Compressive Sensing for Array Signal Processing

Anila Satheesh B., Department of ECE, Sree Narayana Gurukulam College of Engineering, Kerala

Deepa B, Naval Physical and Oceanographic Laboratory, Kerala

Subhadra Bhai, Naval Physical and Oceanographic Laboratory, Kerala

Anjana Devi S, Department of Electronics, Government Model Engineering College, Kerala

***Abstract* - Compressive sensing (CS) is an emerging area in which the conventional two step process of data acquisition and compression can be integrated into a single step. Compressive sensing exploits the sparsity of a signal and allows the digital signal to be reconstructed from far fewer measurements than the original size of the signal. This is possible as long as the measurements satisfy certain reasonable conditions such as Restricted Isometry Property, Incoherence, etc. The theory of compressed sensing can also be applied to the field of sensor array processing . In this paper, compressive sensing is applied to the problem of Direction of Arrival estimation. We perform Compressive Beamforming using two different approaches. In the time domain approach, CS can be applied to reduce the sampling rate of the Analog-to-Digital Converter, i.e., the number of samples received by each sensor of the array. In the spatial domain approach, CS can be applied to compress the array of large number of elements into an array of much smaller number of elements. Both these approaches are compared to the conventional beamforming technique and found to be close to the ideal impulse output. Compressed sensing recovery is performed using Subspace Pursuit (SP) and the two approaches are compared. The results obtained using SP is found to outperform the results obtained in the previous papers.**

Keywords: Compressive Sensing, Sparsity, Incoherence, Orthogonal Matching pursuit, Subspace Pursuit

I. INTRODUCTION

Compressive sensing (CS) is an emerging area that can replace the conventional approach of data acquisition based on the Shannon-Nyquist theorem. CS enables sparse or compressible signals to be captured and stored at a rate much below the Nyquist rate. The reconstruction of the original signal from its random projections is possible by means of an optimization process as long as the measurements satisfy reasonable conditions such as incoherence, Restricted Isometry Property etc.

An important feature of compressive sensing is that practical reconstruction can be performed by using efficient algorithms. Since the interest is in the vastly undersampled case, the linear system describing the measurements is underdetermined and therefore has infinitely many solutions. The key idea is that the sparsity helps in isolating the original vector. The first naive approach to a reconstruction algorithm consists in searching for the sparsest vector that is consistent with the linear measurements. This leads to the combinatorial ℓ_0-problem, which unfortunately is NP-hard in general. There are essentially two approaches for tractable alternative algorithms. The first is convex relaxation leading to ℓ_1-minimization, also known as basis pursuit, while the second constructs greedy algorithms. The iterative greedy algorithms received significant attention due to their low complexity and simple geometric interpretation. They include Orthogonal Matching Pursuit (OMP) [1], Regularized OMP (ROMP), etc. However their reconstruction capability was less than that of basis pursuit. This lead to the development of a new algorithm by Wei Dai and Olgica Milenkovic [2] titled 'Subspace Pursuit' (SP). The SP algorithm has provable reconstruction quality comparable to that of basis pursuit and also has low complexity comparable to that of OMP technique for very sparse signals.

The theory of Compressive Sensing can be applied to the field of array processing as well. Estimation of the Direction of Arrival (DOA) of propagating plane waves is studied extensively in many research areas such as radar and sonar systems, medical imaging, acoustic signal processing, and seismology. The aim is to estimate signal arriving from a desired direction in the presence of noise and other interferences. Instead of using a single physical antenna, an array of sensor elements is used for the DOA estimation. Compressive sampling can be applied to sensor array processing field by exploiting the sparsity in angle spectrum. This can be done in two ways: either by taking compressive measurements of the incoming signal at each sensor element in the array [3] or by compressing the array of a large number of elements to an array of a much smaller number of elements [4]. In this paper, both the above discussed methods are implemented and simulation results are compared. The DOA estimation problem is solved using Subspace Pursuit algorithm which was found to give better results than the algorithms used in [3] and [4]. The advantage of using SP algorithm is that it is not based on any constraint to reach a global optimum and also it can reach an acceptable solution with lesser number of iterations when compared with the FOCUSS algorithm used in [4]. The results obtained in this paper are found to work for lower SNR cases than which used in [3] and [4].

