# Maximum Likelihood Estimation of Time-Varying Sparsity Level for Dynamic Sparse Signals

**THIRUPPATHIRAJAN S.**[1], **LAKSHMI NARAYANAN R.**[2], **SREELAL S.**[1], (Senior Member, IEEE), **AND MANOJ B. S.**[2], (Senior Member, IEEE)

[1]Vikram Sarabhai Space Centre (VSSC), Indian Space Research Organisation (ISRO), Thiruvananthapuram 695022, India
[2]Indian Institute of Space Science and Technology (IIST), Thiruvananthapuram 695547, India

Corresponding author: Manoj B. S. (bsmanoj@iist.ac.in)

**ABSTRACT** In the field of Compressed Sensing, estimation of sparsity level is very essential as the sparsity level determines the minimum number of (i) measurements to be obtained of a sparse signal during acquisition and (ii) iterations to be performed for many of the greedy techniques for the perfect recovery of the sparse signal from the obtained measurements. In this paper, we propose a Maximum Likelihood (ML) estimator to estimate the instantaneous sparsity level during acquisition and an ML sequence (MLS) estimator of sparsity levels during recovery. As the sparsity level varies in time due to the continuous birth of newer supporting components and death of existing supporting components, this paper models the sparsity level variation as a stochastic birth-death process. The real-world applications of the proposed estimators are presented on the compression of aircraft vibration signals and the estimation of wireless channels. The simulation results on real-world and model-generated data demonstrate the performance merits of the proposed estimators compared to other existing methods.

**INDEX TERMS** Dynamic compressed sensing, markov birth-death process, maximum likelihood sequence estimation, time-varying sparsity level estimation.

## I. INTRODUCTION

In recent years, Compressed Sensing (CS) [1]–[5] has emerged as a powerful technique for acquiring high-dimensional sparse signals with fewer measurements than what is dictated by classical Shannon-Nyquist theory. An $N-$dimensional sparse signal is said to be $k-$sparse when it is composed of $k \ll N$ active (or supporting or significant) components belonging to a certain class of orthogonal basis such as the Fourier, cosine, wavelets, etc. The collection of indices of those active components forms the support $\mathcal{S}$ and the cardinality of the set $\mathcal{S}$ is the sparsity level $k$.

In the CS acquisition process, $m$ samples or measurements are obtained using $m \times N-$dimensional random or deterministic sensing basis, where $m < N$, and $m \geq ck \log(N/k)$ for some constant $c$ [4], [5]. In the CS recovery process, the $N-$dimensional sparse signal is reconstructed from those $m$ CS measurements using either convex relaxation based algorithms [6]–[8] or greedy techniques [9], [10]

The associate editor coordinating the review of this manuscript and approving it for publication was Qingchun Chen.

with the prior knowledge of the sensing basis involved during acquisition. The basic theoretical concepts underlying CS can be found in [11].

### A. NEED OF SPARSITY LEVEL ESTIMATION

The objective of CS is to minimize the number of measurements during acquisition and the reconstruction error during recovery. As the sparsity level dictates the number of measurements during acquisition and the number of iterations for many of the greedy techniques during recovery, the efficiency of acquisition and recovery hinges on the knowledge of the sparsity level. Researchers have thus far assumed that the sparsity level is known beforehand [12]–[15] and is time-invariant [16]. In many practical scenarios, sparse signals are characterized by the time-varying sparsity level whose value is seldom known *a priori* [17]–[20]. The wireless channel with time-varying tap coefficients and distribution (tap distribution is the support, whereas the number of tap coefficients is the sparsity level) is a typical example of a real-life sparse system. While applying CS to such systems, it is essential to know the time-varying sparsity level during acquisition and recovery.

A practical CS acquisition system obtains each measurement using a separate Integrator and Dump (I&D) circuit, which multiplies the input sparse signal with an independent sensing basis (typically either a Gaussian or a Bernoulli signal) and integrates it for a fixed duration. There are $m$ such separate I&D circuits working in parallel to obtain $m$ discrete measurements at every time instant. Minimizing the number of measurements amounts to minimizing the number of I&D circuits. Hence instantaneous estimation of sparsity level becomes essential to optimally utilize the hardware resources during acquisition.

The discrete CS measurements are generally stored either for off-line CS recovery as in the case of Magnetic Resonance Imaging (MRI) or near real-time CS recovery as in the case of wireless channel estimation. As the mandate of the CS recovery hardware is to minimize the recovery error, the instantaneous sparsity levels estimated from the obtained measurements need to be refined.

### B. RELATED LITERATURE

Recent approaches to sparsity level estimation [21]–[23] require additional measurements to estimate the sparsity level, where those measurements are not useful for the recovery of the sparse signal. Sparse Gaussian Sensing Matrix (GSM) based sparsity level estimation is given in [24], [25] where the knowledge of signal statistics is required for constructing the sparse GSM, that is seldom known *a priori*. In the context of cognitive radio spectrum sensing, sparsity level estimation using the Monte Carlo simulations is given in [26], and computationally intensive eigenvalue method is presented in [27] assuming that the sparsity level is time-invariant. The estimation based on the trace of the covariance matrix of the measurements is given in [28] assuming that the signal statistics are known *a priori*.

There exist literature on tracking of dynamic sparse signals [17], [20] which assumes that sparse signals are slow-varying and the amplitude of each supporting component varies according to a predefined Gauss-Markov model exhibiting temporal correlation. They recover sparse signals by estimating the support and do not involve sparsity level estimation. Kalman filtering on CS recovery is proposed in [17] where support is estimated using convex optimization based CS recovery. Dynamic Compressed Sensing Approximate Message Passing (DCS-AMP) based tracking is proposed in [20], where in addition to amplitude modeling, the support is modeled using the Discrete Markov process. These support estimation techniques require a longer execution time and are suitable only during recovery and not during CS acquisition.

Recently, the Deterministic Binary Block Diagonal (DBBD) matrix-based sensing [29] and Kronecker-based recovery [30] are developed for acquiring sparse signals. Though the sensing matrix has deterministic binary entries and reduces the hardware complexity, the recovery performance is degraded under noisy settings due to the structure of DBBD. As each row of the DBBD matrix has no overlapping ones with any other rows and accumulates the neighboring significant components' energy, the support estimation of the underlying sparse signals becomes difficult.

Considering the above, there remain issues in sparsity level estimation such as (i) doing away with or minimizing the additional measurements [21]–[23], (ii) overcoming the assumption that the knowledge on sparse signal statistics [24], [25], [28] is available *a priori*, and (iii) adapting to the time-varying signal statistics. In addition to this, practical implementation aspects of sparsity level estimators in the CS acquisition and recovery systems are not being addressed in the literature thus far, to the extent of our knowledge.

We present practically implementable sparsity level estimators that track the time-varying sparsity level using the same set of measurements acquired for use during recovery. We estimate the signal statistics on the fly from the obtained measurements and adapt our CS acquisition system according to the instantaneous signal statistics and sparsity level. The proposed CS recovery system performs better by exploiting the temporal characteristics of sparsity level variation.

### C. KEY CONTRIBUTIONS

As the sparse signal varies in time, there are inclusions of newer basis functions or deletions of the existing basis functions. These inclusions/deletions of basis functions, i.e., births/deaths of the supporting components, result in the variation in the sparsity level lending itself to be modeled using a birth-death process. Hence the time-varying sparsity level is modeled as a discrete-valued Markov birth-death process and is used for the sparsity level estimation.

We use a composite sensing basis to acquire sparse signals where the first few measurements are obtained using a deterministic sparse Binary basis and the rest using a random Gaussian basis. All the acquired measurements are used for recovering the sparse signal. Here the Binary basis is used for estimating the statistics of sparse signal, and the number of measurements obtained using Binary basis is limited due to its weaker isometry bounds [1]. The knowledge of statistics thus obtained is used to estimate the instantaneous sparsity level with the help of measurements obtained using a random Gaussian basis. Thus no additional measurements are required for estimating the sparsity level.

During CS recovery, real-time recovery of sparse signals is not the mandate; whereas, minimizing the recovery error is. Hence we propose a sequence estimation method that exploits the birth-death model mentioned above to output the most probable sequence of sparsity level estimates for use during near real-time recovery. The following is the summary of our contributions:

1) Modeling of time-varying sparsity level using a generalized discrete Markov birth-death process and estimating the model parameters.
2) Statistical characterization of the sparsity level variation using the concept of survival time.
3) Obtaining the Maximum Likelihood (ML) estimator of instantaneous sparsity level, estimating signal and

noise statistics required for the ML estimator, and deriving the performance bounds for the ML estimator.

4) Refining the sequence of ML estimates by exploiting its Markovian property to obtain the most probable sequence of sparsity level estimates.

5) Implementation aspects of sparsity level estimator in CS acquisition and recovery systems.

The proposed method is shown to have improved performance compared to other existing methods in terms of the standard measures through both analysis and simulations.

In essence, the paper obtains the ML estimate of time-varying sparsity level during acquisition and refines it using the Viterbi algorithm during recovery. Since the proposed technique makes no assumptions regarding the signal or noise statistics, it morphs into a joint estimation problem, as detailed in the subsequent sections.

The rest of the paper is structured as follows. Section II presents the CS acquisition model and introduces the Markov birth-death framework for modeling the time-varying sparsity level. Section III derives the sparsity level estimator along with the estimators for signal and noise statistics. Further refinement of the ML estimates using sequence estimation procedure for use during CS recovery is presented in Section IV. The practical implementation aspects of the proposed CS acquisition system aiding the ML estimation and Viterbi algorithm for MLS estimation in the recovery system are given in Section V. Simulation results comparing the recovery performance of the proposed ML-MLS algorithm against the existing methods are presented in Section VI. Real-world applications of the proposed estimators on compression of aircraft vibration signal and estimation of a wireless channel are also presented in Section VI.

The operators commonly used in this paper are $\|.\|_p$, $\Pr[.]$, $\mathbb{E}\{.\}$, and $\mathrm{VAR}\{.\}$ referring to $\ell_p$ norm, probability, expectation, and variance, respectively. The operators $\lfloor.\rceil$, $\lfloor.\rfloor$, and $\lceil.\rceil$, round the argument to nearest integer, greatest preceding integer and least succeeding integer, respectively. The notations $\mathbb{R}$ and $\mathbb{Z}$ indicate domain of reals and integers, respectively. The notation $\mathcal{O}$ represents the order of complexity. The notation $x \sim \mathcal{N}(\mu, \sigma^2)$ indicates $x$ is a Gaussian random variable having mean $\mu$ and variance $\sigma^2$ whereas, $x \sim \mathcal{B}(0, \lambda)$ indicates $x$ is a Bernoulli random variable such that $\Pr[x = 0] = \lambda$ and $\Pr[x = 1] = 1 - \lambda$.

## II. MODELING OF CS ACQUISITION SYSTEM AND TIME-VARYING SPARSITY LEVEL

Consider a continuous time dynamic sparse signal $\mathbf{x}(t)$ which has an $N-$dimensional sparse vector representation $\mathbf{s}(t) = \{s_j(t)\}_{j=1}^N = \{s_1(t), s_2(t), \dots, s_N(t)\}$ using an orthogonal sparsifying basis set $\boldsymbol{\Psi} = \{\psi_j(t)\}_{j=1}^N$ at $t^{th}$ time as given below.

$$\mathbf{x}(t) = \sum_{j=1}^N s_j(t)\psi_j(t), \tag{1}$$

where $s_j(t)$ is the component or coefficient of the $j^{th}$ basis $\psi_j(t)$.

