for Compressive Sensing and recovery is shown in Fig. 1. The data from the wave file is segmented into frames followed by transforming the signal into suitable domain for making it sparse. It is then compressed with the help of a measurement matrix. These values can either be stored or transmitted, depending on the requirement. These signals are reconstructed with the help of a compressing sensing algorithm, followed by transforming it to the domain in which the data was acquired and the frames are reassembled to regenerate the signal.

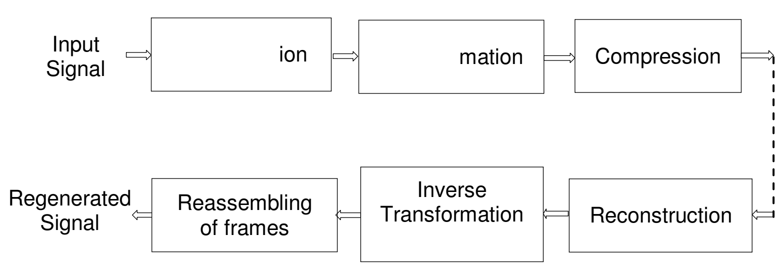

FIGURE 1: Block Diagram for Compressive Sensing and Recovery.

Solution to the sparse recovery problem can be achieved with Relaxation Methods or with Greedy Algorithms. Relaxation methods replace the original sparse recovery problem with a convex optimization problem, whereas Greedy Algorithms focus on finding the non-zero values of **x** at their respective locations, which are determined iteratively. The Greedy Algorithms accumulate the approximations by making locally optimal choices iteratively.

3.1 Relaxation Methods

ℓ_1-norm based sparse recovery problems can be solved using a variety of existing solvers such as ℓ_1-magic, YALL1, l1_ls, etc.. ℓ_1-magic involves recovery of a sparse vector **x** from a small number of linear measurements **y=ϕx** or **y=ϕx+e** using primal-dual method [13], [14], where **e** is the measurement noise,. YALL1 is another solver that can be applied to ℓ_1-optimization, which is a collection of fast ℓ_1-minimization algorithms based on the dual alternating direction method [15], [16]. It is a first order primal-dual algorithm, as it explicitly updates both primal and dual variables at every iteration.

l1_ls solves an optimization problem of the form

$$\text{minimize } \|\boldsymbol{\phi}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda\|\mathbf{x}\|_1 \tag{7}$$

where $\lambda > 0$ is the regularization parameter. Equation (7) is referred to as an ℓ_1-regularized least squares problem. l1_ls is a specialized interior-point method [17] that uses the preconditioned conjugate gradients algorithm to compute the search direction.

3.2 Greedy Algorithms

Greedy Algorithms include Orthogonal Matching Pursuit (OMP), Compressive Sampling Matching Pursuit (CoSaMP), etc.. Orthogonal Matching Pursuit, which is one of the earliest methods for sparse approximation, provides simple and fast implementation [18]. It is an iterative algorithm that selects at each step the dictionary element best correlated with the residual part of the signal [19]. New approximation is generated by projecting the signal on to the dictionary elements that have already been selected and solving a least squares problem. The residual is updated in every iteration. The number of iterations can be made equal to the sparsity of the signal or the stopping criteria can be based on the magnitude of the residual [9]. Compressive Sampling Matching Pursuit [20] which selects multiple columns per iteration [9], is an enhancement to OMP. Each iteration of CoSaMP reduces the error in the current signal approximation.

4. PROPOSED APPROACH FOR SPARSIFICATION AND RECOVERY

In view of the overwhelming limitations of the existing compressive sensing algorithms, in terms of computational complexities, response time, overall reconstruction capabilities, etc., there was a constant search for high performance, more capable and reliable techniques for the sparsification and recovery of audio, speech and natural images. Hence, a new method has been proposed in this paper which involves padding the matrix ϕ during compression phase for the purpose of solving the underdetermined system of simultaneous linear equations, followed by least mean square based adaptation during the reconstruction phase. The solution is obtained with the help of a pseudo-inverse matrix by matrix padding, which is then corrected using iterative least mean square based adaptation.