This paper compares the two approaches discussed above with the help of numerical simulation results. The paper is organized as follows. Section II gives an introduction to the compressive sensing problem. Section III deals with the two approaches implemented for the case of a Uniform Linear Array. Section IV gives an overview on the reconstruction algorithm used. Numerical Simulation results are given Section V. Section VI concludes the work.

II. PRELIMINARIES

II.a *Direction Of Arrival Estimation Using Sensor Array*

Consider a sensor array having L sensor elements. Let there be K sources at angles $\theta_k; k = 1, 2, ... K$. Let the sources be at far-field such that the spherical wavefront of the sources becomes planar at the array. The received signal at the array can be written as :

$$x(n) = \sum_{k=1}^{K} a(\theta_k)\, s_k(n) + w(n) \qquad (2.1)$$

where x(n) is an LxN matrix, $a(\theta_k)$ is the array steering vector of size Lx1 corresponding to a source from direction θ_k, $s_k(n)$ is a planar wavefront of size 1xN impinging on the array, where N is the number of samples, and $w(n)$ is the additive noise present at each sensor and is a matrix of size LxN. The array steering vector can be expressed as:

$$a(\theta) = \left[1\ e^{-j\alpha} \cdots e^{-j\alpha(L-1)}\right]^T \qquad (2.2)$$

where $\alpha = 2\pi(d/\lambda)\sin\theta$ is the phase shift along the uniform linear array from element to element and λ is the wavelength of the input signal. The equation (2.1) can also be expressed in Matrix-Vector-Multiplication form as:

$$\mathbf{x} = \mathbf{A}\,\mathbf{s} + \mathbf{w} \qquad (2.3)$$

In (2.3), **A** is an LxK matrix of steering response vectors given by $A = [a(\theta_1)\, a(\theta_2) \ldots a(\theta_K)]$ and ***s*** is a KxN matrix corresponding to the K independent sources.

II.b *Introduction To Compressive Sensing*

The CS framework was introduced by Candes et al. in [5][6][7] and Donoho in [8]. CS theory enables sparse or compressible signals to be captured and stored at a rate much below the Nyquist rate. As long as the measurements satisfy certain conditions (such as incoherence, RIP), the recovery of the original signal from its randomized projections can be made possible by means of an optimization process.

The theory of Compressed Sensing is based on two fundamental principles: Sparsity and Incoherence. A signal is said to be sparse or compressible if it has a concise representation when expressed in a proper basis. Consider a real-valued, finite-length, one-dimensional, discrete-time signal x, which can be viewed as an Nx1 column vector in R^N. Any signal in R^N can be represented in terms of a basis of Nx1 vectors as [5]:

$$x = \sum_{i=1}^{N} s_i \psi_i = \psi \mathbf{s} \qquad (2.4)$$

where ***s*** in the Nx1 column vector. The signal ***s*** is said to be K-sparse if it is a linear combination of only K basis vectors; that is, only K of the coefficients of ***s*** in (2.4) are zero and (N-K) are non-zero. Incoherence extends the duality between time and frequency and expresses the idea that objects having a sparse representation in ψ must be spread out in the domain in which they are acquired. A signal can be perfectly reconstructed from its compressed samples only if the sparsifying basis and measurement basis have small mutual coherence, or they are incoherent. The mutual coherence between two bases Φ and ψ can be expressed as:

$$\mu(\phi,\psi) = \sqrt{N} \max_{k \le K, j \le N} \phi_k^T \psi_j \qquad (2.5)$$

That is, each column (row) of the measurement domain Φ must be spread out in the ψ domain and vice-versa.