The sparse representation vector $\mathbf{s}(t)$ has very few active components with time-varying support and amplitude and is given as,

$$\mathbf{s}(t) = \mathrm{diag}(\boldsymbol{\kappa}(t))\acute{\mathbf{s}}(t), \tag{2}$$

where

1) $\acute{\mathbf{s}}(t) = \{\acute{s}_j(t)\}_{j=1}^N$ is an $N-$dimensional Gaussian amplitude vector,

2) $\boldsymbol{\kappa}(t) = \{\kappa_j(t) \in \{0, 1\}\}_{j=1}^N$ is a Bernoulli vector, and $s_j(t) = \acute{s}_j(t)$ when $\kappa_j(t) = 1$ and $s_j(t) = 0$ when $\kappa_j(t) = 0$.

For example, $\boldsymbol{\kappa}(t) = [1, 0, 1, 1, 0, \dots, 0, 1]$ indicates that the $1^{st}$, $3^{rd}$, $4^{th}$, and $N^{th}$ components of $\mathbf{s}(t)$ or in other words the $1^{st}$, $3^{rd}$, $4^{th}$, and $N^{th}$ basis functions of $\boldsymbol{\Psi}$ are active at $t^{th}$ time instant. Thus the sparse representation of $\mathbf{x}(t)$ is $\mathbf{s}(t) = [\acute{s}_1(t), 0, \acute{s}_3(t), \acute{s}_4(t), 0, \dots, 0, \acute{s}_N(t)]$ with the support $\mathcal{S}(t) = \{1, 3, 4, N\}$ and the sparsity level $k(t) = 4$.

### A. CS ACQUISITION MODEL

The CS acquisition model acquires $\mathbf{x}(t)$ for every $\mathcal{T}$ seconds to obtain an $m-$dimensional measurement vector $\mathbf{y}(n\mathcal{T}) = \{y_i(n\mathcal{T}) : n \in \mathbb{Z}\}_{i=1}^m$ using $m$ random or deterministic sensing basis functions $\{\phi_i(t)\}_{i=1}^m$ of basis $\boldsymbol{\Phi}$. An $i^{th}$ measurement $y_i(n\mathcal{T})$ is obtained as given below:

$$y_i(n\mathcal{T}) = \int_{t=(n-1)\mathcal{T}}^{n\mathcal{T}} \phi_i(t)\mathbf{x}(t)dt + \vartheta_i(n\mathcal{T}), \tag{3}$$

where

1) $\phi_i(t)$ is either a Gaussian signal or distributed impulses for a duration of $\mathcal{T}$ seconds and it repeats after $\mathcal{T}$ seconds.

2) $\vartheta_i(n\mathcal{T}) \sim \mathcal{N}(0, \sigma_\vartheta^2)$ is the $i^{th}$ component of measurement noise vector $\boldsymbol{\vartheta}(t)$.

Without loss of generality, the CS acquisition model is represented in discrete domain now onwards for better understanding.

### B. DISCRETE CS ACQUISITION MODEL

The discrete version of Equations (1),(2),(3) is given as,

$$\mathbf{x}(n) = \boldsymbol{\Psi}\mathbf{s}(n), \tag{4}$$

$$\mathbf{s}(n) = \mathrm{diag}(\boldsymbol{\kappa}(n))\acute{\mathbf{s}}(n), \tag{5}$$

$$\mathbf{y}(n) = \boldsymbol{\Phi}\mathbf{x}(n) + \boldsymbol{\vartheta}(n), \tag{6}$$

where continuous time $t$ is substituted with discrete time step $n$. The discrete sparse signal $\mathbf{x}(n) = \{x_j(n)\}_{j=1}^N$ has $N$ samples in every time step of duration $\mathcal{T}$ seconds and has a sparse representation $\mathbf{s}(n) = \{s_j(n)\}_{j=1}^N : s_j(n) \sim \mathcal{N}(\mu_{s_j}, \sigma_{s_j}^2)$ when projected onto an $N \times N$ orthogonal signal basis matrix $\boldsymbol{\Psi}$. The discrete version of sensing basis is the $m \times N-$dimensional sensing matrix $\boldsymbol{\Phi} = \{\phi_{i,j}\}_{i=1}^m, _{j=1}^N : \phi_{i,j} \sim \mathcal{N}(0, \sigma_\phi^2)$ when it is Gaussian distributed and $\phi_{ij} \in \{0, 1\}$ when it is binomial distributed.
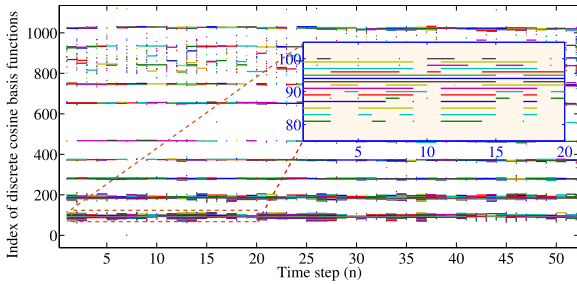
**FIGURE 1.** Time-varying support of a real-world vibration signal.

Substituting Equation (4) in Equation (6), the discrete CS acquisition model becomes,

$$\mathbf{y}(n) = \mathbf{\Phi}\mathbf{\Psi}\mathbf{s}(n) + \boldsymbol{\vartheta}(n). \quad (7)$$

We devise $\mathbf{\Phi}$ as a composite matrix such that

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_{BSM}\mathbf{\Psi}^{-1} \\ \mathbf{\Phi}_{GSM}\mathbf{\Psi}^{-1} \end{bmatrix}, \quad (8)$$

where $\mathbf{\Phi}_{BSM}$ is the deterministic sparse Binary Sensing sub-Matrix (BSM) and $\mathbf{\Phi}_{GSM}$ is the random Gaussian Sensing sub-Matrix (GSM). Their multiplication with inverse of sparsifying basis i.e., $\mathbf{\Psi}^{-1}$ results in direct acquisition of sparse representation $\mathbf{s}(n)$. Thus Equation (7) becomes,

$$\mathbf{y}(n) = \begin{bmatrix} \mathbf{\Phi}_{BSM} \\ \mathbf{\Phi}_{GSM} \end{bmatrix} \mathbf{s}(n) + \boldsymbol{\vartheta}(n). \quad (9)$$

The reason for devising $\mathbf{\Phi}$ as a composite matrix is to maximize the sparsity level estimation performance using BSM and recovery performance using GSM and is explained in Section III.

### C. DISCRETE MARKOV BIRTH-DEATH MODELING OF SPARSITY LEVEL

Any real-world sparse signal has time-varying support due to inclusions (births) and deletions (deaths) of the signal basis functions, as shown in Fig. 1. In this figure, the sparse nature of the support spectrum of a vibration signal of an aircraft [31] using the Discrete Cosine Transform (DCT) basis is shown. The figure captures the time evolution of the active cosine harmonics using different colored tracks. Here it may be noted that there are (*i*) clusters of active basis functions centered about certain resonant frequencies, as shown in the zoomed portion of the plot, (*ii*) large spectral gaps between these clusters which reveal sparsity in the DCT domain, and (*iii*) inclusion and deletion of the DCT basis functions with time, as revealed by the discontinuous lines. Thus the support and its cardinality, i.e., the sparsity level, vary with time and can be modeled as a birth-death process.

The variation in the support is modeled by appropriately changing the entries of the Bernoulli vector $\boldsymbol{\kappa}(n)$ given in Equation (2). The value of difference $(\kappa_j(n) - \kappa_j(n - 1))$ represents the birth, or death, or survival of the $j^{th}$ component depending on whether it is $+1$, $-1$, or $0$, respectively. Due to this difference, the sparsity level $k$ at the $n^{th}$ time step is a

generalized discrete Markov birth-death process and is given as,

$$k(n) = k(n-1) + \underbrace{\sum_{j=1}^{N}(\kappa_j(n) - \kappa_j(n-1))}_{\text{sparsity level variation}}. \quad (10)$$

The probabilities $Pr[(k(n) - k(n - 1)) = d] = p_d : d > 0$, $d < 0$, and $d = 0$ are associated with birth, death and survival of the supporting components. In particular, the probability that the sparsity level remains unchanged between two time steps $p_{(d=0)} = p_0$ determines the degree of sparsity level variation.

### D. STATISTICAL CHARACTERIZATION OF THE SPARSITY LEVEL VARIATION

The sparsity level variation can be either slow, moderate, fast, or rapid. The sparse signals exhibit slow and gradual variations in the sparsity level under steady-state conditions and rapid variations during transients and bursty behavior. The idea of survival time $\widetilde{L}$ is introduced here to statistically quantify the degree and duration of invariance of the sparsity level in time akin to similar concepts in the context of modeling time-varying wireless channels in digital communications.

*Definition 1: Survival time $\widetilde{L}$ is the statistical average number of time steps for which the sparsity level is practically time-invariant and is given by Equation* (11).

As the probability that the sparsity level remains unchanged for $l$ time steps is $p_0^l(1 - p_0)$, the average time steps for which the sparsity level remains unchanged is $p_0/(1 - p_0)$. Thus the survival time $\widetilde{L}$ is,

$$\widetilde{L} = \left\lfloor \frac{p_0}{1 - p_0} \right\rfloor. \quad (11)$$

Intuitively, larger survival time indicates slow variation and smaller survival time indicates rapid variation of the sparsity level. To exemplify, here we classify the sparsity level variation as (*i*) slow when $0.9 \leq p_0 < 1$ ($\widetilde{L} \geq 9$), (*ii*) moderate when $0.8 \leq p_0 < 0.9$ ($4 \leq \widetilde{L} < 10$), (*iii*) fast when $0.6 \leq p_0 < 0.8$ ($2 \leq \widetilde{L} < 5$), and (*iv*) rapid when $p_0 < 0.6$ ($\widetilde{L} \approx 1$), respectively.

An example simulation of time-varying sparsity level $k(n)$ for four different values of $p_0$ is shown in Fig. 2. It is observed that, as $p_0$ decreases, the survival time $\widetilde{L}$ becomes smaller, and the rate of change of sparsity level increases, i.e., when (*i*) $p_0 = 0.98\{\widetilde{L} = 49\}$, (*ii*) $p_0 = 0.9\{\widetilde{L} = 9\}$, (*iii*) $p_0 = 0.75\{\widetilde{L} = 3\}$, and (*iv*) $p_0 = 0.5\{\widetilde{L} = 1\}$, the sparsity level variations are slow, moderate, fast, and rapid, respectively.

### III. MAXIMUM LIKELIHOOD (ML) SPARSITY LEVEL ESTIMATION

We derive an ML estimator to estimate the instantaneous sparsity level when the measurements are obtained using a composite matrix. For the sake of brevity, we ignore the time index $n$ in this section.
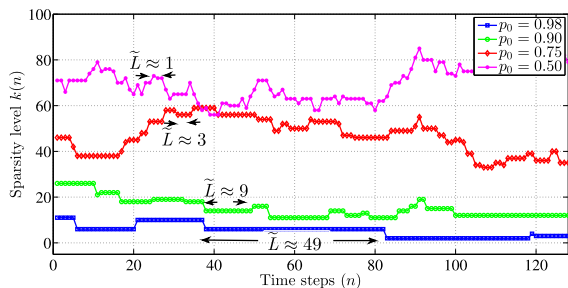
**FIGURE 2.** Simulation of time-varying sparsity level using markov birth-death process.

