Adaptive Design for Distributed MIMO Radar Using Sparse Modeling*

Sandeep Gogineni, *Student Member, IEEE* and Arye Nehorai* , *Fellow, IEEE*
Department of Electrical and Systems Engineering
Washington University in St. Louis
One Brookings Drive, St. Louis, MO 63130, USA
Email: {sgogineni,nehorai}@ese.wustl.edu
Phone: 314-935-7520 Fax: 314-935-7500

***Abstract*—Multiple Input Multiple Output (MIMO) radar systems with widely separated antennas provide spatial diversity gain by viewing the targets from different angles. In this paper, we propose an approach to accurately estimate the properties (position, velocity) of multiple targets using such systems by employing sparse modeling. We also propose a new metric to analyze the performance of the radar system. We develop an adaptive mechanism for optimal energy allocation at different transmitters. We show that this adaptive mechanism outperforms MIMO radar systems that transmit fixed equal energy across all the antennas.**

***Index Terms*—Adaptive, Multiple Input Multiple Output (MIMO) radar, sparse modeling, widely separated antennas, multiple targets**

I. Introduction

Multiple Input Multiple Output (MIMO) radar [1]–[5] has attracted a lot of attention in recent times due to the improvement in performance it offers over conventional single antenna systems. MIMO radar is typically used in two configurations, namely distributed (widely separated) and colocated. In distributed MIMO radar [1]–[3], the antennas are widely separated. This enables viewing the target from different angles. Hence, if the target returns between a particular transmitter and receiver are weak, then it is highly likely that they will be compensated by the returns between other antenna pairs. While distributed MIMO radar exploits spatial diversity, colocated MIMO radar [4] exploits the waveform diversity. In this configuration, all the antennas are closely spaced and hence the target Radar Cross Section (RCS) values are the same for all transmitter-receiver pairs. In this paper, we study MIMO radar with widely separated antennas in the context of sparse modeling for estimating the positions and velocities of multiple targets.

Since the number of targets in a radar scene is limited, we can use sparse modeling to represent the radar data. Sparse modeling and compressive sensing have been applied to the field of radar [6]–[8]. So far, it has been used for MIMO radar only in the context of closely spaced antennas [7]–[9]. To the best of our knowledge, the important configuration of distributed MIMO radar has not been approached from a sparse modeling perspective. This configuration is very important since it provides spatial diversity. In [8], the authors call their system distributed MIMO radar even though they are actually using colocated MIMO radar. This is evident from the fact that they use the same RCS value for all transmitter-receiver pairs. In this paper, we deal with this important aspect.

Adaptive radar design has been a hot topic for a number of years. Implementing the radar system in a closed loop by adaptively choosing the properties of the transmitted waveforms based on the knowledge of the environment gives significant improvement in performance. References [3], [10], [11] demonstrate the advantages of adaptive design under different configurations. In [10], adaptive polarization design is considered in the context of Single Input single Output (SISO) radar systems.References [3], [11] discuss the problem of adaptive waveform design for MIMO radar. In this paper (see also [12]), we propose an optimal adaptive energy allocation mechanism for distributed MIMO radar using the reconstructed sparse vectors. We also introduce a new metric to analyze the performance of the radar system. Using this metric, we show that this adaptive mechanism outperforms MIMO radar systems that transmit fixed equal energy across all the antennas.

In section II, we derive the signal model for MIMO radar using sparse representation. In section III, we present different algorithms for sparse support recovery to infer about the targets (position, velocity). In section IV, we propose the optimal energy allocation mechanism. In section V, we use numerical results to show the performance of the sparse recovery algorithms in estimating the target positions and velocities. We demonstrate the improvement offered by adaptive energy allocation. In section VI, we conclude this paper.

II. Signal Model

We assume that there are M_{T} transmitters, M_{R} receivers, and K targets. Further, we assume that all the targets are moving in a 2 dimensional plane. This can be extended without loss of generality to the 3 dimensional case. The k^{th} target is located at $\overrightarrow{\boldsymbol{p}^k} = \left[p_x^k, p_y^k\right]$ and it moves with a velocity $\overrightarrow{\boldsymbol{v}^k} = \left[v_x^k, v_y^k\right]$. The i^{th} transmitter and j^{th} receiver are located

⋆This work was supported by the Department of Defense under the Air Force Office of Scientific Research MURI Grant FA9550-05-1-0443, and ONR Grant N000140810849.