The signal has to be converted to the domain in which it is sparse, depending on the information and type of the signal. In the simulation studies, the domain chosen is Discrete Cosine Transform (DCT). Conversion of the signal to DCT results in a signal which is sparse with real valued coefficients, thus making the reconstruction easier [21]. The advantages of the Discrete Cosine Transform over Discrete Fourier Transform lies on the fact that it is real-valued, has better energy compaction and as such a sizeable fraction of the signal energy can be represented by a few initial coefficients [22]. The DCT of a 1-D sequence *f(x)* of length *N* is

$$c(u) = \alpha(u) \sum_{x=0}^{N-1} f(x)\, cos\left[\frac{\pi(2x+1)u}{2N}\right] \tag{8}$$

for $u = 0,1,2,\ldots,(N-1)$.
where

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}}, for\ u = 0 \\ \sqrt{\frac{2}{N}}, otherwise \end{cases} \tag{9}$$

The first coefficient, being the average value of the sample sequence, is referred to as the dc coefficient, while all other transform coefficients are called the ac coefficients. Similarly, the inverse DCT is defined as

$$f(x) = \sum_{u=0}^{N-1} \alpha(u)\, c(u)\, cos\left[\frac{\pi(2x+1)u}{2N}\right] \tag{10}$$

for *x = 0, 1, 2,… (N −1)*.

The proposed technique converts the signal into sparse domain by applying DCT, followed by compressing it using a modified measurement matrix. This modification of the measurement matrix has been effected by padding it with a suitable sub matrix for resolving the singularity problems, while solving the underdetermined system of simultaneous linear equations. Making use of this technique, a computationally efficient sparse signal reconstruction can be achieved.

4.1 Matrix Padding

The data from the wave file is divided into N' frames of N samples and these frames are then converted to the frequency domain by using DCT which resulted in a sparse data representation. The compression matrix used is a random Gaussian measurement matrix ϕ of size M x N with $M < N$. In order to make the computation of the matrix inverse feasible during the reconstruction phase at the receiver, it is padded with $(N\text{-}M)$ x N ones which makes the matrix size to N x N. Operation of this modified matrix ϕ' upon the framed sparse data results in a signal matrix **y** that has two sub matrices of which the first sub matrix $\mathbf{y_c}$ gives the data pertaining to the matrix operation $\mathbf{y=\phi x}$, while the other sub matrix $\mathbf{y_{am}}$ provides certain redundant data consequent to the process of matrix padding. Removing the redundant data from $\mathbf{y_{am}}$ results in a vector $\mathbf{y_{av}}$ of size 1 x N'. The matrix $\mathbf{y_c}$ of order M x N' and the vector $\mathbf{y_{av}}$ of order 1 x N' are to be transmitted or stored separately. The algorithmic procedure for the compression is given in ALGORITHM I.

ALGORITHM I: Compression Procedure

Begin
- Read the wave file
- Convert it into N' frames of N samples
- Create a matrix **x(N,N')** with the N' frames of samples
- Generation of modified Measurement matrix
  - Generate Gaussian random measurement matrix ϕ
  - Generate modified Measurement matrix ϕ'
- Compression Phase
  - Multiply **x** with DCT matrix ψ
  - Compress the signal using ϕ'
- Generate y_c (compressed data) and y_{am} (auxiliary matrix)
- Store / transmit y_c and y_{av} (auxiliary vector) separately

End

The signal is reconstructed at the receiver by generating the signal matrix **y'** by appending the received $\mathbf{y_c}'$ with $\mathbf{y_{am}}'$, which is generated from the received $\mathbf{y_{av}}'$ by performing the reverse of the operations carried out at the transmitter. The Moore Penrose inverse of ϕ' is taken and multiplied with **y'** and the data so obtained is converted back to the time domain by the Inverse DCT operation to generate the initial solution, which is refined further by iterative LMS adaptation. The procedure for the recovery of the signal at a later stage is furnished in ALGORITHM II.