Each GSM measurement $y_i$ is given by,

$$y_i = \sum_{j \in \mathcal{S}} \phi_{i,j} s_j + \vartheta_i.$$

Since each $\phi_{i,j} \sim \mathcal{N}(0, \sigma_\phi^2)$ of $\mathbf{\Phi}_{GSM}$ is i.i.d Gaussian and is independent of $s_j, \forall j$, the mean and variance of $y_i$ are given as,

$$\mathbb{E}\{y_i\} = 0 \quad \text{and} \quad \text{VAR}\{y_i\} = k\sigma_\phi^2 \sigma_s^2 + \sigma_\vartheta^2,$$

where $\sigma_s^2 = \frac{1}{k} \sum_{j \in \mathcal{S}} (\mu_{s_j}^2 + \sigma_{s_j}^2)$ is the mean energy of supporting components. Since each $y_i$ is sum of $k$ random components, the probability density function (pdf) of $y_i$ follows a Normal distribution according to central limit theorem, i.e.,

$$y_i \sim \mathcal{N}(0, k\sigma_\phi^2 \sigma_s^2 + \sigma_\vartheta^2).$$

Then the joint pdf of $m$ independent random variables $\{y_i\}_{i=1}^m$ parameterised by $k$ is given as,

$$p_Y(y; k) = \frac{1}{(2\pi(k\sigma_\phi^2 \sigma_s^2 + \sigma_\vartheta^2))^{\frac{m}{2}}}$$
$$\times \exp\left[\frac{-1}{2(k\sigma_\phi^2 \sigma_s^2 + \sigma_\vartheta^2)} \sum_{i=1}^m y_i^2\right]. \quad (12)$$

Using the above defined statistical properties of sparse representation and measurement vector, the following theorem defines the ML estimator for the sparsity level.

*Theorem 1 (ML Estimator for the Sparsity Level): Let $\mathbf{y} = \mathbf{\Phi}_{GSM}\mathbf{s} + \mathbf{\vartheta}$ be the CS acquisition model obtaining $m_{ML}$ measurements, where (i) $\mathbf{\Phi}_{GSM} = \{\phi_{i,j}\}_{i=1}^{m_{ML}}{}_{,j=1}^{N}$ is the CS acquisition matrix containing i.i.d entries such that $\phi_{i,j} \sim \mathcal{N}(0, \sigma_\phi^2)$, (ii) $\mathbf{s} = \{s_j\}_{j=1}^N$ is the $k$−sparse vector whose mean energy is $\sigma_s^2$, and (iii) $\mathbf{\vartheta} = \{\vartheta_i\}_{i=1}^{m_{ML}}$ is the measurement noise containing $m_{ML}$ i.i.d entries such that $\vartheta_i \sim \mathcal{N}(0, \sigma_\vartheta^2)$. Then a real-valued ML estimator $\widehat{K}_r$ of sparsity level $k$ which maximizes the joint pdf of measurements parameterised by $k$ and provides the estimate $\widehat{k}_r$ is given by*

$$\widehat{K}_r = \frac{\frac{1}{m_{ML}} \sum_{i=1}^{m_{ML}} y_i^2 - \sigma_\vartheta^2}{\sigma_\phi^2 \sigma_s^2}; \quad (13)$$

*Since the sparsity level $k$ is an integer, an integer-valued ML estimate $\widehat{k}_{ML}$ for the sparsity level $k$ can be obtained from*

$\widehat{K}_r$ by choosing one among the two integer valued estimates $\lfloor \widehat{k}_r \rfloor$ and $\lceil \widehat{k}_r \rceil$ which maximizes the pdf $p_Y(y; k)$ i.e.,

$$\widehat{k}_{ML} = \underset{k \in \{\lfloor \widehat{k}_r \rfloor, \lceil \widehat{k}_r \rceil\}}{\arg \max} \quad p_Y(y; k). \quad (14)$$

Equation (13) reveals that the ML estimator requires the knowledge of $\sigma_s^2$ and $\sigma_\vartheta^2$. The estimate of $\sigma_\vartheta^2$ i.e., $\widehat{\sigma_v^2}$ is obtained from measurements as given in [32],

$$\widehat{\sigma_v^2} = \frac{(n+m+1)\|\mathbf{y}\|_2^2 - \|\mathbf{\Phi}_{GSM}^T \mathbf{y}\|_2^2}{m(m+1)}. \quad (15)$$

The mean energy $\sigma_s^2$ is not available *a priori*, and it has to be computed from the measurements. However, it is not possible to calculate the same from the statistics of GSM based measurements as the variance of each measurement is the same and is a product of both $k$ and $\sigma_s^2$ that are not available *a priori*. Thus the difficulty of estimating $\sigma_s^2$ can be overcome by obtaining a few measurements initially using a sparse BSM, as explained below.

## A. BSM BASED CS ACQUISITION AND ESTIMATION OF STATISTICS OF SUPPORTING COMPONENTS

The sparse BSM $\mathbf{\Phi}_{BSM}$ has a fixed number of $k_b \ll k$ ones in each row. Thus the probability of any randomly chosen element $\phi_{ij} = 1$ in any row of $\mathbf{\Phi}_{BSM}$ is $k_b/N$. Using Equation (9), each sparse BSM measurement $y_i$ is a random sum of supporting components corrupted by measurement noise, i.e.,

$$y_i = \sum_{j=1}^N \phi_{ij} s_j + \vartheta_i = \sum_{j \in \{\mathcal{S}_{\Phi_i} \bigcap \mathcal{S}\}} s_j + \vartheta_i, \quad (16)$$

where $\mathcal{S}_{\Phi_i}$ is the support set of $i^{th}$ row of $\mathbf{\Phi}_{BSM}$ and $\{\mathcal{S}_{\Phi_i} \bigcap \mathcal{S}\}$ contains the set of indices which are common to both $\mathcal{S}_{\Phi_i}$ and $\mathcal{S}$. Since each $s_j \sim \mathcal{N}(\mu_{s_j}, \sigma_{s_j}^2)$, every $y_i$ is a random sum of Gaussians. Consider a special case when none of the ones in $i^{th}$ row of $\mathbf{\Phi}_{BSM}$ multiply with a supporting component in $\mathbf{s}$, resulting in the measurement having contribution from the noise alone, i.e., $y_i = \vartheta_i$. There exists a finite probability of obtaining such measurements, which can be used to estimate the statistics of supporting components.

Since there are $k_b$ ones in each row of the BSM at predetermined positions and $k$ supporting components at randomly distributed over $N$ possible locations, the random sum in Equation (16) consists of an average of $\tau_s = k_b k/N$ supporting components. The mean of the random sum is $\mu_y = \mathbb{E}\{y_i\} = \tau_s \mu_s$, where $\mu_s = \frac{1}{k} \sum_{j \in \mathcal{S}} \mu_{s_j}$ is the mean amplitude of a supporting component. Similarly the variance of random sum is $\sigma_y^2 = \text{VAR}\{y_i\} = \tau_s \sigma_s^2 + \sigma_\vartheta^2$.

As every $y_i$ is a random sum of maximum $k_b$ supporting components, its probability density function (pdf) $p_Y(y_i)$ can be approximated as,

$$p_Y(y_i) : y_i \sim \mathcal{N}(\tau_s \mu_s, \tau_s \sigma_s^2 + \sigma_\vartheta^2). \quad (17)$$

Thus the estimate of statistics $\mu_s$ and $\sigma_s^2$ of the supporting components can be obtained from the BSM measurements,

as given below.

$$\widehat{\mu_s} = \frac{\widehat{\mu_y}}{\tau_s}; \quad \widehat{\sigma_s^2} = \frac{\widehat{\sigma_y^2} - \widehat{\sigma_\vartheta^2}}{\tau_s}, \quad (18)$$

where $\widehat{\mu_y}$ and $\widehat{\sigma_y^2}$ are computed from the $m_b$ number of BSM measurements as,

$$\widehat{\mu_y} = \frac{\sum_{i=1}^{m_b} y_i}{m_b}. \quad \widehat{\sigma_y^2} = \frac{\sum_{i=1}^{m_b}(y_i - \widehat{\mu_y})^2}{m_b - 1}.$$

Note that the estimate of $\sigma_s^2$ depends on $\tau_s = k_b k / N$, which in turn depends on the value of the sparsity level $k$ that is to be estimated. However, $\tau_s$ can be estimated using the following ingenious method.

Suppose $P_\alpha$ is the probability that $\alpha$ number of ones in a row of the BSM multiplies with supporting components of **s**. Then $P_\alpha$ follows a Poisson distribution for very small $k_b$ and it is given as, $P_\alpha = \frac{\exp(-\tau_s)(\tau_s)^\alpha}{\alpha!}$. When $\alpha = 0$, the measurement $y_i = \vartheta_i$. Thus the probability $Pr[y_i = \vartheta_i] = P_0 = \exp(-\tau_s)$, from which $\tau_s = -\ln(P_0)$. The probability $P_0$ can be estimated by finding the proportion of measurements $y_i$ such that $|y_i| \le 3\sigma_v$ covering 99% confidence interval. However, there exists a finite probability $Q$ that for some measurements, the random sum of two or more supporting components would resemble the measurement noise i.e.,

$$Q = \sum_{\alpha=2}^{k_b}(Pr[\text{choosing } \alpha \text{ supporting components}]$$
$$\times Pr[\text{magnitude of sum of those } \alpha \text{ components} \le 3\sigma_\vartheta]);$$

$$= \sum_{\alpha=2}^{k_b} \left( \frac{\exp(-\tau_s)(\tau_s)^\alpha}{\alpha!} \int_{-3\sigma_\vartheta}^{3\sigma_\vartheta} p_Y(y_i) dy_i \right). \quad (19)$$

Thus the estimate of $P_0$ is,

$$\widehat{P_0} = \frac{(\text{number of } y_i \text{ in } \mathbf{y} \text{ such that } |y_i| \le 3\widehat{\sigma_\vartheta})}{m_b} - Q, \quad (20)$$

where $m_b$ is the total number of BSM measurements. Therefore, on computing $Q$, the required probability $P_0$ is estimated to compute $\tau_s = -\ln(\widehat{P_0})$ and substituting in Equation (18) gives the estimates of statistics as,

$$\widehat{\mu_s} = \frac{\widehat{\mu_y}}{-\ln(\widehat{P_0})}; \quad (21)$$

$$\widehat{\sigma_s^2} = \frac{\widehat{\sigma_y^2} - \widehat{\sigma_\vartheta^2}}{-\ln(\widehat{P_0})}; \quad (22)$$

It may be noted that $Q$ depends on $p_Y(y_i)$ which is a function of $\mu_s$ and $\sigma_s^2$. Thus the Equations (19)-(22) are quite interdependent on each other. Therefore, they need to be recursively updated starting with the initial value $Q = 0$. In other words, the updated values of $\tau_s$, $\widehat{\mu_s}$ and $\widehat{\sigma_s^2}$ are used in (19) to compute $Q^1$ and the iteration proceeds through Equations (19)-(22) till $\tau_s$ converges as given in Algorithm 1.

---

[1]As it is difficult to obtain a closed form expression for $Q$, a pre-computed look-up table can be used for the same.

---

**Algorithm 1** : Estimation of Statistics of Supporting Components

**Input:** BSM based measurements $\mathbf{y} : \{y_1, y_2, .., y_{m_b}\}$, Estimate of variance of measurement noise (computed from GSM measurements): $\widehat{\sigma_\vartheta^2}$;

1. Compute mean and variance of the measurements:
   $$\widehat{\mu_y} = \frac{\sum_{i=1}^{m_b} y_i}{m_b}, \quad \widehat{\sigma_y^2} = \frac{\sum_{i=1}^{m_b}(y_i - \widehat{\mu_y})^2}{m_b - 1}.$$

2. Initialise the energy of supporting components:
   $$\widehat{\sigma_s^2} = \widehat{\sigma_y^2} - \widehat{\sigma_\vartheta^2}, \quad \widehat{\mu_s} = \mu_y.$$

3. Compute the probability $\widehat{P_{\sigma_\vartheta}}$ that $|y_i| \le 3\widehat{\sigma_\vartheta}$ i.e.,
   $$\widehat{P_{\sigma_\vartheta}} = \frac{(\text{number of } y_i \text{ in } \mathbf{y} \text{ such that } |y_i| \le 3\widehat{\sigma_\vartheta})}{m_b}.$$

3. Initial estimate of $\tau_s$ is, $\tau_s = -\ln(\widehat{P_{\sigma_\vartheta}})$.

4. Begin iteration:

   4.1 Compute the probability $Q$, a function of $\tau_s, \mu_s, \sigma_s^2$,
   $$Q = \sum_{\alpha=2}^{k_b} \left( \frac{\exp(-\tau_s)(\tau_s)^\alpha}{\alpha!} \int_{-3\widehat{\sigma_\vartheta}}^{3\widehat{\sigma_\vartheta}} p_Y(y_i) dy_i \right).$$

   4.2 Compute $\widehat{P_0} = \widehat{P_{\sigma_\vartheta}} - Q$.

   4.3 Update the estimate of $\tau_s$ and compute the statistics of supporting components.
   $$\tau_s = -\ln(\widehat{P_0}), \widehat{\mu_s} = \frac{\widehat{\mu_y}}{\tau_s}, \widehat{\sigma_s^2} = \frac{\widehat{\sigma_y^2} - \widehat{\sigma_\vartheta^2}}{\tau_s}.$$

   4.4 go to 4.1 until $\tau_s$ converges.