Compressive Sensing requires the original Nx1 length signal *x* (sparse in domain ψ) to be multiplied by a measurement matrix Φ to obtain its projections of length M (M≪N). The Mx1 measurement vector is given by:

$$y = \phi x = \phi \psi s = \Theta s \qquad (2.6)$$

where *Φ* is the MxN measurement matrix (M<<N). The choice of measurement matrix is important since it decides the stability and reliability of the compressive sensing process. For good recovery, the measurement matrix must be incoherent with the sparsity basis and also it should satisfy the Restricted Isometry property. It has been proved in [9] that both these conditions are satisfied if the measurement matrix is selected as a random matrix. Several choices for random matrices are given in [9].

Reconstruction of the N-length signal *x* from M measurements is an ill-posed problem. But since the original signal is sparse, the recovery can be cast as an optimization problem where the objective is to maximize an appropriate measure of sparsity while simultaneously satisfying the constraints y=Θs. This can be expressed mathematically as

$$\hat{s} = \arg\min_{s} \|s\|_0 \quad \textit{subject to } y = \Theta s \qquad (2.7)$$

where $\|s\|_0$ is the ℓ_0 norm of s. In this paper, an alternate algorithm called 'Subspace Pursuit' is used for sparse signal recovery. Results obtained with Subspace Pursuit algorithm is found to have good reconstruction capability but with much lesser complexity. The Subspace Pursuit algorithm is explained in Section IV.

III. COMPRESSIVE SENSING ARRAY PROCESSING

The theory of Compressive Sensing can be applied to the field of array processing as well. The aim is to extract useful characteristics of the received signal, for example its direction. For DOA estimation problems, the angle spectrum found to be very sparse. Since Subspace Pursuit is proved to work well for very sparse signals, this motivated us to employ this algorithm for the estimation problem. As explained in section I, DOA estimation using CS can be performed in two ways. This is explained in the following section.

III.a *Time Domain Approach*

The DOA estimation using compressive sampling in time domain is implemented by reducing the number of time samples received by each sensor element in the array. In this approach, we take only random projections of the received signals at the sensors. In passive sensing problems, where the source signals are unknown, one of the sensors (called reference sensor) acquire the signal at the Nyquist rate whereas the rest of the sensors take only the random projections. The sparsity pattern vector is reconstructed from the optimization problem which gives the number of sources and their directions. The sparsity basis in passive sensing problems can be constructed using proper shifts of the reference signal for each angle π_j from the set of angles B= $\{\pi_1, \pi_2, \ldots \pi_R\}$ where R determines the resolution. In case of active sensing problems, where the source signals are known, all sensors acquire random projections of the incoming data. The sparsity basis is constructed using proper shifts of the known source signals.

Let the sensor array consists of L sensors. The sparsity pattern vector *s* can be related to the received signal at the i^{th} sensor, ψ_i as follows:

$$x_i = \psi_i s \qquad (3.1)$$

where x_i is an Nx1 input signal vector as defined in section II.a, ψ_i is the sparsity basis for the i^{th} sensor and is a matrix of size NxR. The delay at the i^{th} sensor at angle π_s for a ULA is given by:

$$\Delta_i(\pi_S) = (i-1)\frac{d}{c}\sin(\pi_S) \qquad (3.2)$$

The compressed measurements can be written for i^{th} sensor as:

$$\beta_i = \phi_i x_i = \phi_i \psi_i s \qquad (3.3)$$

where β_i is an Mx1 vector corresponding to the compressed measurements at the i^{th} sensor , M being the number of compressed measurements taken and $M \ll N$, Φ_i is the random [7] measurement matrix of size MxN. The DOA estimation problem is now recast as a Compressed Sensing problem whose objective is to find the sparsity pattern vector *s* in (3.3). The sparse signal reconstruction is performed for each sensor using Subspace Pursuit Algorithm explained in section IV. The sparsity pattern vectors, $\boldsymbol{s_i}$ obtained for each sensor are then added together to form the final estimated output. The results obtained are given in section V.

