COOPERATIVE SPARSITY PATTERN RECOVERY IN DISTRIBUTED NETWORKS VIA DISTRIBUTED-OMP

Thakshila Wimalajeewa and Pramod K. Varshney

Department of Electrical Engineering and Computer Science
Syracuse University, Syracuse, NY 13244
Email: twwewelw@syr.edu, varshney@syr.edu

ABSTRACT

In this paper, we address the problem of sparsity pattern recovery of a sparse signal with multiple measurement data in a distributed network. We consider that each node in the network makes measurements via random projections regarding the same sparse signal. We propose a distributed greedy algorithm based on Orthogonal Matching Pursuit (OMP) in which the locations of non zero coefficients of the sparse signal are estimated iteratively while performing fusion of estimates at distributed nodes. In the proposed distributed framework, each node has to perform less number of iterations of OMP compared to the sparsity index of the sparse signal. With each node having a very small number of compressive measurements, a significant performance gain in sparsity pattern detection is achieved via the proposed collaborative scheme compared to the case where each node estimates the sparsity pattern independently and then fusion is performed to get a global estimate. We further extend the algorithm to a binary hypothesis testing framework, where the algorithm first detects the presence of a sparse signal collaborating among nodes with a fewer number of iterations of OMP and then increases the number of iterations to estimate the sparsity pattern only if the signal is detected.

Index Terms— Compressive sensing, Sparsity pattern detection, multiple measurement vectors, distributed networks

1. INTRODUCTION

In the *Compressive Sensing* (CS) framework, a small collection of linear random projections of a sparse signal contains sufficient information for complete signal recovery [1–3]. There is a considerable amount of work on the development of computationally efficient and tractable algorithms to recover sparse signals from CS based measurements obtained via random projections, for example in [4–12]. There are certain signal processing applications where complete signal recovery is not necessary. For example, in applications such as source localization in sensor networks [13, 14], estimation of frequency band locations in cognitive radio networks [15], localization of neural electrical activities from a huge number of potential locations in magnetoencephalography (MEG) and electroencephalography (EEG) for medical imaging applications [16–18], sparse approximation [19], subset selection in linear regression [20, 21], and signal denoising [22], it is only necessary to estimate the locations of the significant coefficients of a sparse signal. Further, in the CS framework, once the the locations of non zero coefficients are identified, the signal can be estimated using standard techniques. Thus, the problem of support identification of sparse signals is important with many applications.

The problem of sparsity pattern detection has been addressed by several authors in the CS literature [23–31]. These studies mostly focus on single measurement vectors. In distributed networks including sensor and cooperative cognitive radio networks, multiple measurements appear quite naturally since multiple nodes make observations regarding the same phenomenon of interest (PoI). Extensions of standard sparse signal recovery techniques for multiple measurement vectors are considered in [32, 33] assuming that the multiple measurements are available at a single location to perform the desired task. In such centralized settings, each node in the network has to transmit its observations along with the elements of the measurement matrix to solve the sparse signal recovery problem. However, the trade-off between the resource constraints (e.g. node power and communication bandwidth) and the achievable performance gain is a core issue to be addressed in many practical distributed networks. The problem of distributed sparsity pattern recovery is considered recently in the context of cognitive radio networks in [34, 35]. There, decentralized consensus based algorithms for support recovery of sparse signals were proposed when each cognitive radio makes CS based measurements.

This paper focuses on further reducing the computational and communication burden at individual nodes while performing sparsity pattern recovery distributively. In the proposed distributed scheme, each node finds an estimate of the sparsity pattern by proper collaboration and fusion. More specifically, we develop a greedy algorithm based on Orthogonal Matching Pursuit (OMP) where the indices of the sparse support are iteratively identified while fusing the estimated indices at distributed nodes at each iteration. We show that, in the proposed distributed algorithm, each node has to perform less number of iterations of the greedy algorithm compared to the sparsity index of the sparse signal to reliably estimate the sparsity pattern (in the standard OMP algorithm, at least K iterations are required for sparsity pattern recovery where K is the sparsity index of the sparse signal). Moreover, in the proposed algorithm, each node transmits only one index at each iteration to perform fusion. Thus, each node has to communicate less information while estimating the sparsity pattern distributively compared to the schemes proposed in [34, 35]. Further, we extend the results and develop a joint algorithm to both detect the sparse signal and to perform the sparsity pattern recovery only if the sparse signal is detected.

2. PROBLEM FORMULATION AND MOTIVATION

Consider a distributed network with L distributed nodes. We assume that each node obtains a $M(< N)$-length measurement vector $\mathbf{y}_l$ via linear random projections. The measurement vector at the l-th node is given by,

$$\mathbf{y}_l \quad = \quad \mathbf{A}_l \mathbf{s} + \mathbf{v}_l; \tag{1}$$

for $l = 0, 1, \cdots, L-1$ where $\mathbf{s}$ is the $N \times 1$ sparse signal vector and $\mathbf{A}_l$ is the $M \times N$ random projection matrix at the l-th node. The noise vector $\mathbf{v}_l$ at the l-th node is assumed to be iid Gaussian with zero mean vector and the covariance matrix $\sigma_v^2 \mathbf{I}_M$ where $\mathbf{I}_M$ is the $M \times M$ identity matrix. When the signal $\mathbf{s}$ is sparse in an orthonormal basis Ψ such that $\mathbf{s} = \Psi\boldsymbol{\beta}$ where $\boldsymbol{\beta}$ contains only $K << N$ number of significant coefficients, it has been shown that the randomized lower dimensional projections of the form (1) can capture significant information of the sparse signal $\mathbf{s}$ [1–3].