   End iteration;

**Output:** Estimate of statistics of supporting components: $\widehat{\mu_s}$ and $\widehat{\sigma_s^2}$

---

### B. CONSTRUCTION OF BSM

The number of ones $k_b$ i.e. the sparsity level in each row and the number of rows $m_b$ i.e., the number of BSM measurements are the two parameters involved in constructing the BSM. To determine the minimum number of BSM measurements required for obtaining an accurate estimate of sparsity level of the underlying sparse signal, one can minimize the variance of $\tau_s$. The Taylor series approximation of variance of $\tau_s$ as a function of $\frac{k_b}{N}$ is given as,

$$VAR\{\tau_s\} = VAR\{-\ln(\widehat{P_0})\} \quad (23)$$

$$\approx \frac{\sigma_{\widehat{P_0}}^2}{\mu_{\widehat{P_0}}^2} = \frac{\frac{\left(1 - \frac{k_b}{N}\right)^k \left(1 - \left(1 - \frac{k_b}{N}\right)^k\right)}{m_b}}{\left(1 - \frac{k_b}{N}\right)^{2k}} \quad (24)$$

$$= \frac{1 - \left(1 - \frac{k_b}{N}\right)^k}{m_b \left(1 - \frac{k_b}{N}\right)^k}, \quad (25)$$

where $\mu_{\widehat{P_0}}$ and $\sigma_{\widehat{P_0}}^2$ are the mean and variance of $\widehat{P_0}$, respectively. The optimum value of $k_b$ that minimizes $VAR\{\tau_s\}$ satisfies $\frac{\partial VAR\{\tau_s\}}{\partial k_b} = 0$, which results in,

$$\left(1 - \frac{k_b}{N}\right)^k - \frac{k}{2} \ln\left(1 - \frac{k_b}{N}\right) - 1 = 0,$$

which upon simplification using the fact that $\frac{k_b}{N} \ll 1$ gives,

$$k_b \approx N(1 - \exp(-1.6/k)) \qquad (26)$$

From Equation (26), it is observed that the optimum value of $k_b$ is different for different $k$. As the sparsity level $k(n)$ for the current time step is not known, $k_b(n)$ is computed using the previous estimate $\widehat{k}_{ML}(n-1)$.

The size of $m_b$ is determined such that BSM spans all the supporting components which are distributed across $N$ positions for the purpose of estimating the statistics $\sigma_s^2$. Hence the deterministic BSM matrix is designed by stacking $k_b$ numbers of $\lfloor \frac{N}{k_b} \times \frac{N}{k_b} \rfloor$ Identity matrices horizontally. Thus each row of deterministic BSM has $k_b$ ones and there are $\lfloor \frac{N}{k_b} \rfloor$ rows which spans all the components of sparse signal.

As the structure of BSM (i.e., $k_b$) changes according to $k(n)$, the BSM has to be transmitted to the recovery process in every time step along with the acquired measurements for recovery. However, the deterministic construction of BSM avoids transmitting such overhead as recovery process also performs instantaneous sparsity level estimation from the obtained measurements as done during acquisition.

### C. PROPERTIES OF $\widehat{k}_{ML}$ ESTIMATOR

1) Unbiased estimator.
2) The variance of the ML estimator $\widehat{k}_{ML}$ is,

$$\text{VAR}\{\widehat{k}_{ML}\} = \sigma_{\widehat{k}}^2 \approx \frac{2(k\sigma_\phi^2\sigma_s^2 + \sigma_\vartheta^2)^2}{m_{ML}(\sigma_\phi^2\sigma_s^2)^2} + \frac{1}{12}; \quad (27)$$

3) The Cramer Rao Lower Bound (CRLB) for the integer parameter $k$ is given by,

$$\text{CRLB}(\widehat{k}_{ML}) \approx \frac{2(k\sigma_\phi^2\sigma_s^2 + \sigma_\vartheta^2)^2}{m_{ML}(\sigma_\phi^2\sigma_s^2)^2}; \quad (28)$$

It is observed from Equation (28) that, in the absence of measurement noise, the variance of ML estimator approaches the lower bound $2k^2/m_{ML}$ which determines the spread in the ML estimate. The spread in the ML estimate is reduced in the acquisition process when the measurement size $m_{ML}$ is increased, however, as the lower bound is directly proportional to $k^2$, the measurement size $m_{ML}$ required to reduce the spread increases quadratically with the sparsity level $k$. For an efficient CS acquisition, the measurement size cannot be increased beyond the minimum number required for perfect CS recovery. Hence, with a limited number of measurements acquired for CS recovery, the ML estimator provides a ML estimate at every time step.

The number of measurements $m(n)$ required for the perfect CS recovery of $\mathbf{x}(n)$ is given in [33] as $m(n) \geq 1.69 \; k(n) \log(\frac{N}{k(n)})$. Since $k(n)$ is unknown, $\widehat{k}(n-1)$ is used in the place of it for determining $m(n)$. To overcome the degradation in recovery performance due to the randomness of $\widehat{k}(n-1)$, we consider the number of GSM
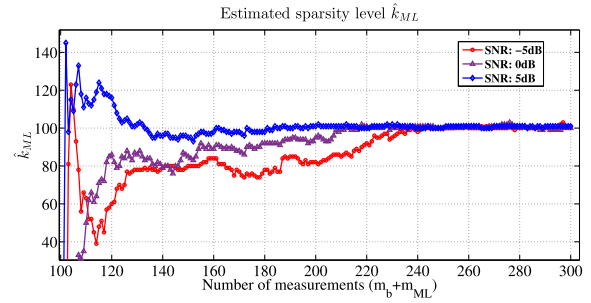


**FIGURE 3.** ML estimation of sparsity level $k = 100$ for different number of measurements under different SNR values.

measurements (after excluding the BSM measurements) to be,

$$m_{ML}(n)$$
$$= 1.69((\widehat{k} + \sigma_{\widehat{k}})(n-1)) \log\left(\frac{N}{(\widehat{k} + \sigma_{\widehat{k}})(n-1)}\right) - m_b(n).$$

Thus this estimator obtains sufficient measurements using BSM and GSM without any compromise on the recovery performance compared to other estimators given in [21], [22], where separate measurements are obtained just for sparsity level estimation that are not useful in the recovery process and entail additional costs during acquisition.

### D. EXPERIMENTAL EVALUATION

This section presents the simulation results on ML estimation. A sparse signal of dimension $N = 1024$ with sparsity level $k = 100$ is generated. The SNR values considered are $-5$, $0$ and $5$ dB. The SNR is computed as $10 \log(k\sigma_s^2/(m\sigma_\vartheta^2))$. For each SNR value, we considered 100 trials and the results are averaged. In every trial, though sparsity level $k$ is maintained as 100, the support is randomly chosen. Also, in every trial, initially, 100 measurements are obtained using BSM for estimating the mean energy of supporting components, i.e., $m_b = 100$. Figure 3 shows the ML estimates we obtained for different SNR conditions. For the SNR values 0 and 5 dB, the averaged sparsity level estimate converges to $100 \pm 1$ with nearly $m_{ML} = 125$ measurements. Whereas for the SNR value of $-5$ dB, the effect of measurement noise is significant, and the estimator requires nearly $m_{ML} = 150$ measurements for convergence. Figure 4 shows the comparison of recovery performance for the different number of measurements in a single trial under different SNR values. Normalized Recovery Error (NRE) $= \|\widehat{\mathbf{x}} - \mathbf{x}\|_2^2/\|\mathbf{x}\|_2^2$, i.e., the ratio of the energy of the difference between the recovered and original signal to the energy of the original signal is shown. It is observed that NRE is initially higher due to the weaker isometry conditions of BSM based measurements. Then NRE reduces drastically as the number of GSM based measurements increases.

### 1) PERFORMANCE COMPARISON OF SPARSITY LEVEL ESTIMATORS

The performance of ML estimator is evaluated and compared with existing Lopes [22] and 2-GMM [24] methods in terms
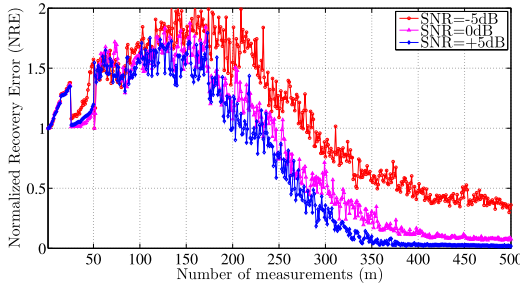
**FIGURE 4.** Recovery performance for different number of measurements under different SNR values.
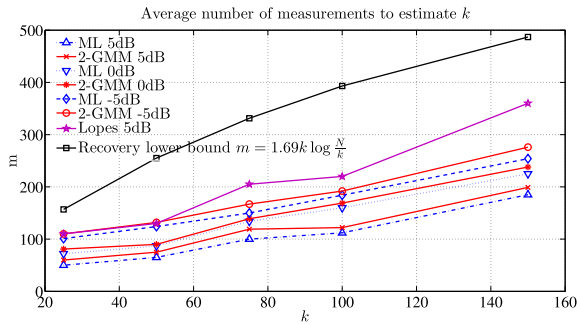


**FIGURE 5.** Comparison of ML estimator with other existing methods in terms of average number of measurements needed to estimate different sparsity levels for different SNR values.



(a) Slow varying: $p_0 = 0.98, SNR = 0dB$
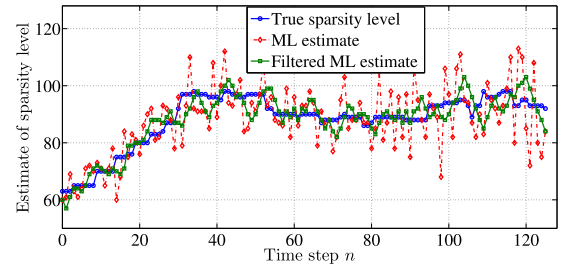


(b) Fast varying: $p_0 = 0.75, SNR = 0dB$

**FIGURE 6.** Effect of MA filtering of ML estimates of slow and fast varying sparsity levels.

of average number of measurements required $m_{ML}$ to estimate various sparsity levels for the SNR values of $-5$, $0$, and $5$ dB. The sparsity levels chosen in the simulation are $k = 25$, $50$, $75$, $100$, and $150$. The sparse signal is of dimension $N = 1024$. The average number of measurements required for estimating each $k$ with an error $|k - \hat{k}_{ML}| \leq 5$ is presented in Fig. 5.

*Observation:* (i) $(m_b + m_{ML})$ is less than the number of measurements required for Lopes and 2-GMM methods for a given SNR, and (ii) $(m_b + m_{ML}) < m = 1.69k \log(N/k)$.

## IV. IMPROVING SPARSITY LEVEL ESTIMATES DURING RECOVERY

During recovery, the ML estimates can be refined further by exploiting the Markovian property of sparsity level variation. This property fits into the framework of MLS estimation that is implemented using the classical Viterbi algorithm. Thus a two-step estimation, i.e., ML, as discussed in Section III followed by MLS (ML-MLS) estimation of the sequence of sparsity levels can be used during recovery.