4.2 Initial Solution

The problem in sparse recovery addresses the selection of the right solution from the feasible set. In order to find an appropriate solution, consider

$$\text{minimize } \mathbf{J(x)} \text{ subject to } \mathbf{y{=}Ax} \tag{11}$$

Choosing **J(x)** as the squared Euclidean norm $\|\mathbf{x}\|_2^2$ and using Lagrange multipliers,

$$\mathcal{L}(\mathbf{x},\lambda) = \|\mathbf{x}\|_2^2 + \lambda^T(\mathbf{Ax}-\mathbf{y}) \tag{12}$$

where λ is the Lagrange multiplier or dual variable. Derivative of (12) with respect to **x**,

$$\nabla_{\mathbf{x}}\mathcal{L} = 2\mathbf{x} + \mathbf{A}^{\mathrm{T}}\lambda \tag{13}$$

Equating (13) to 0 yields,

$$\mathbf{x}_{\text{initial}} = -\frac{1}{2}\mathbf{A}^{\mathrm{T}}\lambda \tag{14}$$

Substituting (14) into the constraint **y=Ax** of (11) gives,

$$\mathbf{Ax}_{\text{initial}} = -\frac{1}{2}\mathbf{AA}^{\mathrm{T}}\lambda = \mathbf{y} \quad \Longrightarrow \; \lambda = -2(\mathbf{AA}^{\mathrm{T}})^{-1}\mathbf{y} \tag{15}$$

Putting the value of λ from (15) into (14) yields,

$$\mathbf{x}_{\text{initial}} = \mathbf{A}^{\mathrm{T}}(\mathbf{AA}^{\mathrm{T}})^{-1}\mathbf{y} = \mathbf{A}^{\dagger}\mathbf{y} \tag{16}$$

which is the pseudo-inverse solution. As the initial solution is not sparse, an LMS based adaptation has been used to refine the result.

ALGORITHM II: Recovery Procedure

Begin
- Generate the initial solution
  - Retrieve / receive the signals y_c' and y_{av}'
  - Generate y_{am}' from y_{av}'
  - Append y_{am}' to y_c' and generate y'
  - Generate Moore Penrose inverse of ϕ' and multiply it with y'
  - Perform Inverse DCT operation and reassembling of frames
- Perform LMS based adaptation for signal refinement
- For n=1,……Length of *x(n)*
  - Compute the output *y(n)*
  - Compute the error signal *e(n)*
  - Update the filter coefficients *w(n+1)*
- End

End

4.3 LMS Based Adaptation

An adaptive filter is a self-designing one, which relies on a recursive algorithm for its operation that makes it possible for the filter to perform satisfactorily in an environment where complete knowledge of the relevant signal characteristics is not available [23]. For descending towards the minimum on the mean-square-error performance surface of the adaptive filter, least-mean-square or LMS algorithm can be used, which is simple and has less computational complexity. LMS filter is used in a large number of applications like echo cancellation, channel equalization, signal prediction, etc..

If **y**(n) is the n^{th} sample of the observed output signal [24], then

$$\mathbf{y}(\mathrm{n}) = \mathbf{x}^{\mathrm{T}}(\mathrm{n})\, \mathbf{w}(\mathrm{n}) \tag{17}$$

where $\mathbf{w}(\mathrm{n}) = [\mathrm{w}_0(\mathrm{n})\ \mathrm{w}_1(\mathrm{n}) \ldots \ldots \mathrm{w}_{\mathrm{L}-1}(\mathrm{n})]^{\mathrm{T}}$ and $\mathbf{x}(\mathrm{n}) = [\mathrm{x}(\mathrm{n})\ \mathrm{x}(\mathrm{n}-1) \ldots \ldots \mathrm{x}(\mathrm{n}-\mathrm{L}+1)]^{\mathrm{T}}$denote the filter coefficient vector and input vector respectively and *L* is the filter length. The error of the adaptive filter output with respect to the desired response signal **d**(n) is

$$\mathbf{e}(\mathrm{n}) = \mathbf{d}(\mathrm{n}) - \mathbf{x}^{\mathrm{T}}(\mathrm{n})\, \mathbf{w}(\mathrm{n}) \tag{18}$$