∗Corresponding author

at $\overrightarrow{\boldsymbol{t}_i} = \left[t_{i_x}, t_{i_y}\right]$ and $\overrightarrow{\boldsymbol{r}_j} = \left[r_{j_x}, r_{j_y}\right]$, respectively. We transmit orthonormal waveforms from the different transmitters. Let $w_i(t)$ be the baseband waveform transmitted from the i^{th} transmitter. Further, we assume that the cross correlations between these waveforms is close to zero for different delays [1]–[3]. Define $a_{ij}^k(t)$ as the RCS of the k^{th} target between the i^{th} transmitter and the j^{th} receiver. Then, the bandpass signal arriving at the j^{th} receiver can be expressed as

$$\begin{aligned} y_j(t) \;\; = \;\; \mathrm{Re}\Bigg\{ & \sum_{k=1}^{K}\sum_{i=1}^{M_{\mathrm{T}}} a_{ij}^k(t) w_i\left(t-\tau_{ij}^k\right) \times \\ & e^{j2\pi\left(f_{\mathrm{D}_{ij}}^k\left(t-\tau_{ij}^k\right)+f_{\mathrm{c}}\left(t-\tau_{ij}^k\right)\right)}\Bigg\}, \end{aligned}$$

where f_{c} is the carrier frequency, τ_{ij}^k and $f_{\mathrm{D}_{ij}}^k$ are the delay and Doppler shift corresponding to the k^{th} target.

$$\tau_{ij}^k = \frac{1}{c}\left(\|\overrightarrow{\boldsymbol{p}^k} - \overrightarrow{\boldsymbol{t}_i}\| + \|\overrightarrow{\boldsymbol{p}^k} - \overrightarrow{\boldsymbol{r}_j}\|\right), \tag{1}$$

$$f_{\mathrm{D}_{ij}}^k = \frac{f_{\mathrm{c}}}{c}\left(\langle\overrightarrow{\boldsymbol{v}^k}, \overrightarrow{\boldsymbol{u}^k}_{\boldsymbol{r}_j}\rangle - \langle\overrightarrow{\boldsymbol{v}^k}, \overrightarrow{\boldsymbol{u}^k}_{\boldsymbol{t}_i}\rangle\right), \tag{2}$$

where $\overrightarrow{\boldsymbol{u}^k}_{\boldsymbol{t}_i}, \overrightarrow{\boldsymbol{u}^k}_{\boldsymbol{r}_j}$ denote the unit vector from the i^{th} transmitter to the k^{th} target and the unit vector from the target to the k^{th} receiver, respectively; $\langle,\rangle$ is the inner product operator, and c is the speed of propagation of the wave in the medium.

We define the target state vector $\boldsymbol{\zeta} = \left[p_x, p_y, v_x, v_y\right]^T$. The goal is to estimate $\boldsymbol{\zeta}$ for all the K targets. The received signals at each receiver are first down converted from the radio frequency and then passed through a bank of M_{T} matched filters, each of which corresponds to a particular transmitter. Under the orthogonality assumption [1]–[3], the sampled outputs of the i^{th} matched filter at the j^{th} receiver is given as

$$y_{ij}(n) = \sum_{k\in\mathcal{K}} a_{ij}^k(n) e^{j2\pi\left(f_{\mathrm{D}_{ij}}^k\left(n-\tau_{ij}^k\right)-f_{\mathrm{c}}\tau_{ij}^k\right)} + e_{ij}(n), \tag{3}$$

where $e_{ij}(n)$ is the additive noise at the output of the i^{th} matched filter of the j^{th} receiver, $\mathcal{K}$ represents a set containing all the targets that contribute to the matched filter output at time n. In order to arrive at the above expression, we also assume that the target RCS values do not vary within a pulse duration and the Doppler shift is small. Hence, $a_{ij}^k(t)e^{j2\pi f_{\mathrm{D}_{ij}}t}$ varies slowly when compared with $w_i(t)$. Now, we discretize the target state space into a grid of L values $\left\{\boldsymbol{\zeta}^l, \forall l = 1,\ldots,L\right\}$. Hence, each of the targets is associated with a state vector belonging to this grid. If the presence of a target at $\boldsymbol{\zeta}^l$ contributes to the matched filter output at n, define $\psi_{ij}^l(n) = e^{j2\pi\left(f_{\mathrm{D}_{ij}}^l\left(n-\tau_{ij}^l\right)-f_{\mathrm{c}}\tau_{ij}^l\right)}$, else $\psi_{ij}^l(n) = 0$. Also, if $\boldsymbol{\zeta}^l$ is the state vector of the k^{th} target, we define $s_{ij}^l(n) = a_{ij}^k(n)$, else $s_{ij}^l(n) = 0$. For each j, we stack $s_{ij}^l(n)$, $y_{ij}(n)$, and $e_{ij}(n)$ corresponding to different transmitters to obtain M_{T} dimensional column vectors $\boldsymbol{s}_j^l(n)$, $\boldsymbol{y}_j(n)$ and $\boldsymbol{e}_j(n)$, respectively. Similarly, we arrange $\psi_{ij}^l(n)$ into $(M_{\mathrm{T}})\times(M_{\mathrm{T}})$ dimensional diagonal matrix $\boldsymbol{\Psi}_j^l(n)$.