The MLS estimates $\{\hat{k}_{MLS}(n)\}_{n=1}^{L}$ are obtained by identifying an optimal $L$ length sequence of sparsity levels which has the maximum probability of occurrence under the condition that the ML estimates of sparsity levels $\{\hat{k}_{ML}(n)\}_{n=1}^{L}$ are available, as given below in Equation (29).

$$
\begin{aligned}
&\{\hat{k}_{MLS}(n)\}_{n=1}^{L} \\
&= \underset{\{k_{MLS}(n)\}_{n=1}^{L}}{\arg\max} \; Pr\left[\{k_{MLS}(n)\}_{n=1}^{L} \mid (\{\hat{k}_{ML}(n)\}_{n=1}^{L}, \, p_d)\right].
\end{aligned}
$$
(29)

Before implementing Equation (29), it is required to filter the ML estimates to remove outliers, if any, and to reduce the spread of values in the sequence of the ML estimates $\{\hat{k}_{ML}(n)\}_{n=1}^{L}$. As the sparsity level remains constant on an average of $\widetilde{L}$ Survival time steps, a simple $\widetilde{L}$−tap moving average (MA) filter is considered for prefiltering the ML estimates. The filtered estimate is then rounded to the nearest integer. The effect of filtering the ML estimates using MA is shown in Fig. 6a and Fig. 6b for two different sequences of sparsity levels with slow and fast variations, respectively. The ML estimates of slow and fast varying sparsity levels are filtered using $\widetilde{L} = 10$ and $\widetilde{L} = 3$ taps, respectively. These filtered ML estimates are then refined using the Viterbi algorithm. For the sake of maintaining brevity of the notations, the filtered estimates are also denoted as $\hat{k}_{ML}$ in this and subsequent sections.

### A. VITERBI ALGORITHM

The MLS estimation using the Viterbi algorithm is similar to estimating the most probable sequence of $L$ hidden states from the sequence of $L$ observations emitted by those hidden states in the hidden Markov problem. Here, the MLS and filtered ML estimates are the hidden states and the observations, respectively. Since hidden states are the refined versions of observations in this hidden Markov problem, the transition probability values of hidden states and the emission probability values of observations from those hidden states are the same.

The MLS estimate $\hat{k}_{MLS}(L)$ at $L^{th}$ instant is obtained by maximizing the probability $q_L^v$, which is the probability that the sequence $\{\hat{k}_{MLS}(n)\}_{n=1}^{L-1}$ accounts for first $L - 1$ refined estimates of $\{\hat{k}_{ML}(n)\}_{n=1}^{L}$ and $\hat{k}_{MLS}(L) = v$ at $L^{th}$ instant given
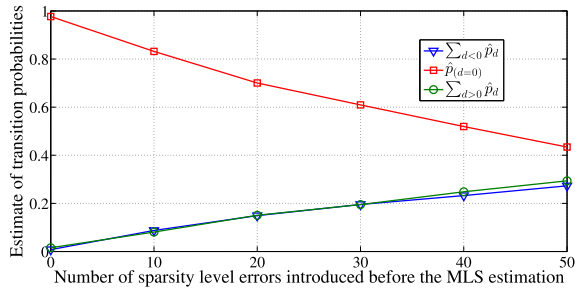
(a) case1: $p_0 = 0.98$
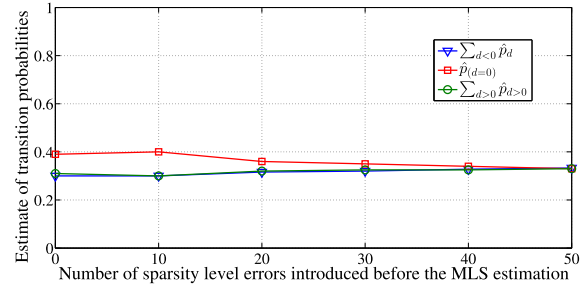


(b) case2: $p_0 = 0.33$

**FIGURE 7.** Effect of ML estimation error on estimating the transition probabilities.

the knowledge of sparsity level variation probability $p_d$, i.e.,

$$q_L^v = \max_{\{\hat{k}_{MLS}(n)\}_{n=1}^{L-1}} Pr[\{\hat{k}_{MLS}(n)\}_{n=1}^{L-1}, \hat{k}_{MLS}(L) = v,$$

$$\times \{\hat{k}_{ML}(n)\}_{n=1}^{L} \mid p_d]. \quad (30)$$

Using induction principle, the highest probability for selecting the sub sequence $\{\hat{k}_{MLS}(n)\}_{n=1}^{L-2}$ such that the sequence $\{\hat{k}_{MLS}(n)\}_{n=1}^{L-2}, \hat{k}_{MLS}(L-1) = u, \hat{k}_{MLS}(L) = v\}$ ending at the sparsity level $v$ can be computed as,

$$q_L^v = \max_u \left[ q_{L-1}^u p_d \right]. \quad (31)$$

Considering $\Gamma_L(v) = \arg \max_u \left[ q_{L-1}^u p_d \right]$, the MLS estimates at the time steps $n = L-1, L-2, \ldots, 1$ can be computed using trace-back technique, i.e., $\hat{k}_{MLS}(n) = \Gamma_{n+1}(\hat{k}_{MLS}(n+1))$.

### B. ML ESTIMATION OF MARKOV MODEL PARAMETERS
As the probability $p_d$ is not available *a priori* for the MLS estimation, it is estimated using the ML principle [34], from the filtered ML estimates. Considering the sequence $\{\hat{k}_{ML}(n)\}_{n=1}^{L}$, the ML estimated transition probability $\hat{p}_d$ is given as,

$$\hat{p}_d = \frac{\text{number of } d \text{ transitions in } \{\hat{k}_{ML}(n)\}_{n=1}^{L}}{L}. \quad (32)$$

### C. SENSITIVITY OF MLS ESTIMATION
The errors in the input sequence $\{\hat{k}_{ML}(n)\}_{n=1}^{L}$ to the MLS estimator affect the estimate of $p_d$, leading to the model mismatch. To demonstrate the sensitivity of the MLS estimator, consider the simulation of the discrete Markov model where the birth and death transition probabilities are considered equal and kept constant throughout the simulation. For each $p_0$, a sequence of 128 sparsity levels is generated and perturbed by introducing errors at random locations to mimic the errors in the ML estimate after filtering. The probabilities $p_{<0} = \sum_{d<0} p_d$, $p_0$, and $p_{>0} = \sum_{d>0} p_d$ are estimated using Equation (32), and the Viterbi estimator is applied on the perturbed sequence. For each $p_0$, the number of errors introduced is 10, 20, 30, 40, and 50. The estimate of transition probabilities for each $p_0$, after the introduction of errors, is shown in Fig. 7. The robustness of MLS estimation method
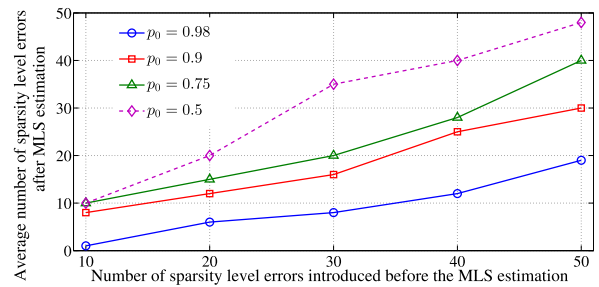


**FIGURE 8.** Robustness of MLS estimation method on correcting the sparsity level errors (errors in the $\hat{k}_{ML}$ estimates) for different values of markov model parameters.

in terms of the number of sparsity level errors before and after the MLS estimation is shown in Fig. 8 for the different values of model parameters.

#### 1) OBSERVATIONS
1) For the slow varying case ($p_0 = 0.98$, $p_{<0} = 0.01$, and $p_{>0} = 0.01$), when the errors in ML estimates are less, the errors in the estimated transition probabilities ($\hat{p}_{<0}, \hat{p}_{>0}$, and $\hat{p}_0$) are also less, as shown in Fig. 7a, and the MLS estimation is robust enough to correct most of the errors, as shown in Fig. 8. However, when the errors in ML estimates are large and bursty in nature, the errors in the estimated transition probabilities are large, as shown in Fig. 7a, and the performance of Viterbi algorithm deteriorates and exhibits poor performance as shown in Fig. 8.

2) The worst-case scenario is for the rapid varying case i.e., when $p_0 = 0.33$, $p_{<0} = 0.33$, and $p_{>0} = 0.34$. Here the errors in the ML estimates of sparsity level are indistinguishable from that of natural variation in the sparsity level, and the ML estimated transition probabilities are independent of the number of errors introduced as shown in Fig. 7b. Therefore, the Viterbi estimator cannot recognize those errors, and hence the correction is reduced, as shown in Fig. 8.

Thus the error in the estimate of transition probabilities harms the MLS output only if many ML estimates are in error or if the rate of change of sparsity level is very high. It may be observed from Fig. 7a–7b that when no errors are introduced

---

**Algorithm 2** ML-MLS - During Recovery

**Input:** $\mathbf{y}(n)$, Known values: $\widehat{\sigma_\vartheta^2}, \widetilde{L}, L$;

1. Estimate the mean energy of supporting components using Algorithm 1.
2. Estimate the sparsity level using GSM:
$$\hat{k}_r(n) = \frac{\frac{1}{m_{ML}}\sum_{i=1}^{m_{ML}} y_i^2(n) - \widehat{\sigma_\vartheta^2}}{\sigma_\phi^2 \widehat{\sigma_s^2}};$$
   Integer-valued ML estimate:
$$\widehat{k}_{ML}(n) = \arg\max_{k \in \{\lfloor \hat{k}_r \rfloor, \lceil \hat{k}_r \rceil\}} p_Y(y; k);$$
4. Filter the ML estimate using a MA filter of length $\widetilde{L}$:
$$\hat{k}_{ML}(n) = \lfloor \textstyle\sum_{I=n-\widetilde{L}+1}^{n} \hat{k}_{ML}(I)/\widetilde{L} \rceil;$$
5. Estimate the transition probability values of the Markov model:
$$\hat{p}_d = \frac{\text{number of } d \text{ transitions in } \{\hat{k}_{ML}(I)\}_{I=n-L+1}^{n}}{L};$$
6. Apply Viterbi algorithm on the sequence $\{\hat{k}_{ML}(I)\}_{I=n-L+1}^{n}$ using estimated transition probability values to get the MLS estimates $\{\hat{k}_{MLS}(I)\}_{I=n-L+1}^{n}$;
7. $n=n+1$; Repeat from step 1.

**Output:** MLS estimates: $\hat{k}_{MLS}(n)$

---

in the sparsity level, the estimate of transition probabilities is converging to the true value of transition probabilities.

The proposed two-step ML-MLS estimation technique is summarized and is given as Algorithm 2.

## V. PRACTICAL IMPLEMENTATION ASPECTS
This section covers the implementation aspects of the CS acquisition system proposed for ML estimation of sparsity level and Viterbi algorithm based MLS estimation for the CS recovery system.

### A. IMPLEMENTATION OF CS ACQUISITION SYSTEM
The practical real-time CS acquisition hardware for the proposed estimation method is shown in Fig. 9. There are $m$ identical and independent modulator circuits working in parallel. The hardware components of a modulator circuit are (i) a multiplexer to select between Gaussian basis $\mathbf{g}(t)$ (continuous-time version of rows of GSM) and Bernoulli basis $\mathbf{b}(t)$ (continuous-time version of rows of BSM) and (ii) a multiplier and an I&D circuit to multiply the sparse signal with measurement basis for a duration of $\mathcal{T}$ seconds to output a measurement. The select signal $\beta_i(t)$ of the multiplexer takes the value 0 for $i \leq m_b$ to select the Bernoulli basis and 1 for $m_b < i \leq m$ to select Gaussian basis.