III.b *Spatial Domain Approach*

In the spatial domain approach, a sensor array of large number of elements is compressed or transformed into an array of much smaller number of elements. The DOA estimation problem is formulated in terms of a compressed sensing problem and sparse vector is recovered using the Subspace Pursuit algorithm. The relation between sparsity pattern vector and the incoming signal is the same as given in (3.1). The sparsifying basis, also called angle scanning matrix in this case, is constructed by finding out the steering vectors for each angle π_j from the set of angles B = $\{\pi_1, \ldots \pi_R\}$ where R determines the resolution. The steering vector at angle π_S is given as:

$$a(\pi_S) = \left[1 \;\; e^{-j\alpha} \;\; e^{-2j\alpha} \cdots e^{-j\alpha(L-1)}\right] \qquad (3.4)$$

where L is the number of sensors, α is the delay from element to element in the sensor array and given by:

$$\alpha = 2\pi(d/\lambda)\sin\pi_S \qquad (3.5)$$

where d is the distance between the sensor elements and λ is the wavelength of the received signal. The angle scanning matrix can be expressed as:

$$\psi = [a(\pi_1) \;\; a(\pi_2) \cdots a(\pi_N)] \qquad (3.6)$$

The compressed array signal can be expressed as

$$\beta = \phi x = \phi \psi s \qquad (3.7)$$

Recovery of the sparsity pattern vector *s* from (3.7) forms the compressed sensing problem and is recovered using Subspace Pursuit algorithm. The results obtained are given in section V.

IV. SUBSPACE PURSUIT ALGORITHM

As seen in section III, the problem of DOA estimation in sensor array using CS requires determining the sparsity pattern vector *s* from the compressed samples β_i at the i^{th} sensor. From the principles of CS, this problem of DOA estimation can be formulated as a compressed sensing problem and can be solved using an optimization problem of the form:

$$\hat{s} = \arg\min_b \|s\|_p^p \quad \textit{subject to } \Theta_i s = \beta_i \qquad (4.1)$$

where $\Theta_i = \Phi_i \psi_i$. The term $\|s\|_p^p$ is called the p^{th} norm of $\boldsymbol{s}$ and p>0 . A least squares solution is obtained with p=2 but it does not give the sparsest solution. That is, ℓ_2 norm measures only the signal energy and not signal sparsity. An alternate method is to use the ℓ_0 norm. The ℓ_0 norm counts the number of non-zero samples in *s* and therefore can result in the sparsest solution. However, solving (4.1) using ℓ_0 is both numerically unstable and NP hard. The work by Donoho and Candes [5][10] demonstrated that CS reconstruction is a polynomial time problem albeit under the constraint that more than 2K measurements are used where K is the sparsity of the signal. The key observation behind these findings is that it is not necessary to resort to ℓ_0 optimization to recover the signal from the underdetermined inverse problem; a much easier ℓ_1 optimization, based on linear programming (LP) techniques, yields an equivalent solution, if the sampling matrix satisfies the restricted isometry property (RIP). The ℓ_1 optimization problem can be recast as a Linear Programming (LP) problem which can efficiently solved using the algorithm Basis Pursuit [10]. Even though ℓ_1 has high reconstruction accuracy, their complexity is impractical for several applications. For a signal of length N, the complexity of ℓ_1 optimization is $O(N^3)$.

Several classes of low complexity reconstruction techniques were later put forward as alternatives to LP-based recovery [11]. These include the greedy algorithms including Orthogonal Matching Pursuit (OMP). OMP computes the non-zero values in the signal in an iterative fashion. The basic idea behind these methods is to find the support of the unknown signal sequentially. At each iteration of the algorithm, one or several coordinates of the vector *s* are selected for testing based on the correlation values between the columns of the measurement basis Φ and the regularized measurement vector. If deemed sufficiently reliable, the candidate column indices are subsequently added to the current estimate of the support set of *s*. The OMP algorithm iterates this procedure until all the coordinates in the correct support set are included in the estimated support set. The complexity of such algorithms is significantly smaller than that of LP methods especially when the signal sparsity level K is small. However, its guarantees are not as strong as that of Basis Pursuit [11].