By minimizing the cost function which is the mean squared error [25], the filter coefficients are updated iteratively, so that

$$\mathbf{w}(\mathrm{n}+1) = \mathbf{w}(\mathrm{n}) + \mu\, \mathbf{e}(\mathrm{n})\, \mathbf{x}(\mathrm{n}) \tag{19}$$

where μ is the adaptation step size. If **R** is the autocorrelation matrix of the input vector $\mathbf{x}(\mathrm{n})$ and λ_{max}, its maximum eigen value, the condition for convergence for the LMS is [24]

$$0 < \mu < \frac{1}{\lambda_{max}} \tag{20}$$

The structure of adaptive LMS FIR filter is shown in Fig. 2. It is known that the *optimum weight vector*, which is the point at the bottom of the performance surface, is

$$\mathbf{W}^* = \mathbf{R}^{-1}\mathbf{P} \tag{21}$$

where **R** is the input correlation matrix given by

$$\mathbf{R} = \mathrm{E}[\mathbf{x}(\mathrm{n})\mathbf{x}(\mathrm{n})^{\mathrm{T}}] \tag{22}$$

and

$$\mathbf{P} = \mathrm{E}[\mathbf{d}(\mathrm{n})\mathbf{x}(\mathrm{n})] \tag{23}$$

Taking expectation on both sides of (19),

$$\begin{aligned}
\mathrm{E}[\mathbf{w}(\mathrm{n}+1)] &= \mathrm{E}[\mathbf{w}(\mathrm{n})] + \mu\, \mathrm{E}[\mathbf{e}(\mathrm{n})\mathbf{x}(\mathrm{n})] \\
&= \mathrm{E}[\mathbf{w}(\mathrm{n})] + \mu\, \mathrm{E}[\mathbf{d}(\mathrm{n})\mathbf{x}(\mathrm{n}) - \mathbf{x}(\mathrm{n})\, \mathbf{x}^{\mathrm{T}}(\mathrm{n})\, \mathbf{w}(\mathrm{n})] \\
&= \mathrm{E}[\mathbf{w}(\mathrm{n})] + \mu(\mathbf{P} - \mathrm{E}[\mathbf{x}(\mathrm{n})\, \mathbf{x}^{\mathrm{T}}(\mathrm{n})\, \mathbf{w}(\mathrm{n})]) \\
&= \mathrm{E}[\mathbf{w}(\mathrm{n})] + \mu(\mathbf{P} - \mathrm{E}[\mathbf{x}(\mathrm{n})\, \mathbf{x}^{\mathrm{T}}(\mathrm{n})]\, \mathrm{E}[\mathbf{w}(\mathrm{n})])
\end{aligned}$$

($\because$ Coefficient vector $\mathbf{w}(\mathrm{n})$ is independent of input vector $\mathbf{x}(\mathrm{n})$)

$$\mathrm{E}[\mathbf{w}(\mathrm{n}+1)] = \mathrm{E}[\mathbf{w}(\mathrm{n})] + \mu(\mathbf{P} - \mathbf{R}\, \mathrm{E}[\mathbf{w}(\mathrm{n})]) \tag{24}$$

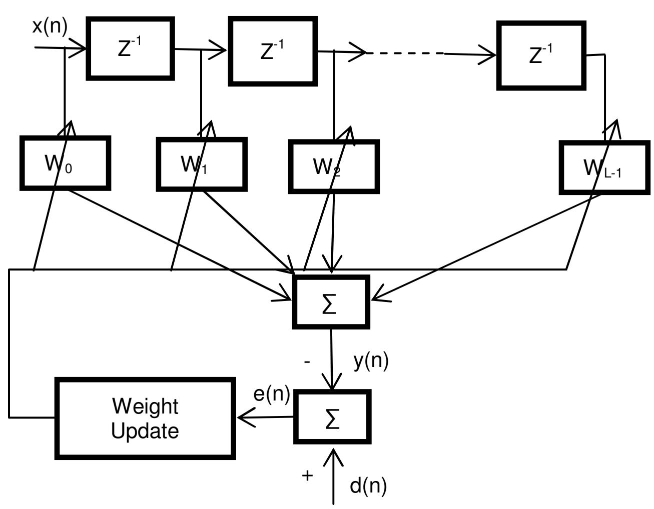

FIGURE 2: Tapped Delay Line Structure of the LMS Filter.