$$\begin{aligned} \boldsymbol{s}_j^l(n) &= \left[s_{1j}^l(n),\ldots,s_{M_{\mathrm{T}}j}^l(n)\right]^T, && (4)\\ \boldsymbol{y}_j(n) &= \left[y_{1j}(n),\ldots,y_{M_{\mathrm{T}}j}(n)\right]^T, && (5)\\ \boldsymbol{e}_j(n) &= \left[e_{1j}(n),\ldots,e_{M_{\mathrm{T}}j}(n)\right]^T, && (6)\\ \boldsymbol{\Psi}_j^l(n) &= \mathrm{diag}\left\{\psi_{1j}^l(n),\ldots,\psi_{M_{\mathrm{T}}j}^l(n)\right\}, && (7) \end{aligned}$$

where $\mathrm{diag}\{\cdot\}$ refers to a diagonal matrix whose entries are given by $\{\cdot\}$ and $[\cdot]^T$ denotes the transpose of $[\cdot]$. Further, we arrange $\left\{\boldsymbol{s}_j^l(n)\right\}_{j=1}^{M_{\mathrm{R}}}$, $\left\{\boldsymbol{y}_j(n)\right\}_{j=1}^{M_{\mathrm{R}}}$, and $\left\{\boldsymbol{e}_j(n)\right\}_{j=1}^{M_{\mathrm{R}}}$into $M_{\mathrm{T}}M_{\mathrm{R}}$ dimensional column vectors $\boldsymbol{s}^l(n)$, $\boldsymbol{y}(n)$, and $\boldsymbol{e}(n)$, respectively and $\left\{\boldsymbol{\Psi}_j^l(n)\right\}_{j=1}^{M_{\mathrm{R}}}$, into $(M_{\mathrm{T}}M_{\mathrm{R}})\times(M_{\mathrm{T}}M_{\mathrm{R}})$ dimensional diagonal matrix $\boldsymbol{\Psi}^l(n)$.

$$\begin{aligned} \boldsymbol{s}^l(n) &= \left[\left(\boldsymbol{s}_1^l(n)\right)^T,\ldots,\left(\boldsymbol{s}_{M_{\mathrm{R}}}^l(n)\right)^T\right]^T, && (8)\\ \boldsymbol{y}(n) &= \left[\left(\boldsymbol{y}_1(n)\right)^T,\ldots,\left(\boldsymbol{y}_{M_{\mathrm{R}}}(n)\right)^T\right]^T, && (9)\\ \boldsymbol{e}(n) &= \left[\left(\boldsymbol{e}_1(n)\right)^T,\ldots,\left(\boldsymbol{e}_{M_{\mathrm{R}}}(n)\right)^T\right]^T, && (10)\\ \boldsymbol{\Psi}^l(n) &= \mathrm{diag}\left\{\left(\boldsymbol{\Psi}_1^l(n)\right)^T,\ldots,\left(\boldsymbol{\Psi}_{M_{\mathrm{R}}}^l(n)\right)^T\right\}. && (11) \end{aligned}$$

Finally, stacking $\left\{\boldsymbol{s}^l(n)\right\}_{l=1}^{L}$ and $\left\{\boldsymbol{\Psi}^l(n)\right\}_{l=1}^{L}$ into $LM_{\mathrm{T}}M_{\mathrm{R}}$ dimensional column vector and $(M_{\mathrm{T}}M_{\mathrm{R}})\times(LM_{\mathrm{T}}M_{\mathrm{R}})$ dimensional matrix respectively, we obtain

$$\begin{aligned} \boldsymbol{s}(n) &= \left[\boldsymbol{s}^1(n),\ldots,\boldsymbol{s}^L(n)\right]^T, && (12)\\ \boldsymbol{\Psi}(n) &= \left[\boldsymbol{\Psi}^1(n),\ldots,\boldsymbol{\Psi}^L(n)\right]. && (13) \end{aligned}$$