The main contribution of this paper is the application of Subspace Pursuit algorithm for CS signal recovery in the array processing area. The Subspace Pursuit algorithm starts by selecting K columns of the matrix Θ that contain the measured signal. This is done by selecting the K columns of Θ that has the highest correlation with the measurement vector β. If the distance of the received vector to the selected subspace is deemed large, then the list is refined by retaining the reliable candidates and discarding the unreliable ones while adding the same number of new candidates.

The main steps of the SP algorithm are summarized as below [2]:

- Input : K , Θ , β
- Initialization :
  1. T^0 = { K indices corresponding to the largest magnitude entries in $\Theta^*\beta$ }
  2. β_r^0= residue ($\beta, \Theta_{\hat{T}^0}$)
- Iteration : At the l^{th} iteration , go through the following steps
  1. $\hat{T}^l = T^{l-1} \cup$ {K indices corresponding to the largest magnitude entries in $\Theta^*\beta_r^{l-1}$ }
  2. Set $s_p = \Theta_{\hat{T}^l}^{+}\beta$
  3. T^l = { K indices corresponding to the largest magnitude elements of s_p}
  4. $\beta_r^l = \text{residue}\left(\beta, \Theta_{T^l}\right)$
  5. If $\left\|\beta_r^l\right\|_2 > \left\|\beta_r^{l-1}\right\|_2$, let $T^l = T^{l-1}$ and quit the iteration.
- Output: The estimated signal $\hat{s}$, satisfying $s_{\{1,..,N\}-T^l} = 0$ and $\hat{s}_{T^l} = \Theta_{T^l}^{+}\beta$

The residue of a vector is given by :

$$\beta_r = residue(\beta, \Theta_I) = \beta - \beta_p$$

where β_p is the projection of β onto span (Θ_I) and is given by:

$$\beta_p = proj(\beta, \Theta_I) = \Theta_I \Theta_I^{+}\beta$$

where $\Theta_I^{+} = \left(\Theta_I^*\Theta_I\right)^{-1}\Theta_I^*$ is the pseudo-inverse of the matrix Θ_I.

In compressed sensing recovery, the major challenge lies in identifying the correct subspace in which the measured signal lies. The subspace is to be chosen with not more than K columns of the matrix Θ. Once the correct subspace is identified, the sparse vector is obtained by applying the pseudo-inversion process.

V. SIMULATION RESULTS

The principal focus of this paper has been to employ the subspace pursuit algorithm for DOA estimation using CS. The algorithm has been employed for two different cases and the simulation results are compared.

In the first case, DOA estimation in time domain is implemented in MATLAB. Two broadband source signals are taken and are placed at the far-field of a ULA with 36 sensors placed on the x-axis with 0.25m spacing between the sensor elements. The targets are placed at 32^0 and 27^0 respectively. The two signals are assumed to be unknown. The number of snapshots taken is 50 and total length of the signal is taken to be 5000 which corresponds to 0.16 seconds of data. Out of this, only M = 13 (≈ K log (N)) random samples are taken by each sensor which makes a total of 468 samples compared to the 100*36 samples that would have been processed per block if conventional sampling was used. Therefore, the total number of samples taken is much less than the total time samples of the signals. Each block is processed independently and the outputs of each block are exponentially averaged using the equation:

$$s_{avg}(n) = \tau s(n) + (1-\tau)s(n-1) \qquad (5.1)$$

where s(n) is the sparse signal obtained and τ is the time constant which is found out as:

$$\tau = \frac{1}{1+\left(T_c / T_{in}\right)} \qquad (5.2)$$

where T_C is the total integration time, which is 0.16s and T_{in} is the time period of each block which is 3.2ms.