Let the error in coefficient vector **s**(n) be,

$$\mathbf{s}(\mathrm{n}) = \mathbf{w}(\mathrm{n}) - \mathbf{W}^* \tag{25}$$

Substituting (25) in (24),

$$\begin{aligned} \mathrm{E}[\mathbf{s}(\mathrm{n}+1)] &= \mathrm{E}[\mathbf{s}(\mathrm{n})] + \mu\big(\mathbf{P} - \mathbf{R}\,(\mathrm{E}[\mathbf{s}(\mathrm{n})] + \mathbf{W}^*)\big) \\ &= \mathrm{E}[\mathbf{s}(\mathrm{n})] + \mu(\mathbf{P} - \mathbf{R}\,\mathrm{E}[\mathbf{s}(\mathrm{n})] - \mathbf{RW}^*) \\ &= \mathrm{E}[\mathbf{s}(\mathrm{n})] - \mu\,\mathbf{R}\,\mathrm{E}[\mathbf{s}(\mathrm{n})] \qquad (\because \mathbf{P} = \mathbf{RW}^*) \\ &= (\mathbf{I} - \mu\,\mathbf{R})\,\mathrm{E}[\mathbf{s}(\mathrm{n})] \end{aligned} \tag{26}$$

This implies that the mean error in filter coefficients at instant n+1 depend on step size, autocorrelation of the input vector and the mean error in filter coefficients at the instant n.

5. RESULTS AND DISCUSSIONS

In the simulation studies, the number of samples per frame is chosen to be 2048, which resulted in 22 frames for the test signal and the tapped delay line structure has been used for the LMS adaptation with 32 weights.

The proposed approach for compressive sensing has been simulated under noiseless and noisy environments and the performance of this approach has been *vis-a-vis* compared with a few of the widely used compressive sensing recovery methods like ℓ_1-magic, l1_ls, YALL1, OMP and CoSaMP. Comparison of the performances of various algorithms assuming x' as the recovered signal and x as the original signal, has been performed in terms of signal-to-noise ratio, correlation and mean squared error.

Signal-to-Noise Ratio is computed using

$$\mathrm{SNR} = 10\log\frac{\sum \mathrm{x}^2}{\sum(\mathrm{x}-\mathrm{x}')^2} \tag{27}$$

Correlation is computed as

$$\mathrm{R}_{\mathrm{xx}'} = \frac{\sum \mathrm{x}\,\mathrm{x}'}{\sqrt{\sum \mathrm{x}^2 \sum \mathrm{x}'^2}} \tag{28}$$

Mean Squared Error is computed using

$$\mathrm{MSE} = \frac{1}{\mathrm{n}}\sum(\mathrm{x}-\mathrm{x}')^2 \tag{29}$$

5.1 Performance Comparison under Noiseless Scenario

Figs. 3, 4 and 5 show the results of comparison of the performances of the proposed method with the widely used compressive sensing recovery algorithms at 50% compression. The plots show the values of signal-to-noise ratio, correlation and mean squared error for the sparse recovery algorithms under consideration.

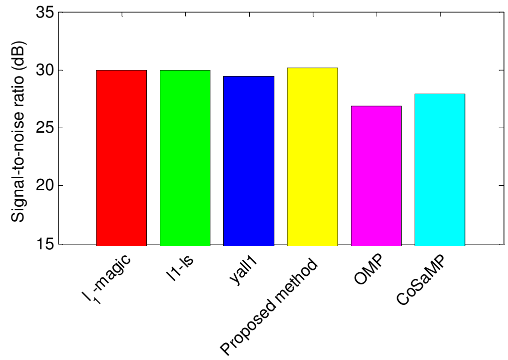

FIGURE 3: Signal-to-noise ratio Performance of the proposed method is compared with the other compressive sensing methods.

These plots reveal that the l1_ls, ℓ_1-magic, YALL1, and the proposed method give comparable and good performance under noiseless scenarios, whereas the signal-to-noise ratio, correlation and mean squared error values of OMP and CoSaMP show that they do not guarantee adequate performance.