Therefore, we can express the received vector at the n^{th} time snapshot as

$$\boldsymbol{y}(n) = \boldsymbol{\Psi}(n)\boldsymbol{s}(n) + \boldsymbol{e}(n), \tag{14}$$

where $\boldsymbol{s}(n)$ is a sparse vector with $KM_{\mathrm{T}}M_{\mathrm{R}}$ non-zero entries. We have expressed our observed data using sparse representation. We assume that the target RCS values do not vary over a period of N time snapshots. Now, we stack $\{\boldsymbol{y}(n)\}_{n=1}^N$, $\{\boldsymbol{e}(n)\}_{n=1}^N$, and $\{\boldsymbol{\Psi}(n)\}_{n=1}^N$ into

$$\begin{aligned} \boldsymbol{y}_{(NM_{\mathrm{T}}M_{\mathrm{R}})\times(1)} &= \left[\left(\boldsymbol{y}(1)\right)^T,\ldots,\left(\boldsymbol{y}(N)\right)^T\right]^T,\\ \boldsymbol{e}_{(NM_{\mathrm{T}}M_{\mathrm{R}})\times(1)} &= \left[\left(\boldsymbol{e}(1)\right)^T,\ldots,\left(\boldsymbol{e}(N)\right)^T\right]^T,\\ \boldsymbol{\Psi}_{(NM_{\mathrm{T}}M_{\mathrm{R}})\times(LM_{\mathrm{T}}M_{\mathrm{R}})} &= \left[\left(\boldsymbol{\Psi}(1)\right)^T,\ldots,\left(\boldsymbol{\Psi}(N)\right)^T\right]^T, \end{aligned}$$

to obtain

$$\boldsymbol{y} = \boldsymbol{\Psi}\boldsymbol{s} + \boldsymbol{e}. \tag{15}$$

$\boldsymbol{\Psi}$ is known and only $\boldsymbol{s}$ depends on the actuals targets.

III. Sparse Support Recovery

In order to find the properties of the targets (position, velocity), we need to recover the sparse vector $\boldsymbol{s}$ from the measurements $\boldsymbol{y}$. The two most popular approaches for sparse signal recovery are Matching Pursuit [13] (MP) and Basis Pursuit [14] (BP).

A. Matching Pursuit (MP)

Matching pursuit is an iterative algorithm [13] that can be used for sparse signal recovery. Since all the columns of $\boldsymbol{\Psi}$ are not necessarily independent, there are infinitely many solutions for $\boldsymbol{s}$ even when there is no noise. In MP, we first initialize the reconstructed vector $\boldsymbol{s}^{(0)} = \mathbf{0}$ and the residual $\boldsymbol{r}^{(0)} = \boldsymbol{y}$. In each subsequent iteration k', we project the residual vector $\boldsymbol{r}^{(k'-1)}$ onto all the columns of $\boldsymbol{\Psi}$ and pick the column $\boldsymbol{\psi}^{(k')}$ that has the highest correlation with the residual. We update the estimated reconstructed vector

$$\boldsymbol{s}^{(k')} = \boldsymbol{s}^{(k'-1)} + \frac{\langle \boldsymbol{r}^{(k'-1)}, \boldsymbol{\psi}^{(k')} \rangle}{\langle \boldsymbol{\psi}^{(k')}, \boldsymbol{\psi}^{(k')} \rangle} \boldsymbol{\psi}^{(k')}.$$

We finally update the residual as

$$\boldsymbol{r}^{(k')} = \boldsymbol{r}^{(0)} - \boldsymbol{s}^{(k')}.$$

After sufficient number of iterations, the residual approaches zero.

B. Basis Pursuit (BP)

Basis pursuit is an optimization principle. In MP, we are not optimizing a specific objective function. We just proceed heuristically to obtain the solution. However, in BP, we minimize an objective function [14] and we are guaranteed to solve the optimization problem. BP is presented under two scenarios; in the absence of noise and in the presence of noise.

1) Absence of Noise: In the absence of noise, BP aims at minimizing $\|\boldsymbol{s}\|_1$ under the constraint $\boldsymbol{y} = \boldsymbol{\Psi}\boldsymbol{s}$. Since usually $N \ll L$, there are many different vectors $\boldsymbol{s}$ that satisfy the constraint. We choose the solution that has the least l_1 norm. This optimization problem can be modeled as a linear program [14]. There are many existing algorithms to solve this problem. One such algorithm is the BP-Simplex algorithm. In MP, we begin with an empty basis and we keep adding columns to the basis in each iteration by computing the column that gives the maximum correlation with the residue. However, in BP we begin with a basis set $\boldsymbol{\Psi}_{\mathrm{B}}^{(0)}$ that contains $KM_{\mathrm{T}}M_{\mathrm{R}}$ linearly independent columns of $\boldsymbol{\Psi}$ such that $\boldsymbol{y}$ lies in the column space of $\boldsymbol{\Psi}_{\mathrm{B}}^{(0)}$. The rest of the columns form another set $\overline{\boldsymbol{\Psi}_{\mathrm{B}}^{(0)}}$. We recover the initial estimate $\boldsymbol{s}^{(0)}$ by using the basis $\boldsymbol{\Psi}_{\mathrm{B}}^{(0)}$. In every subsequent iteration $k' \geq 1$, we choose the swap of a column between $\boldsymbol{\Psi}_{\mathrm{B}}^{(k'-1)}$ and $\overline{\boldsymbol{\Psi}_{\mathrm{B}}^{(k'-1)}}$ such that the l_1 norm of the reconstructed vector $\boldsymbol{s}^{(k')}$ corresponding to this swap is lower than that corresponding to any other swap. We update the basis sets according to this swap. Once we reach the optimal solution, there will be no swap that will further reduce the l_1 norm. This algorithm always guarantees convergence to the optimal solution in the absence of noise.