The entries of the random measurement matrices for each sensor is drawn randomly from N(0, 1) independently. WGN is added to the compressive measurements with signal-to-noise ratio (SNR) equals 10 dB. Figure 5.1 shows the DOA estimation in time domain output. The figure also shows the conventional time domain beamformer output with 36 sensors.

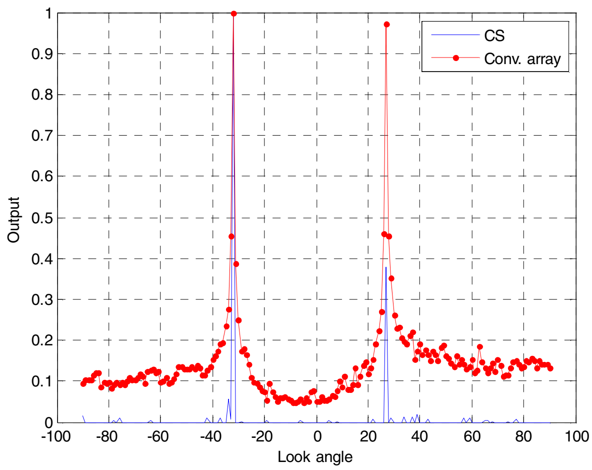

Figure 5.1 DOA estimation in time domain for SNR = 10dB

The experiment is repeated for targets in the same beamwidth of a sensor element. The beamwidth for a 36-element array is around 3^0. Two audio sources are kept 3^0 apart at 32^0 and 35^0 respectively. The sources are train.mat and laughter.mat which are available in MATLAB. With compressive sensing array, the two sources could be resolved perfectly. Figure 5.2 shows the results.

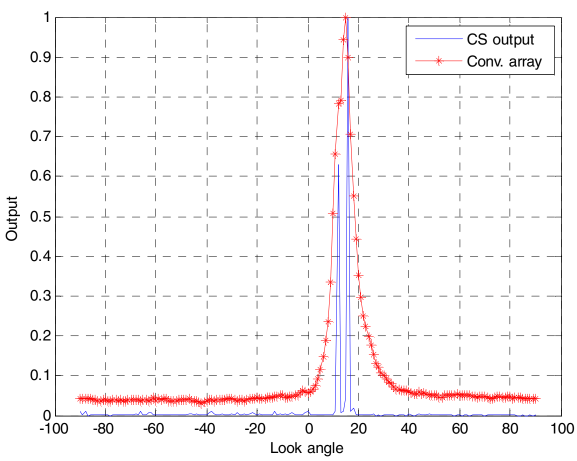

Figure 5.2 CS output in time domain for targets in the same beamwidth

In the second case, DOA estimation in spatial domain is implemented in MATLAB. The number of compressed measurements in spatial domain depends on the number of sensors employed. This differentiates the spatial domain method from time domain method, where, the number of compressed measurements depends on the block length. As in time domain case, two sources are kept at 35^0 and 24^0 respectively. The number of sensors taken is 36. The number of compressed sensors is found to be 8 ($\approx$ K logL). The number of snapshots taken is 50. Total length of the signal is 5000 which corresponds to 0.16 seconds of data. As in time domain case, estimation is performed for each block and the outputs are exponentially averaged as in (5.1). SNR is fixed at 10dB. Figure 5.3 shows the output for DOA estimation in spatial domain. The plot also compares the output with conventional beamforming technique using 36 sensors as well as 8 sensors. The compression ratio in the spatial domain case is higher than that of time domain method and is $\log_N L$.