### B. IMPLEMENTATION OF VITERBI ALGORITHM IN CS RECOVERY SYSTEM
The main component of sparsity level estimator in the CS recovery system is the Viterbi algorithm. Here sliding window based Viterbi algorithm is considered. There exist two parallel sliding windows and each window requires a Viterbi
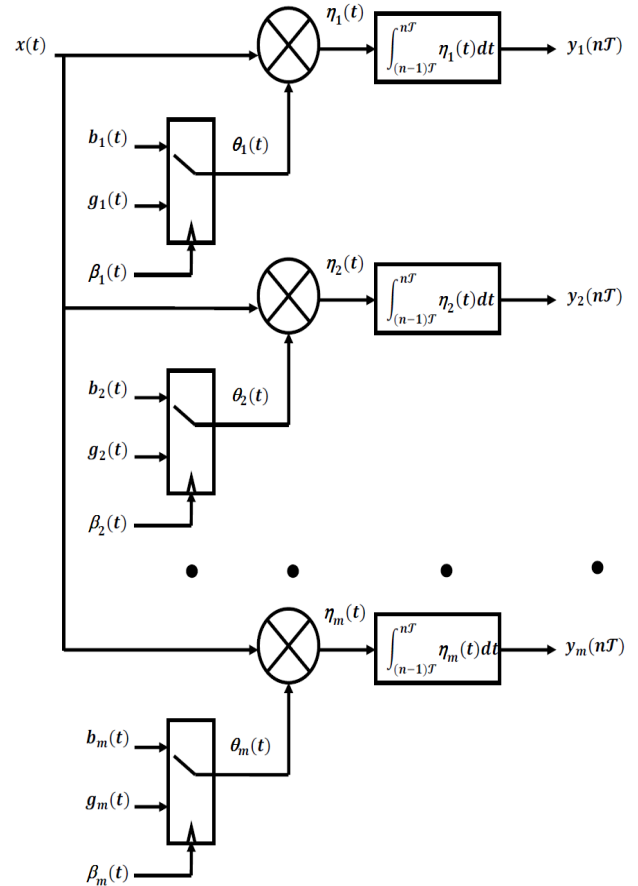


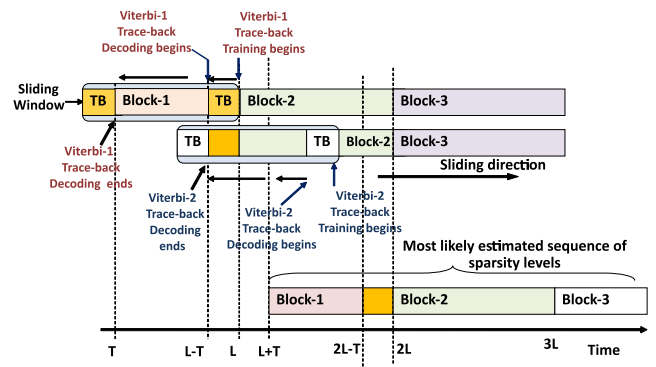**FIGURE 9.** CS acquisition hardware.



**FIGURE 10.** Illustration of sliding window viterbi algorithm.

estimator. The $L$ length sequence of ML estimates of sparsity level is decomposed into blocks of $M$ number of ML estimates. The value $M$ is five times the trace-back length $T$ where $T$ is the maximum sparsity level transition possible in a time step. Each sliding window size is kept similar to the block size, $M$. Both the Viterbi estimators do not require the knowledge of initial state probabilities and the trellis terminated state.

From Fig. 10, it can be observed that, after receiving Block-1 at time $n = M$, the first Viterbi estimator starts the training of trace-back from the sub-block, *TB*. After $T$ time steps

i.e., at $n = (M + T)^{th}$ time step, the first Viterbi estimator starts providing the MLS estimates of sparsity levels received during time steps $n = T$ to $n = M - T - 1$. In the meanwhile, the second Viterbi estimator starts the training of trace-back from the time step $n = 2M - 2T$. After $T$ time steps i.e., at $n = (2M - T)^{th}$ time step, the second Viterbi estimator starts providing the MLS estimates of sparsity levels received during time steps $n = M - T$ to $n = 2M - 3T - 1$. Thus with a group delay of $M$ time steps, the sliding window based Viterbi algorithm provides MLS estimates of sparsity levels without knowing initial state probabilities and trellis termination.

It may be noted that for fast and rapid varying sparsity levels, the Viterbi algorithm has poor performance in correcting the errors introduced by the filtered estimates $\hat{k}_{ML}$ as shown in Fig. 8. Thus the proposed Viterbi algorithm based MLS estimation technique is suitable for slow and moderate varying sparsity levels to provide better MLS estimates with less computational complexity. If the rate of sparsity level variation is higher than that of the estimation speed, then the filtered ML estimates are retained as the best estimates without executing the Viterbi algorithm.

## VI. PERFORMANCE EVALUATION FOR THE PROPOSED ML-MLS ESTIMATION

The performance of the proposed estimator is evaluated by applying it to model-generated data and real-world signals. The performance measures such as Mean Recovery Error (MRE) $= |(\hat{k} - k)/k|$, NRE, and the computational complexity are compared between the proposed and other sparsity level estimators described in [22], [24] and [28] as well as support estimators (size of estimated support gives sparsity level) such as [20], [30], and [17], and an Oracle CS estimator which is aware of signal and noise statistics for the support recovery.

### A. COMPARISON ON COMPUTATIONAL COMPLEXITY

The performance of ML-MLS estimation technique is compared in terms of the computational complexity with Lopes method [22], 2-GMM method [24], and Trace based method [28] where sparsity level is estimated, and DCS-AMP method [20], DBBD-Kronecker [30], and KFCS method [17] where support is estimated which gives knowledge on the sparsity level for a given $m$ number of measurements, as follows.

The proposed estimator requires $m = m_b + m_{ML}$ measurements per time step, where $m_b$ measurements are for estimating the mean energy of the supporting components, and $m_{ML}$ measurements are for estimating the sparsity level. The computational complexity for estimating the instantaneous sparsity level using $m$ measurements is $\mathcal{O}(m)$.

At any time step, the Viterbi algorithm performs the MLS estimation over a block of $L$ length whose computational complexity is $\mathcal{O}(D_R^2 L)$, where $D_R = k_{max} - k_{min} + 1$ is the dynamic range of sparsity level, $k_{min}$ and $k_{max}$ are the minimum and maximum sparsity level, respectively in the $L$ length block. Thus the overall computational complexity for the proposed ML-MLS method is $\mathcal{O}(m) + \mathcal{O}(D_R^2 L)$.

The computational complexity of other methods are given in Table 1, where the remarks explain the reason for their computational complexity. The proposed ML-MLS estimation technique demands an additional computational complexity of $\mathcal{O}(D_R^2 L)$ which is lesser than $\mathcal{O}(mk^3)$ of KFCS and $\mathcal{O}(mN)$ of DCS-AMP methods. Except for estimating the mean energy of supporting components, the ML-MLS technique does not involve any iterations. Though the DBBD-Kronecker method has a better computational complexity of support estimation, which is $\mathcal{O}(N - m)$ and is lesser than that of proposed ML-MLS, its recovery performance is very much affected due to the structure of DBBD.

### B. PERFORMANCE EVALUATION ON MODEL-GENERATED DATA

A time-varying Gaussian sparse signal $\mathbf{x}(n) \in \mathbb{R}^{1024 \times 1}$ with $k(0)$ such that $64 \leq k(0) \leq 128$, and supporting components $\acute{s}_i(n) \sim \mathcal{N}(0, 2)$ is generated using the model parameter $p_{<0} = 0.05, p_0 = 0.9, p_{>0} = 0.05$, for 128 time steps. The input to the Viterbi Algorithm is filtered using $\widetilde{L} = 9-$tap MA filter. With the estimated sparsity level, Orthogonal Matching Pursuit (OMP) [9] algorithm is run to find the supporting components and recover the sparse signal $\hat{\mathbf{x}}(n)$. In every trial of the simulation, the number of obtained measurements $m = 4k$. The measurements are obtained for different SNR values. We considered 500 trials in the simulation, and the averaged MRE and NRE results are plotted in Fig. 11 and Fig. 12, respectively. It is observed that the proposed two-step estimation algorithm significantly outperforms the other methods in terms of MRE and NRE. It is natural to expect improved performance for the ML-MLS estimator, which is close to the oracle estimator because the instantaneous ML estimate is improved upon by exploiting the temporal correlation defined by the Markov model.

It is observed that the performance of the Lopes method [22] is inferior and considerably invariant to the SNR. This is because the Lopes method uses the random Cauchy sensing matrix for which the variance is infinite. As Trace-based method [28] is more suitable for estimating time-invariant sparsity level, its performance is poor. Though both the KFCS and DCS-AMP estimators (which exploit the temporal correlation of sparse signal) perform at par with the proposed ML-MLS estimator, and Oracle estimator, their computational complexity results in harder practical implementation.

### C. PERFORMANCE EVALUATION ON REAL-WORLD DATA

The performance of the ML-MLS estimation method is evaluated here for the real-world applications on compression and recovery of sparse vibration signals and estimation of wireless channels.

#### 1) APPLICATION ON COMPRESSION AND RECOVERY OF VIBRATION SIGNAL DATA OF AN AIRCRAFT

Real-world vibration signals were obtained from Mide Technologies [31]. The vibration signal acquired on the aircraft

**TABLE 1.** Computational complexity comparison.

| Method | Computational complexity | Remarks |
|---|---|---|
| Proposed ML-MLS | $\mathcal{O}(m)+\mathcal{O}(D_R^2 L)$ | ML and MLS estimation |
| Lopes | $\mathcal{O}(m \log m)$ | median calculation from $m$ Cauchy basis based measurements |
| 2-GMM | $\mathcal{O}(km)$ | $k$ iteration per measurement for EM algorithm |
| Trace | $\mathcal{O}(m^3)$ | Computation of Covariance of measurements |
| KF-CS | $\mathcal{O}(mk^3)$ | Complexity of Dantzig selector |
| DCS-AMP | $\mathcal{O}(mN)$ | AMP's $m$ iterative updation of $N$ variables |
| DBBD-Kronecker | $\mathcal{O}(N-m)$ | DBBD acquisition's N-m additions give information about support |
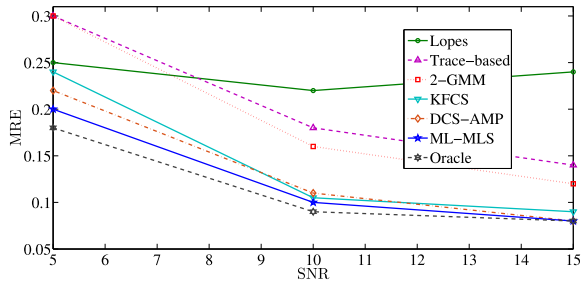


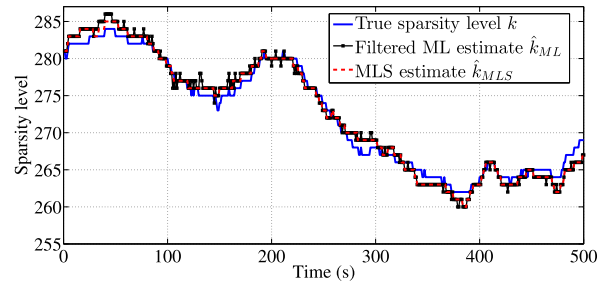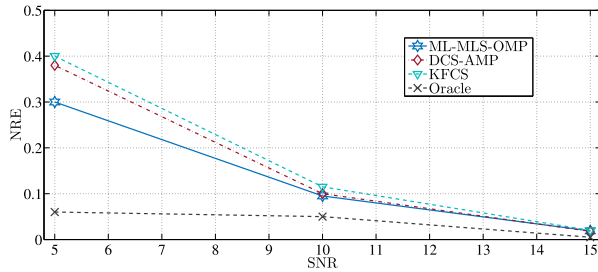**FIGURE 11.** Comparison of MRE performance.