2) Presence of Noise: Clearly the above approach of basis pursuit will fail in the presence of noise. Hence, in [14], the authors propose Basis Pursuit De-Noising (BPDN). This is an unconstrained minimization problem

$$\min \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{\Psi}\boldsymbol{s}\|_2 + \lambda\|\boldsymbol{s}\|_1.$$

Typically $\lambda = \sigma\sqrt{2\log\left(LM_{\mathrm{T}}M_{\mathrm{R}}\right)}$ where σ represents the noise level. To solve this problem we used $\mathtt{CVX}$, a package for specifying and solving convex programs [15], [16]. We present the results in section V.

IV. Optimal Adaptive Energy Allocation

Before we propose the energy allocation mechanism, we shall first define an important performance metric. As mentioned earlier, the $LM_{\mathrm{T}}M_{\mathrm{R}}$ length vector $\boldsymbol{s}$ has only $KM_{\mathrm{T}}M_{\mathrm{R}}$ non-zero entries. Let the reconstructed vector be denoted by $\hat{\boldsymbol{s}}$. We would like to have the most significant $KM_{\mathrm{T}}M_{\mathrm{R}}$ entries of $\hat{\boldsymbol{s}}$ correspond to the same indices as the non-zero entries of $\boldsymbol{s}$. If this is not the case, then we will wrongly map the target states for one or more targets. We define a L length vector $\tilde{\boldsymbol{s}}$

$$\tilde{\boldsymbol{s}}(l) = \sum_{i=1}^{M_{\mathrm{T}}}\sum_{j=1}^{M_{\mathrm{R}}} \|\hat{\boldsymbol{s}}(M_{\mathrm{T}}M_{\mathrm{R}}(l-1) + M_{\mathrm{R}}(j-1) + i)\|^2.$$

Further, define $\widetilde{\boldsymbol{s}^*}$ as a K length vector which contains the values that $\tilde{\boldsymbol{s}}$ carries at the correct K indices. Similarly we define $\overline{\widetilde{\boldsymbol{s}^*}}$ as a L length vector that takes a value of 0 at the correct K indices and takes the same values as $\tilde{\boldsymbol{s}}$ at every other index. It is clear that the non-zero entries of $\overline{\widetilde{\boldsymbol{s}^*}}$ correspond to the non-target states.

We define the metric

$$\triangle = \frac{\min \widetilde{\boldsymbol{s}^*}}{\max \overline{\widetilde{\boldsymbol{s}^*}}}.$$

If this metric has a value greater than 1, then all the actual correct target indices dominate the other indices in $\tilde{\boldsymbol{s}}$. This is desirable. Otherwise, the estimates are not good enough. Hence, the higher the value of $\triangle$, the better the performance of the system. In section V, we use this metric to analyze the results.

Let E_i be the energy of the waveform transmitted from the i^{th} transmitter. So far, we have assumed that all the transmitters send out orthonormal waveforms. Hence, the transmitting energies for all the transmitters $E_i = 1$ and the total energy transmitted $\sum_{i=1}^{M_{\mathrm{T}}} E_i = M_{\mathrm{T}}$. At the receiver side, we use the algorithms mentioned in the previous section to obtain the reconstructed sparse vector $\hat{\boldsymbol{s}}$. This vector contains estimates of the attenuations a_{ij}^k. Since the different antenna pairs view the targets from different angles, these attenuations will be different from each other. Hence, equal energy allocation to all the transmitters does not necessarily give the best performance. We have the over all energy constraint $\sum_{i=1}^{M_{\mathrm{T}}} E_i = M_{\mathrm{T}}$. We initialize the system by transmitting equal energies $E_i = 1$ from all the transmitters and estimate the RCS values $\widehat{a_{ij}^k}$ from the reconstructed vector $\hat{\boldsymbol{s}}$. We aim to optimize the performance of the system for the next processing interval by picking the transmit energy allocation that maximizes the minimum target returns. Hence, the goal is to solve the following optimization problem and find the optimal E_i such that $\sum_{i=1}^{M_{\mathrm{T}}} E_i = M_{\mathrm{T}}$

$$\max_{E_i} \min_{k} \sum_{i=1}^{M_{\mathrm{T}}}\sum_{j=1}^{M_{\mathrm{R}}} E_i \|\widehat{a_{ij}^k}\|^2.$$

Since the operators $\max\{\cdot\}$ and $\min\{\cdot\}$ can be represented as the l_∞ and $l_{-\infty}$ norms respectively, we can solve the above optimization problem using CVX [15], [16]. By this optimization criterion, we are maximizing the numerator in the expression for $\triangle$ for the next processing interval. We shall show in section VI that this optimal choice of waveform energies gives significant improvement in performance.