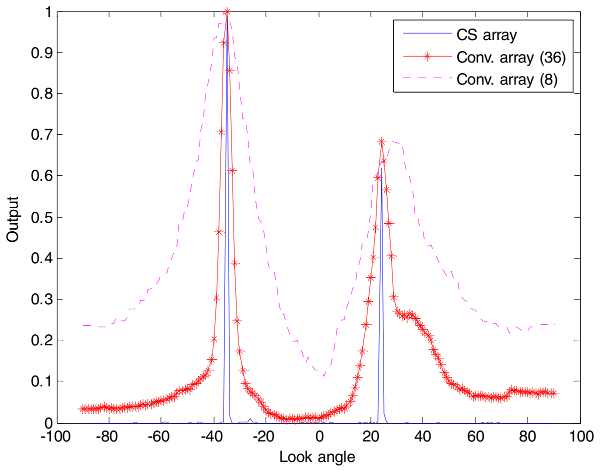

Figure 5.3 DOA estimation in spatial domain for SNR = 10dB

The experiment is repeated for targets in the same beamwidth of a sensor element. Two audio sources are kept 2^0 apart at 0^0 and 2^0 respectively. As in time domain case, the two sources taken are train.mat and laughter.mat which are available in MATLAB. With compressive sensing array, the two sources could be resolved perfectly. Figure 5.4 shows the results.

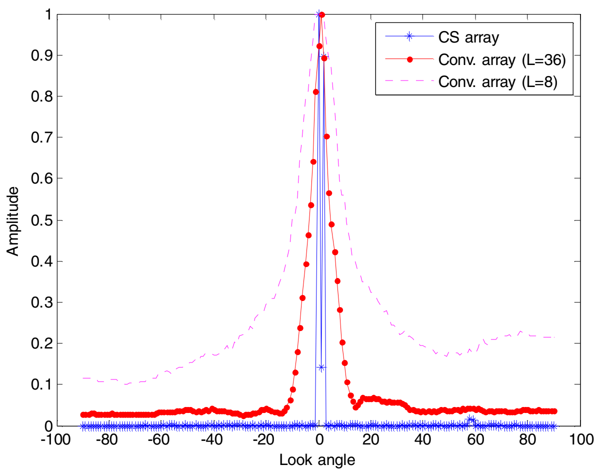

Figure 5.4 CS output in spatial domain for targets in the same beamwidth

Our DOA estimation problem using SP algorithm is also found to work for lower SNR cases as compared to the papers [3] and [4]. Figure 5.5 shows the two outputs for 0dB SNR.

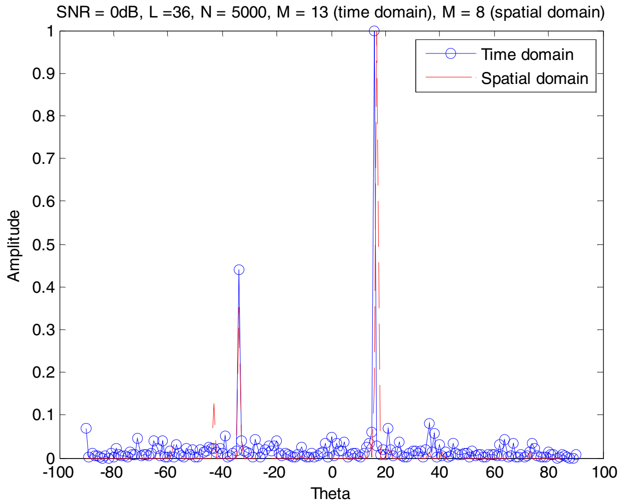

Figure 5.5 Outputs at 0dB SNR

In order to compare the two DOA estimation methods, each experiment is repeated 100 times with different values of sparsity K. Here, K refers to the number of sources. The sensor array consists of 36 sensors and the number of compressed measurements is 8. SNR value is fixed at 10dB. Figure 5.6 depicts the simulation results. The numerical values on x-axis denote the sparsity level K, and the numerical values on the y-axis denote the fraction of exactly recovered signals.

Thus we see that as the sparsity (number of sources) increases, the fraction of exactly recovered signals reduces. Or in other words, when the number of sources is less, DOA estimation gives comparatively good results. Also, it can be concluded from the plot that the probability of exact recovery is greater for time domain method than spatial domain method. But the software complexity of spatial domain method is much less than that of time domain method as the matrices involved in spatial domain is of lesser size than those used in time domain method.