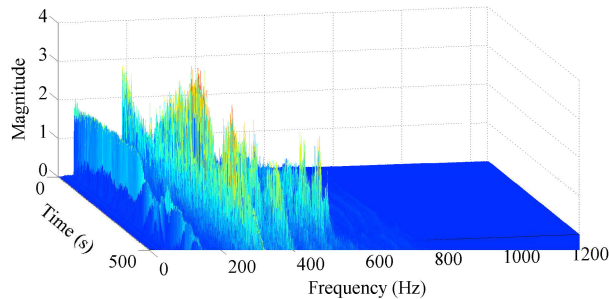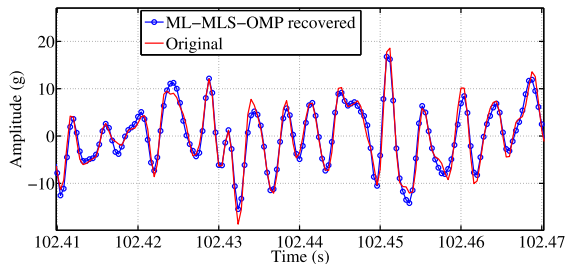**FIGURE 12.** Comparison of NRE performance.



**FIGURE 13.** Time-frequency spectrogram of vibration signal showing that only a few components are having significant magnitudes.

skin at a sampling rate of 2500 sps for 500 seconds during the aircraft's climb is analyzed here. The Time-Frequency spectrogram of that vibration signal is shown in Fig. 13, which reveals that (i) the vibration signal is compressible in the frequency domain as only a very few frequency components are significant at any given time step, and (ii) the number of significant frequency components varies in time.

In our analysis, the duration of 500 seconds is divided into 500 time steps of one second duration and DCT is applied on the vibration signal in every one second time step. The sparsity level $k(n)$ is computed such that 95% energy of $N = 2500$−dimensional vibration signal is contained in



**FIGURE 14.** MLS estimation corrects the ML estimate errors and tracks the actual sparsity levels.

$k(n)$ number of the most significant DCT coefficients and it is plotted in Fig. 14. It is analyzed that the sparsity level variation is moderate and follows a discrete Markov model with transition probabilities $\sum_{d>0} p_d = 0.08$, $\sum_{d<0} p_d = 0.11$, and $p_0 = 0.81$. After obtaining the mean energy of the frequency components using BSM as discussed in Algorithm 1, the ML estimate of $k(n)$ is computed at each time step using Equation (14) and the number of acquired measurements $m(n)$ gives the compression ratio CR $= N/m(n)$. On the recovery side, the MA filter length is kept as $\tilde{L} = 5$. The transition probabilities are estimated as $\sum_{d>0} \hat{p}_d = 0.10$, $\sum_{d<0} \hat{p}_d = 0.14$, and $\hat{p}_0 = 0.76$ and the MLS estimation is performed using Viterbi algorithm. The ML and MLS estimates are plotted in Fig. 14 along with the actual sparsity level and it is verified that MLS estimation refines the ML estimates.

After the ML-MLS estimation, the vibration signal is recovered using the OMP recovery algorithm, which iterates for $\hat{k}_{MLS}(n)$ number of times at each time step. The recovery performance of the proposed estimation followed by the OMP algorithm (ML-MLS-OMP) for the total duration of 500 seconds is shown in Fig. 15. In each time step, the recovery performance is computed in terms of NRE. A snapshot of the ML-MLS-OMP based CS recovered signal along with the original vibration signal is shown in Fig. 15a. The NRE performance of the proposed ML-MLS-OMP method with DCS-AMP and KFCS-based CS methods, and traditional DCT based compression method is presented in Fig. 15b. It is observed that the NRE performance of the ML-MLS-OMP method is comparable with that of the traditional DCT based compression method and at par with DCS-AMP and KFCS methods. For the DCT based compression method, the most significant coefficients and their locations are well known *a priori* resulting in a minimum 5% NRE. Whereas for

(a) A segment of ML-MLS-OMP based CS recovered signal compared with that of the original vibration signal
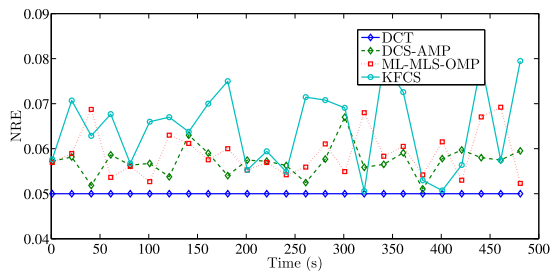


(b) Normalised Recovery Error performance

**FIGURE 15.** Comparison of recovery performance on vibration signal.

**TABLE 2.** Elapsed time comparison among SLE methods.

| SOE methods | Algorithms | Elapsed time |
|---|---|---|
| ML-MLS | ML + Viterbi Algorithm | 55 ms |
| Lopes | median + mean computation | 64 ms |
| 2-GMM | EM algorithm | 125 ms |

the ML-MLS-OMP based CS method, the most significant components and their locations are not available *a priori* or directly, and each measurement is affected by the insignificant DCT coefficients. This makes the NRE performance of the ML-MLS-OMP based CS method to have a slight degradation with respect to the DCT based compression method.

It may be noted that all vibration signals will not be sparse. For example, the vibration signal of a rocket during the transonic regime is almost a random signal rich in significant components making the signal non-sparse. Here, we do not consider such non-sparse signals.

The run-time complexity of the proposed ML-MLS method is evaluated and compared with other existing Sparsity Level Estimation (SLE) methods on a Windows 7 OS based PC having a processor running with 1.66 GHz clock speed. The average elapsed time for the SLE on analyzing each segment of the vibration signal is shown in Table 2.

The time elapsed by the proposed ML-MLS method is significantly less than that of other existing SLE methods.

### 2) APPLICATION ON ESTIMATION OF SPARSE IMPULSE RESPONSE OF WIRELESS CHANNELS

Consider the problem of estimating the wireless channel impulse response (CIR) from the model,

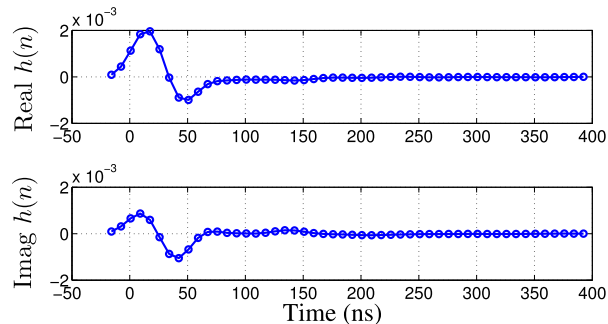$$\mathbf{y} = \mathbf{X}\mathbf{h} + \boldsymbol{\vartheta},$$



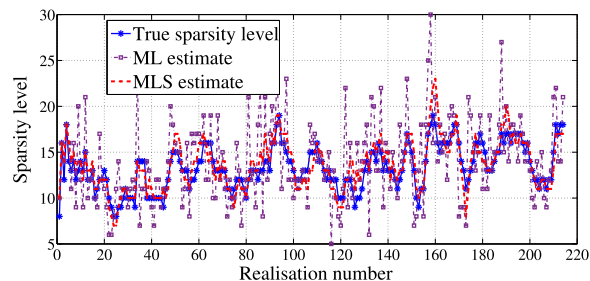**FIGURE 16.** Channel impulse response measured between transmitter 1 and receiver 2.



**FIGURE 17.** Performance of ML and MLS estimation techniques.

where $\mathbf{y}$, $\mathbf{X}$, $\mathbf{h}$ and $\boldsymbol{\vartheta}$ are the received measurements, Toeplitz matrix of known pilots, CIR, and measurement noise, respectively. The above model becomes the CS acquisition model as the CIR $\mathbf{h}$ is sparse, and $\mathbf{X}$ is carefully chosen as a sensing matrix. As the minimum number of measurements obtained are dependent on the sparsity level of CIR, the sparsity level estimation is essential to reduce the number of pilots. Thus the performance of the proposed sparsity level estimator in terms of estimation accuracy is evaluated on a real-world measured CIR obtained from CRAWDAD data set [35]. The CIR was recorded using a 44-node wireless network in a standard office environment. There were $44 \times 43 = 1892$ pairwise links. The transmitter and receiver were moved and CIR measurements were taken multiple times between the nodes. A sample real and imaginary realization of CIR between nodes 1 and 2 are shown in Fig. 16. The time-varying sparsity level is shown in Fig. 17 by stacking the several realizations of CIR measurements. It is observed that sparsity level variation follows a discrete Markov process with transition probabilities $\sum_{d>0} p_d = 0.46$, $\sum_{d<0} p_d = 0.43$, and $p_0 = 0.11$ resulting in fast sparsity level variation. After obtaining the mean energy of the multipath components using Algorithm 1, the ML estimate of $k(n)$ is computed for each realization using Equation (14). The MA filter length is kept as $\widetilde{L} = 3$. As the sparsity level varies fast, the ML estimates of transition probabilities are computed as equally distributed i.e., $\sum_{d>0} \hat{p}_d = 0.33$, $\sum_{d<0} \hat{p}_d = 0.33$, and $\hat{p}_0 = 0.34$, from the sequence of filtered ML estimates. The MLS estimation is performed using the Viterbi algorithm. The unfiltered ML estimates and refined MLS estimates are plotted in Fig. 17 along with the actual sparsity level, and it is verified that MLS estimation refines the unfiltered ML estimates.

## VII. CONCLUSION

In this paper, novel sparsity level estimation techniques for the acquisition and recovery of time-varying sparse signals have been proposed using the ML principle. The ML estimator estimates the sparsity level with a minimum number of measurements without any additional cost or resources compared to any other estimator in the acquisition process. In the recovery process, a two-step sparsity level estimation is proposed, in which the first step is the ML estimation, and the second step is the MLS estimation. The MLS estimation technique filters the ML estimates and refines the sequence of filtered ML estimates using the Viterbi algorithm. The performance of the proposed two-step ML-MLS technique is comparable to that of an oracle estimator. It is an improved one compared to other published methods, even under high level of measurement noise.

## VIII. FUTURE WORK

The present work focuses only on the sparsity level estimation for time-varying sparse signals. However, the problem of estimation and tracking of time-varying sparse signals is three dimensional, where time-varying (*i*) sparsity level, (*ii*) support, and (*iii*) amplitude have to be estimated simultaneously. Hence as future work, characterization of time-varying sparsity level and support together as a two dimensional discrete Markov birth-death model and derivation of corresponding ML algorithms can be considered.

## APPENDIX: PROOF FOR THE PROPERTIES OF ML ESTIMATOR

1) The ML estimator is an unbiased estimator. Since $\widehat{k}_r = \widehat{k}_{ML} + k_\delta$, where $k_\delta \sim \mathbb{U}\{-0.5, 0.5\}$ is the uniformly distributed round-off error. Thus, $\mathbb{E}\{\widehat{k}_r\} = \mathbb{E}\{\widehat{k}_{ML}\} + \mathbb{E}\{k_\delta\} = \mathbb{E}\{\widehat{k}_{ML}\}$. Since,

$$\mathbb{E}\{\widehat{k}_{ML}\} = \mathbb{E}\{\widehat{k}_r\} = \frac{\frac{1}{m_{ML}}\sum_{i=1}^{m_{ML}}\mathbb{E}\{y_i^2\} - b}{a}$$

$$= \frac{\frac{m_{ML}(ak+b)}{m_{ML}} - b}{a} = k,$$

the estimator $\widehat{k}_{ML}$ is an unbiased estimator.