V. Numerical Results

We simulated a 2×2 MIMO radar system. We denote the positions of all the transmitters, targets and receivers on a common Cartesian coordinate system. The transmitters are located at $[100, 0]\,\mathrm{m}$ and $[200, 0]\,\mathrm{m}$. The receiver locations are $[0, 100]\,\mathrm{m}$ and $[0, 200]\,\mathrm{m}$. The carrier frequency of the transmitted waveforms is $f_{\mathrm{c}} = 1\mathrm{GHz}$. Within each processing interval, we consider three pulses that are transmitted $33.3\mathrm{ms}$ apart. We choose $N = 243$ for the simulation results. We choose $N = 243$. Therefore, $\boldsymbol{y}$ has 972 entries. We divide the target position space into 9×9 grid points and the target velocity space into 5×5 grid points. Therefore, the total number of possible target states $L = 2025$. We considered the presence of 3 targets. The positions and the velocities of the targets are given as

$$\begin{aligned}\overrightarrow{\boldsymbol{p}^1} &= [110, 280]\,\mathrm{m},\\ \overrightarrow{\boldsymbol{v}^1} &= [120, 100]\,\mathrm{m/s},\\ \overrightarrow{\boldsymbol{p}^2} &= [80, 280]\,\mathrm{m},\\ \overrightarrow{\boldsymbol{v}^2} &= [110, 110]\,\mathrm{m/s},\\ \overrightarrow{\boldsymbol{p}^3} &= [100, 260]\,\mathrm{m},\\ \overrightarrow{\boldsymbol{v}^3} &= [130, 130]\,\mathrm{m/s}.\end{aligned}$$

The attenuations for the 3 targets are

$$\begin{aligned}\left[a_{11}^1, a_{12}^1, a_{21}^1, a_{22}^1\right] &= \left[0.3\epsilon, 0.3\epsilon, 0.7\epsilon, 0.8\epsilon\right],\\ \left[a_{11}^2, a_{12}^2, a_{21}^2, a_{22}^2\right] &= \left[0.4\epsilon, 0.5\epsilon, 0.3\epsilon, 0.2\epsilon\right],\\ \left[a_{11}^3, a_{12}^3, a_{21}^3, a_{22}^3\right] &= \left[0.4\epsilon, 0.5\epsilon, 0.8\epsilon, 0.7\epsilon\right],\end{aligned}$$

where $\epsilon = 1 + \sqrt{-1}$. The entries of $\boldsymbol{e}$ are generated independently from Gaussian distribution. We assume each of these samples has the same variance σ^2. We define the signal to noise ratio (SNR) for the MIMO radar system as $10 \log\left(\frac{\|\boldsymbol{\Psi s}\|^2}{\mathrm{E}\left(\|\boldsymbol{e}\|^2\right)}\right)$ dB. First we compare the performances of the two algorithms matching pursuit and basis pursuit de-noising. We perform these simulations at an SNR of $13.2\mathrm{dB}$ and for BPDN we chose $\lambda = \sigma\sqrt{2\log\left(LM_{\mathrm{T}}M_{\mathrm{R}}\right)}$. For MP, we used 20 iterations. Since it is not possible to plot the position and velocity on the same plot, we plotted the estimates of position and velocity separately. For computing the estimate at a particular grid point on the position plot, we average over all 5×5 velocity grid points corresponding to that position grid point. Similarly, we average over all the 9×9 position grid points in order to obtain the velocity plot. We do this only to be able to plot position and velocity estimates separately.

From Fig. 1 and Fig. 2, we can see that both the algorithms are able to estimate the positions and velocities of the 3 targets. However, it is important for us to analyze the performances of the two algorithms by evaluating the performance metric $\triangle$.