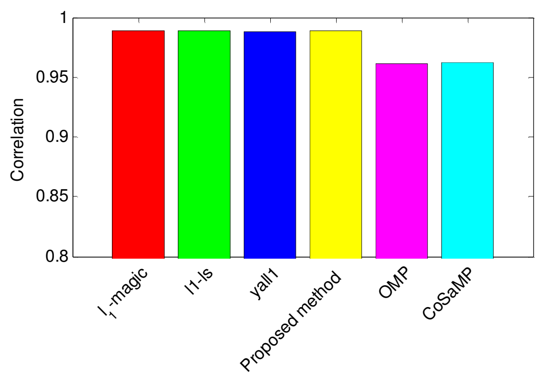

FIGURE 4: Comparison of the Correlation Performance of the proposed method with the other compressive sensing methods.

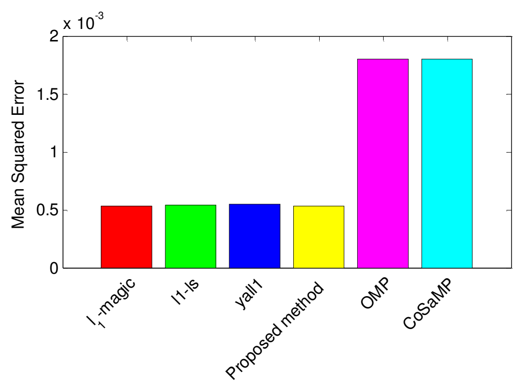

FIGURE 5: Comparison of the Mean Squared Error Performance of the proposed method with the other compressive sensing methods.

5.2 Performance Comparison under Gaussian Noise

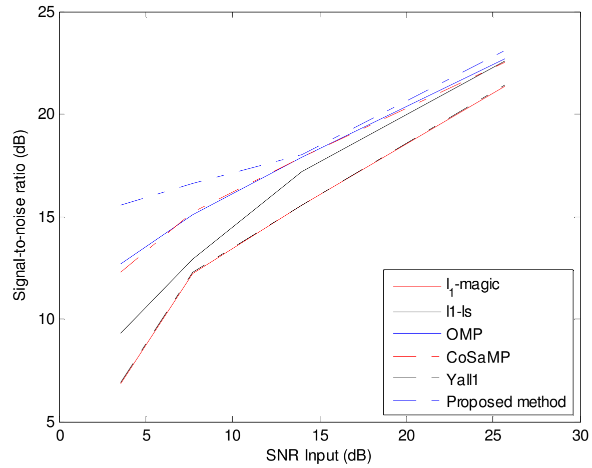

FIGURE 6: Signal-to-noise ratio Performance of the proposed method is compared with the other compressive sensing methods.

Figs. 6, 7 and 8 show the comparison of the performances of the proposed method with the widely used compressive sensing recovery methods, ℓ_1-magic, l1_ls, YALL1, OMP and CoSaMP, under noisy environment. The plots show the variations of output signal-to-noise ratio, correlation and mean squared error with respect to the SNR variation at the input.

At high input signal-to-noise ratios, all the methods show comparable performances. But, as the input signal-to-noise ratio decreases the performance of the proposed method is much better than that of the rest of the methods. OMP and CoSaMP also perform better compared to ℓ_1-magic, l1_ls and YALL1 as the SNR at the input decreases. Thus, the noise immunization of the proposed method is better compared to the other recovery algorithms considered.

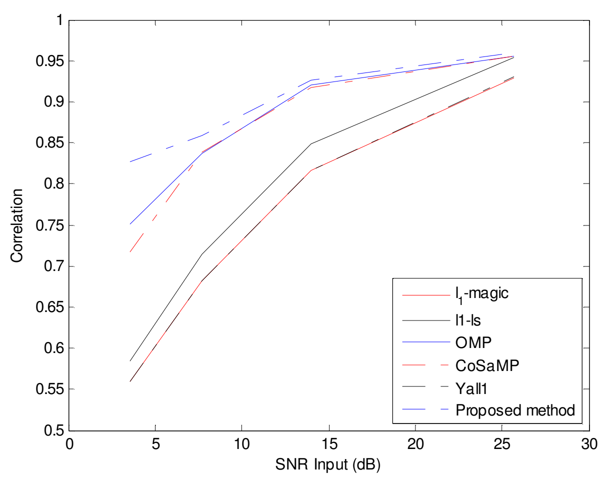

FIGURE 7: Comparison of the Correlation Performance of the proposed method with the other compressive sensing methods.