2) The variance VAR of the ML estimator $\widehat{k}_{ML}$ is derived from the variance of $\widehat{k}_r$ and is given by,

$$\text{VAR}\{\widehat{k}_{ML}\} = \text{VAR}\{\widehat{k}_r\} + \text{VAR}\{k_\delta\} - 2\text{COV}\{\widehat{k}_r, k_\delta\}.$$

Since $\text{VAR}\{\widehat{k}_r\} = \frac{2(k\sigma_\phi^2\sigma_s^2 + \sigma_\vartheta^2)^2}{m_{ML}(\sigma_\phi^2\sigma_s^2)^2}$, $\text{VAR}\{k_\delta\} = \frac{1}{12}$, and $\text{COV}\{\widehat{k}_r, k_\delta\} \approx 0$,

$$\text{VAR}\{\widehat{k}_{ML}\} \approx \frac{2(k\sigma_\phi^2\sigma_s^2 + \sigma_\vartheta^2)^2}{m_{ML}(\sigma_\phi^2\sigma_s^2)^2} + \frac{1}{12}.$$

3) For the integer parameter $k$, the CRLB can be derived by computing either second forward or second backward difference for the log likelihood of Equation (12). Let $a = \sigma_\phi^2\sigma_s^2$ and $b = \sigma_\vartheta^2$. Then the joint probability distribution of $m_{ML}$ independent random variables $\{y_i\}_{i=1}^{m_{ML}}$ parameterised by $k$ is given as,

$$p_Y(y; k) = \frac{1}{(2\pi(ak+b))^{\frac{m_{ML}}{2}}} \exp\left[\frac{1}{2(ak+b)}\sum_{i=1}^{m_{ML}} y_i^2\right].$$

The second forward difference of log likelihood is given by,

$$\ln(p_Y(y; k+2)) - 2\ln(p_Y(y; k+1)) + \ln(p_Y(y; k))$$
$$= \frac{-m_{ML}}{2}\ln\left(\frac{(ak+b)(a(k+2)+b)}{(a(k+1)+b)^2}\right)$$
$$- \frac{1}{2}\sum_{i=1}^{m_{ML}} y_i^2\left(\frac{1}{ak+b} + \frac{-2}{a(k+1)+b} + \frac{1}{a(k+2)+b}\right)$$
$$\approx \frac{-m_{ML}}{2}\left(\frac{-a^2}{(a(k+1)+b)^2}\right)$$
$$- \frac{1}{2}\sum_{i=1}^{m_{ML}} y_i^2\left(\frac{2a^2}{(ak+b)(a(k+1)+b)(a(k+2)+b)}\right).$$

Using the second forward difference of log likelihood, CRLB=

$$\frac{-1}{\mathbb{E}\{\ln(p_Y(y; k+2)) - 2\ln(p_Y(y; k+1)) + \ln(p_Y(y; k))\}}$$
$$= \left(\frac{-m_{ML}}{2}\left(\frac{-a^2}{(a(k+1)+b)^2}\right) + \frac{1}{2}\sum_{i=1}^{m_{ML}}\mathbb{E}\{y_i^2\}\right.$$
$$\times \left.\left(\frac{2a^2}{(ak+b)(a(k+1)+b)(a(k+2)+b)}\right)\right)^{-1}$$
$$\approx \frac{2}{m_{ML}a^2}(ak+b)^2; \quad (k \approx k+1 \approx k+2$$
$$\text{when } k \text{ is large})$$
$$= \frac{2(k\sigma_\phi^2\sigma_s^2 + \sigma_\vartheta^2)^2}{m_{ML}(\sigma_\phi^2\sigma_s^2)^2}.$$

## REFERENCES

[1] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[2] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

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

[4] M. A. Davenport, M. F. Duarte, Y. C. Eldar, and G. Kutyniok, *Compressed Sensing: Theory and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[5] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. Basel, Switzerland: Birkhäuser, 2013.

[6] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.

[7] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc. B, Methodol.*, vol. 58, no. 1, pp. 267–288, 1996.

[8] E. Candes and T. Tao, "The Dantzig selector: Statistical estimation when *p* is much larger than *n*," *Ann. Statist.*, vol. 35, no. 6, pp. 2313–2351, 2007.

[9] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[10] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Commun. ACM*, vol. 53, no. 12, pp. 93–100, Dec. 2010.

[11] R. Arie, A. Brand, and S. Engelberg, "Compressive sensing and sub-Nyquist sampling," *IEEE Instrum. Meas. Mag.*, vol. 23, no. 2, pp. 94–101, Apr. 2020.

[12] Z. Gao, L. Dai, S. Han, C.-L. I, Z. Wang, and L. Hanzo, "Compressive sensing techniques for next-generation wireless communications," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 144–153, Jun. 2018.

[13] S. Lu, I. A. Hemadeh, M. El-Hajjar, and L. Hanzo, "Compressed-sensing-aided space-time frequency index modulation," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6259–6271, Jul. 2018.

[14] V. S. Amin, Y. D. Zhang, and B. Himed, "Sparsity-based time-frequency representation of FM signals with burst missing samples," *Signal Process.*, vol. 155, pp. 25–43, Feb. 2019.

[15] E. Sejdić, I. Orović, and S. Stanković, "Compressive sensing meets time–frequency: An overview of recent advances in time–frequency processing of sparse signals," *Digit. Signal Process.*, vol. 77, pp. 22–35, Jun. 2018.

[16] T. Wimalajeewa and P. K. Varshney, "Sparse signal detection with compressive measurements via partial support set estimation," *IEEE Trans. Signal Inf. Process. Over Netw.*, vol. 3, no. 1, pp. 46–60, Mar. 2017.

[17] N. Vaswani, "Kalman filtered compressed sensing," in *Proc. 15th IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 893–896.

[18] D. Angelosante, G. B. Giannakis, and E. Grossi, "Compressed sensing of time-varying signals," in *Proc. 16th Int. Conf. Digit. Signal Process.*, Jul. 2009, pp. 1–8.

[19] L. Gan, "Block compressed sensing of natural images," in *Proc. 15th Int. Conf. Digit. Signal Process.*, Jul. 2007, pp. 403–406.

[20] J. Ziniel and P. Schniter, "Dynamic compressive sensing of time-varying signals via approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5270–5284, Nov. 2013.

[21] S. Lan, Q. Zhang, X. Zhang, and Z. Guo, "Sparsity estimation in image compressive sensing," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2012, pp. 2669–2672.

[22] M. Lopes, "Estimating unknown sparsity in compressed sensing," in *Proc. ICML*, vol. 28, no. 3, 2013, pp. 217–225.

[23] M. E. Lopes, "Unknown sparsity in compressed sensing: Denoising and inference," *IEEE Trans. Inf. Theory*, vol. 62, no. 9, pp. 5145–5166, Sep. 2016.

[24] C. Ravazzi, S. Fosson, T. Bianchi, and E. Magli, "Sparsity estimation from compressive projections via sparse random matrices," *EURASIP J. Adv. Signal Process.*, vol. 2018, no. 1, pp. 1–18, Sep. 2018.

[25] V. Bioglio, T. Bianchi, and E. Magli, "On the fly estimation of the sparsity degree in compressed sensing using sparse sensing matrices," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 3801–3805.

[26] Y. Wang, Z. Tian, and C. Feng, "Sparsity order estimation and its application in compressive spectrum sensing for cognitive radios," *IEEE Trans. Wireless Commun.*, vol. 11, no. 6, pp. 2116–2125, Jun. 2012.

[27] S. K. Sharma, S. Chatzinotas, and B. Ottersten, "Compressive sparsity order estimation for wideband cognitive radio receiver," *IEEE Trans. Signal Process.*, vol. 62, no. 19, pp. 4984–4996, Oct. 2014.

[28] Z. Zhang, F. Liu, R. Du, C. Huang, and J. Sheng, "Trace-based sparsity order estimation with sparsely sampled random matrices," *IEEE Trans. Commun.*, vol. 64, no. 11, pp. 4790–4799, Nov. 2016.

[29] A. Ravelomantsoa, H. Rabah, and A. Rouane, "Compressed sensing: A simple deterministic measurement matrix and a fast recovery algorithm," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 12, pp. 3405–3413, Dec. 2015.

[30] D. Mitra, H. Zanddizari, and S. Rajan, "Investigation of Kronecker-based recovery of compressed ECG signal," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 6, pp. 3642–3653, Jun. 2020.

[31] S. Hanly. *Vibration Analysis on the Exterior of an Aircraft During Climb*. Accessed: Mar. 20, 2020. [Online]. Available: https://blog.mide.com/vibration-analysis-fft-psd-and-spectrogram

[32] L. H. Dicker, "Variance estimation in high-dimensional linear models," *Biometrika*, vol. 101, no. 2, pp. 269–284, Mar. 2014, doi: 10.1093/biomet/ast065.

[33] J. A. Tropp, J. N. Laska, M. F. Duarte, J. K. Romberg, and R. G. Baraniuk, "Beyond Nyquist: Efficient sampling of sparse bandlimited signals," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 520–544, Jan. 2010.

[34] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[35] N. Patwari. (Sep. 2007). *CRAWDAD Dataset Utah/Cir (V. 2007-09-10)*. [Online]. Available: https://crawdad.org/utah/CIR/20070910

**THIRUPPATHIRAJAN S.** received the Bachelor of Engineering degree in electronics and communication engineering from Manonmaniam Sundaranar University, Tirunelveli, Tamil Nadu, India, in 2001, and the Master of Engineering degree in VLSI design from Anna University, Chennai, India, in 2003. He is currently pursuing the Ph.D. degree in compressed sensing with the Department of Avionics, Indian Institute of Space Science and Technology, Thiruvananthapuram, India. Since 2003, he has been with the Avionics Entity, Indian Space Research Organization, Thiruvananthapuram, where he is engaged in the design and development of advanced signal and data processing systems for aerospace applications.

**LAKSHMI NARAYANAN R.** received the Bachelor of Engineering degree in electrical engineering from Madras University, Chennai, India, and the Ph.D. degree in electrical engineering from Indian Institute of Technology, Chennai, in 2010. From 2010 to 2011, he was working as a Communications Specialist with TATA ELXSI Ltd. Since November 2011, he has been with the Department of Avionics, Indian Institute of Space Science and Technology (IIST), Thiruvananthapuram, India, where he is currently an Associate Professor. His research interests include statistical signal processing and specifically in compressed sensing.

**SREELAL S.** (Senior Member, IEEE) received the Bachelor of Technology degree in applied electronics and instrumentation engineering from the University of Kerala, Thiruvananthapuram, India, in 1990, and the Master of Technology degree in integrated electronic circuits and systems from the Indian Institute of Technology, Chennai, India. He joined the Avionics Entity, Indian Space Research Organization, Thiruvananthapuram, in 1991, where he was engaged in the design and development of advanced, miniaturized telemetry systems for aerospace applications. From 1998 to 2001, he was a Research Scholar with the National University of Singapore, where his work on the design and simulation of short dimension, sub-100 nm MOSFETs led to the award of Ph.D. degree. He is currently pursuing the design and development of MEMS sensors, associated signal conditioning circuitry, advanced data acquisition systems, and application of state-of-the-art microelectronic devices for next generation avionics systems. He has published extensively in national and international conferences, journals, and books, and holds one U.S. and five Indian patents. He is a Life Member of the Electron Devices Society of IEEE and a member of the Distinguished Lecturer Panel formed by IEEE India Council for the National Distinguished Lecturer Program.

**MANOJ B. S.** (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from the Indian Institute of Technology, Chennai, in 2004. He was an Assistant Research Scientist and a Lecturer with the Electrical and Computer Engineering Department, University of California at San Diego. He is currently a Professor with the Department of Avionics, Indian Institute of Space Science and Technology, Thiruvananthapuram, India. His current research interests include complex networks, ad-hoc wireless networks, delay tolerant networks, next-generation wireless architectures, wireless sensor networks, wireless mesh networks, and signal processing over networks. He also serves as an Associate Editor for the IEEE Systems Journal and *Springer Nature Computer Science* journal. He is a fellow of IE(I) and IETE.

• • •