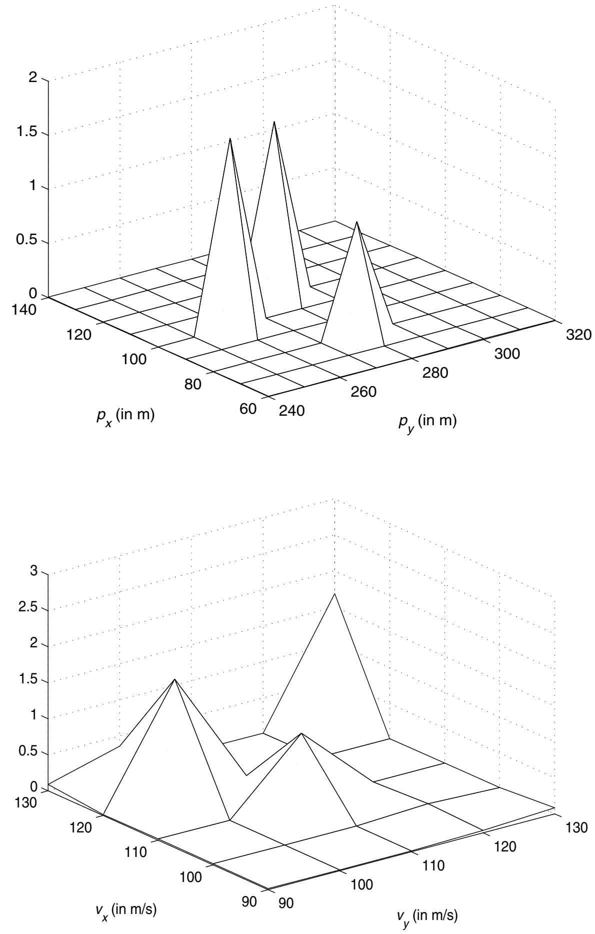

Fig. 1. Reconstructed vectors using matching pursuit at SNR=$13.2\mathrm{dB}$, (a) position estimates, (b) velocity estimates.

Fig. 3 plots $\triangle$ as a function of the SNR and we can clearly see that BPDN outperforms MP. $\triangle$ remains above 1 for much lower SNR for BPDN when compared with MP. When $\triangle <$ 1, some of the non-target states dominate the reconstructed vector. Since BPDN outperforms MP, for all further simulation results, we shall use only BPDN. We used 100 independent Monte Carlo runs to generate these results.

Now we demonstrate the advantages of having adaptive energy allocation. We assume we have estimates of the target RCS values from the estimation of the previous processing interval. We apply the optimization principle described in section IV. As we see from Fig. 4, the adaptive energy allocation gives significant improvement in performance. The value of $\triangle$ is higher when compared with the equal energy transmission. Even at an extremely low SNR of $-1.1\mathrm{dB}$, the value of $\triangle$ remains greater than 1 for the proposed energy allocation scheme. Also, at an SNR of $13.2\mathrm{dB}$, $\triangle$ nearly doubles while using the energy allocation scheme.

VI. Conclusion

We proposed a sparse modeling approach to estimate the positions and velocities of multiple targets using distributed MIMO radar. We demonstrated the accurate reconstruction of

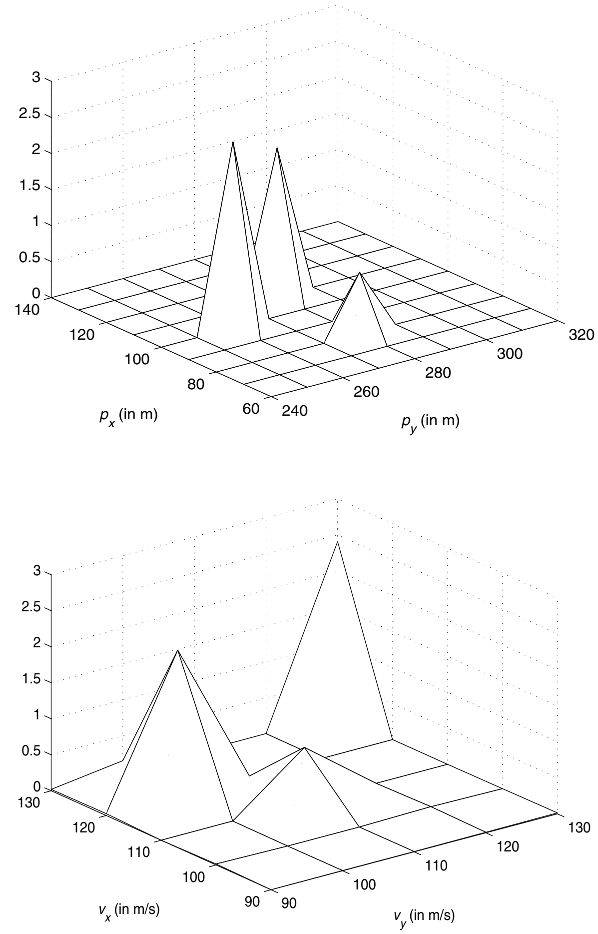

Fig. 2. Reconstructed vectors using basis pursuit de-noising at SNR=13.2dB, (a) position estimates, (b) velocity estimates.