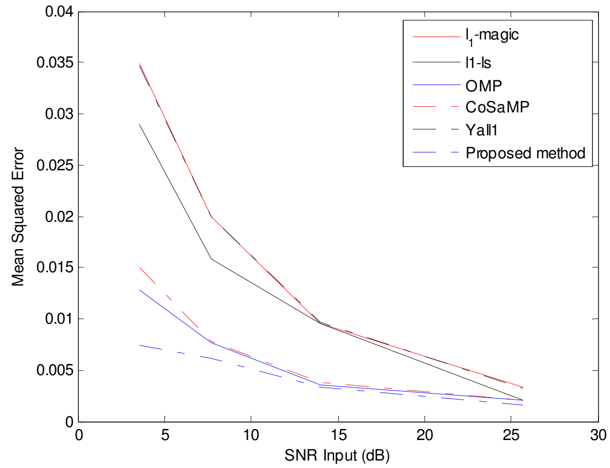

FIGURE 8: Comparison of the Mean Squared Error Performance of the proposed method with the other compressive sensing methods.

The simulation studies further demonstrate the feasibility of improving sparse recovery using the proposed matrix padding technique in both ideal and noisy environments. The mean square error performance of this method is found to be negligibly small when compared to the other compressive sensing algorithms. The proposed sparse recovery algorithms can be effectively used in practical communication scenarios. For undoing the channel effects such as multipath, intersymbol interference, etc., suitable equalization procedures need to be devised. Certain channel characteristics can be estimated with much less overhead using compressive sensing algorithms. The matrix padding sparse reconstruction algorithm will be used for the sparse channel estimation towards nullifying the channel effects.

6. CONCLUSIONS

Compressive sensing recently gained immense attention due to the commendable advantages the technique offers in signal manipulation at comparatively low bit rate requirements. With the help of compressive sensing, the salient information in a signal can be preserved in a relatively small number of linear projections. Compressive sensing has applications in signal processing, in areas such as coding, signal level enhancement, source separation, etc. [6]. For example, a sparse representation has only a few non-zero values, which necessitates encoding only these values to transmit or store the signal. This paper proposed a method for sparse recovery which is robust even in the presence of noise. In the practical scenario, noise cannot be eliminated and hence, the proposed robust signal recovery method is a good choice for the enhancement of corrupted signal. The simulation studies demonstrate that the proposed algorithm can effectively improve sparse recovery with the help of matrix padding and LMS based adaptation in both ideal and noisy environments.

7. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Department of Electronics, Cochin University of Science and Technology, for extending all the facilities for carrying out this work. The authors also acknowledge the Kerala State Council for Science, Technology and Environment as well as the University Grants Commission for the financial assistances.

8. REFERENCES

[1] E.J. Candes and M.B. Wakin. “An Introduction to Compressive Sampling.” IEEE Signal Process. Mag., pp. 21-30, Mar. 2008.

[2] R.G. Baraniuk. “Compressive sensing.” IEEE Signal Process. Mag., vol. 24, no. 4, pp. 118–120,124, 2007.

[3] D.L. Donoho. “Compressed Sensing.” IEEE Trans. Inform. Theory, vol. 52, no. 4, pp. 1289-1305, Apr. 2006.

[4] R.G. Baraniuk, V. Cevher and M.B. Wakin. “Low-Dimensional Models for Dimensionality Reduction and Signal Recovery: A Geometric Perspective.” Proc. IEEE, vol. 98, no. 6, pp. 959-971, Jun. 2010.

[5] L. Vidya, V. Vivekananad, U. ShyamKumar, D. Mishra and R. Lakshminarayanan, “Feasibility Study of Applying Compressed Sensing Recovery Algorithms for Launch Vehicle Telemetry,” in IEEE Int. Conf. Microelectronics, Commun. Renewable Energy, 2013.

[6] M.D. Plumbey, T. Blumensath, L. Daudet, R. Gribonval and M. Davis. “Sparse Representations in Audio and Music: From Coding to Source Separation.” Proc. IEEE, vol. 98, no. 6, pp. 995-1005, Jun. 2010.