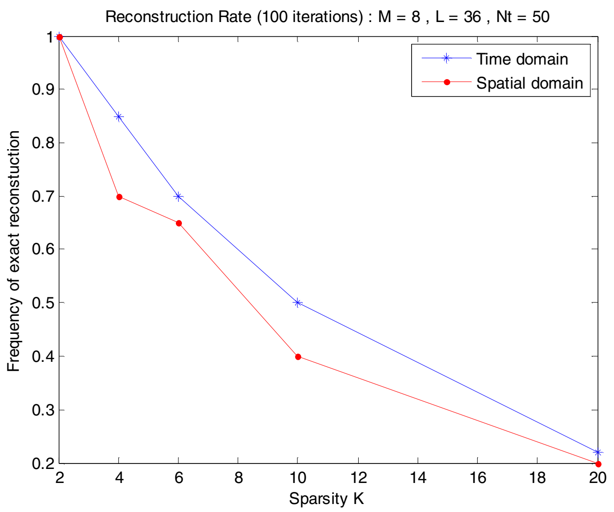

Figure 5.6 Simulation results for exact recovery: varying the sparsity level

VI. CONCLUSION

In this paper, we have implemented the compressive beamforming technique for Direction of Arrival Estimation problem. The theory of CS can be applied in two ways: in time domain and in spatial domain. Performances of both approaches are evaluated in the presence of uncorrelated noise. Our simulation results show that DOA estimation using Compressive sensing, which uses lesser number of samples, gives better resolution than the conventional method. This in turn gives the advantage that with fewer measurements we are getting the performance of a larger sized array. The recovery algorithm implemented is Subspace Pursuit algorithm which has high reconstruction quality and low complexity and is found to give good results in array processing applications also.

REFERENCES

[1] J. A. Tropp and A. C. Gilbert, "Signal Recovery from Random Measurements via Orthogonal Matching Pursuit," *IEEE Trans. Info. Theory*, vol. 53, no. 12, Dec. 2007, pp. 4655-4666

[2] Wei Dai, and Olgica Milenkovic, "Subspace Pursuit for Compressive Sensing Signal Reconstruction," *IEEE Trans. on Info. Theory*, Vol. 55, No. 5,pp 2230-2249, May 2009

[3] Gurbuz, A.C., James H McClellan, and Volkan Cevher, "A Compressive Beamforming Method," in *ICASSP*, 2008,pp 2617-2620.

[4] Ying Wang, Geert Leus, Ashish Pandharipande, " Direction Estimation Using Compressive Sampling Array Processing", in *IEEE/SP 15th workshop on SSP*, pp 626-629, 2009.

[5] Candes,E.J., and T. Tao," Decoding by linear programming," *IEEE Trans. Info. Theory*, vol. 51, no. 12, pp. 4203-4215, Dec. 2005.

[6] Candes,E.J., J.Romberg, and T.Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information, " *IEEE Trans. Info. Theory*, vol. 52, no. 2, pp 489-509, February 2006.

[7] Candes,E.J., Wakin,M.B., "An Introduction To Compressive Sampling," *IEEE Signal Processing Magazine*, vol.25, no.2, pp.21-30, March 2008

[8] Donoho, D.L., "Compressed sensing, " *IEEE Trans. Info. Theory*, vol. 52, no.4, pp 1289-1306, April 2006

[9] Candes, E.J., "Compressive sampling," Int. Congress of Mathematics , No. 3, pp. 1433-1452, Madrid, Spain, 2006

[10] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," SIAM Rev., vol. 43, no. 1, pp. 129159, 2001.

[11] Deanna Needell, "Topics In Compressed Sensing," University Of California,2009

[12] Johnson, D. H., and D. E. Dudgeon, "Array Signal Processing: Concepts and Techniques," Prentice Hall, 1993.

[13] Van Trees, H.L., "Optimum Array Processing," Detection, Estimation and Modulation Theory ,Part IV, John Wiley and Sons, New York,2002