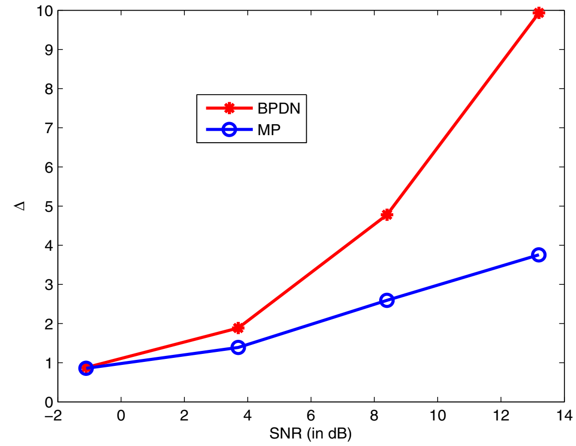

Fig. 3. Performance metric $\triangle$ for matching pursuit and basis pursuit de-noising as a function of SNR.

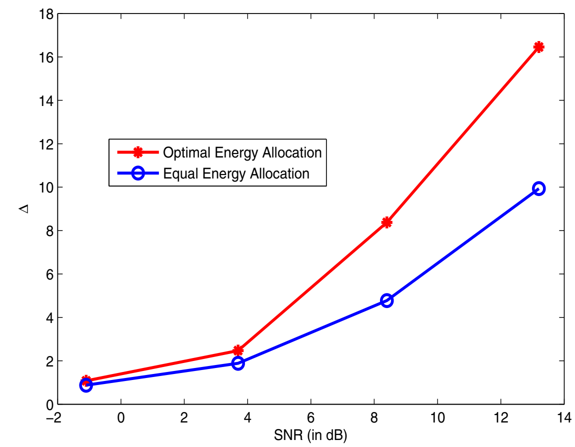

Fig. 4. Performance metric $\triangle$ with and without adaptive energy allocation.

the target state vectors. We introduced a new metric to analyze the performance of the system. Also, we proposed an optimal adaptive energy allocation mechanism to further enhance the performance. We show this improvement using simulations.

REFERENCES

[1] A. M. Haimovich, R. S. Blum, and L. J. Cimini, "MIMO radar with widely separated antennas," *IEEE Signal Process. Mag.*, vol. 25, pp. 116–129, Jan. 2008.

[2] J. Li and P. Stoica, *MIMO radar signal processing*. Hoboken, NJ: John Wiley & Sons, Inc., 2009.

[3] S. Gogineni and A. Nehorai, "Polarimetric MIMO radar with distributed antennas for target detection," *IEEE Trans. Signal Process.*, vol. 58, pp. 1689–1697, Mar. 2010.

[4] J. Li and P. Stoica, "MIMO radar with colocated antennas," *IEEE Signal Process. Mag.*, vol. 24, pp. 106–114, Sep. 2007.

[5] S. Gogineni and A. Nehorai, "Polarimetric MIMO radar with distributed antennas for target detection," in *Proc. 43rd Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, Nov. 2009.

[6] R. Baraniuk and P. Steeghs, "Compressive radar imaging," in *IEEE Radar Conference*, Boston, MA, Apr. 2007, pp. 128–133.

[7] C.-Y. Chen and P. P. Vaidyanathan, "Compressed sensing in MIMO radar," in *42nd Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, Oct. 2008, pp. 41–44.

[8] Y. Yao, A. P. Petropulu, and H. V. Poor, "Compressive sensing for MIMO radar," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Taipei, Apr. 2009, pp. 3017–3020.

[9] Y. Yu, A. P. Petropulu, and H. V. Poor, "MIMO radar using compressive sampling," *IEEE Jour. of Selected Topics in Signal Proc.*, vol. 4, pp. 146–163, Feb. 2010.

[10] M. Hurtado, J. J. Xiao, and A. Nehorai, "Target estimation, detection, and tracking: A look at adaptive polarimetric design," *IEEE Signal Process. Mag.*, vol. 26, pp. 42–52, Jan. 2009.

[11] B. Friedlander, "Waveform design for MIMO radars," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, pp. 1227–1238, Jul. 2007.

[12] S. Gogineni and A. Nehorai, "Compressive sensing for MIMO radar with widely separated antennas," *IEEE Trans. Signal Process.*, submitted.

[13] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, pp. 3397–3415, Dec. 1993.

[14] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, pp. 129–159, Mar. 2001.

[15] M. Grant and S. Boyd. (2009, Jun.) CVX: Matlab software for disciplined convex programming (web page and software). [Online]. Available: http://stanford.edu/∼boyd/cvx

[16] ——. (2008) Graph implementations for nonsmooth convex programs, recent advances in learning and control (a tribute to M. Vidyasagar), V. Blondel, S. Boyd, and H. Kimura, editors, pages 95-110, lecture notes in control and information sciences, Springer. [Online]. Available: http://stanford.edu/∼boyd/graph_dcp.html

978-1-4244-8200-9/10/$26.00 ©2010 IEEE