[7] E.J. Candes and T. Tao. “Decoding by Linear Programming.” IEEE Trans. Inform. Theory, vol. 51, no. 12, pp. 4203-4215, Dec. 2005.

[8] R.G. Baraniuk, E. Candes, M. Elad and Y. Ma. “Applications of Sparse Representation and Compressive Sensing.” Proc. IEEE, vol. 98, no. 6, pp. 906-912, Jun. 2010.

[9] J.A. Tropp and S.J. Wright. “Computational Methods for Sparse Solution of Linear Inverse Problems.” Proc. IEEE, vol. 98, no. 6, pp. 948-958, Jun. 2010.

[10] E.J. Candès. “Compressive sampling,” in Proc. Int. Congr. Mathematicians, Madrid, Spain, vol. 3, 2006, pp. 1433–1452.

[11] E. Candes, J. Romberg and T. Tao. “Stable Signal Recovery from Incomplete and Inaccurate Measurements.” Commun. Pure Applied Math., vol. 59, no. 8, pp. 1207-1223, Aug. 2006.

[12] R. Baraniuk, M. Davenport, R. DeVore and M. Wakin. “A simple proof of the Restricted Isometry Property for Random Matrices.” Constructive Approximation, vol. 28, no. 3, pp. 253–263, Dec. 2008.

[13] S. Boyd and L. Vandenberghe. Convex Optimization, Cambridge University Press, 2004. Available: https://web.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf

[14] E. Cande`s and J. Romberg. “ℓ_1-MAGIC: Recovery of Sparse Signals via Convex Programming.” California Inst. Technol., Pasadena, CA, Tech. Rep., Oct. 2005. Available: http://users.ece.gatech.edu/~justin/l1magic/downloads/l1magic.pdf

[15] J. Yang and Y. Zhang. “Alternating Direction Algorithms For ℓ_1-Problems In Compressive Sensing.” SIAM J. Scientific Computing, vol. 33, no. 1, pp. 250–278, 2011.

[16] Y. Zhang, J. Yang and W. Yin. “User’s guide for YALL1: Your algorithms for L1 Optimization.” Tech. Rep., 2010. [Online]. Available: http://www.caam.rice.edu/~optimization/L1/YALL1/User_Guide/YALL1v1.0_User_Guide.pdf

[17] S.J. Kim, K. Koh, M. Lustig, S. Boyd and D. Gorinevsky. “An Interior-Point method for Large-Scale l1-Regularized Least Squares.” IEEE J. Select. Topics Signal Process., vol. 1, no. 4, pp. 606–617, Dec. 2007.

[18] J.A. Tropp. "Greed is Good: Algorithmic Results for Sparse Approximation." IEEE Trans. Inform. Theory, vol. 50, no. 10, pp. 2231-2242, Oct. 2004.

[19] J.A. Tropp and A.C. Gilbert. "Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit." IEEE Trans. Inform. Theory, vol. 53, no. 12, pp. 4655-4666, Dec. 2007.

[20] D.Needell and J.A.Tropp. "CoSaMP: Iterative Signal Recovery from Incomplete and Inaccurate Samples." Applied Computational Harmonic Anal., vol. 26, no. 3, pp. 301–321, May 2009.

[21] Moreno-Alvarado and M. Martinez-Garcia. "DCT-Compressive Sampling of Frequency-sparse Audio Signals," in Proc. World Congr. Eng. 2011, vol. II, London, UK, Jul. 2011.

[22] K. Rao & P. Yip. Discrete Cosine Transform - Algorithms, Advantages, Applications. 1st Edition, Elsevier, Aug. 1990.

[23] S. Haykin. Adaptive Filter Theory. 3rd Edition, Prentice Hall, 1996.

[24] Y. Chen, Y. Gu and A.O. Hero III. "Sparse LMS for system identification," in IEEE Int. Conf. Acoustics, Speech and Signal Process, Apr. 2009, pp. 3125 – 3128.

[25] Y. Gu, J. Jin and S. Mei. "ℓ_0 Norm Constraint LMS Algorithm For Sparse System Identification." IEEE Signal Process. Letters, vol. 16, no. 9, pp. 774-777, Sept. 2009.