Let $\mathcal{U}$ be the set which contains the indices of the support of the sparse coefficient vector $\boldsymbol{\beta}$ which is defined as, $\mathcal{U} := \{i \in \{1, 2, \cdots, N\} \mid \boldsymbol{\beta}(i) \neq 0\}$ where $\boldsymbol{\beta}(i)$ is the i-th element of $\boldsymbol{\beta}$ for $i = 0, 1, \cdots, N-1$. Then we have $K = n(\mathcal{U})$ where $n(\mathcal{S})$

denotes the cardinality of the set $\mathcal{S}$. Further, let $\mathbf{b}$ be a N-length vector which contains binary elements: i.e.

$$\mathbf{b}(i) = \begin{cases} 1 & if \boldsymbol{\beta}(i) \neq 0 \\ 0 & otherwise \end{cases}$$

for $i = 0, 1, \cdots, N-1$ and $\hat{\mathbf{b}}$ be the estimated binary support vector.

Sparsity pattern recovery is to estimate the set $\mathcal{U}$ (or the vector $\mathbf{b}$) based on the observation model (1). With a single measurement vector, sparsity pattern recovery can be performed via a CS reconstruction algorithm such as regularized least square algorithm (Lasso) [1, 36] or OMP [8]. Although Lasso provides with more promising results in the context of accuracy, its computational complexity is higher than that with greedy techniques such as OMP. It has been shown in [8, 37] that OMP requires approximately $M \geq (1+\epsilon)4K \log N$, $\epsilon > 0$ measurements for reliable sparsity pattern recovery while Lasso scales as $2K \log(N-K) + K + 1$ [38] when the measurement SNR is large. Further, it was shown in [37] that for sparsity pattern recovery with large Gaussian measurement matrices in high SNR, lasso and OMP have almost identical performance.

When multiple measurement vectors as in (1) are collected at a centralized location, the support of the sparse signal can be estimated, for example, by solving the simultaneous OMP algorithm as presented in [32] [33]. To implement such centralized sparsity pattern recovery algorithm with multiple measurement data, it is required that each node transmits its M-length observation vector along with the elements of the measurement matrix to a central fusion center. Since transmitting all the information to a fusion center is not desirable in power and bandwidth constrained communication networks, we consider a distributed scheme in which, sparsity pattern estimation is performed via collaboration among nodes with less communication and with each node estimating the sparsity pattern. In [34, 35], several consensus based distributed schemes are proposed to estimate the support of sparse signals based on Lasso. These schemes estimate the full support set at each node and then perform fusion via several collaboration schemes. However, due to computational complexity, performing Lasso at power constrained sensor nodes may not be desirable. Our focus in this paper is on proposing a distributed greedy algorithm for sparsity pattern recovery with less communication burden compared to consensus based approaches as proposed in [34, 35].

3. DISTRIBUTED SPARSITY PATTERN RECOVERY WITH MULTIPLE MEASUREMENT VECTORS

OMP is a greedy technique in which at each iteration, the location of one non zero coefficient of the signal (or the index of a column of $\Theta = \mathbf{A}\Psi$ that participates in the measurement vector $\mathbf{y}$) is identified. More specifically, at each iteration, the algorithm picks the column of Θ which is most correlated with the remaining part of $\mathbf{y}$. Then the selected column's contribution is subtracted from $\mathbf{y}$ and iterations on the residual are carried out. The standard OMP algorithm for sparsity pattern detection with single measurement vector is presented in Algorithm 1. If we consider that the standard OMP is performed at one sensor node as stated in Algorithm 1, at each iteration, there are $N-K$ possible incorrect indices that can be selected. The probability of selecting an incorrect index at each iteration increases as the number of CS measurements at each node (M) decreases. The goal of the proposed algorithm is to reduce the probability that OMP selects an incorrect index as the true location of non zero entries via collaboration and fusion.

3.1. Distributed and collaborative OMP for sparse support estimation

Define $\mathcal{M}_l$ to be the set containing the neighboring nodes of the l-th node (including itself). As defined in Section 2, let $\mathcal{U}$ be the support set which contains the indices of non zero coefficients of the sparse signal and $\hat{\mathcal{U}}_l$ be the estimated support set at the l-th node. Further, let $\Theta_l = \mathbf{A}_l\Psi$ and $\theta_{l,i}$ be the i-th column of the matrix Θ_l. $\Theta_l(\mathcal{A})$ denotes the submatrix of Θ_l which has columns of Θ_l corresponding to the elements in the set $\mathcal{A}$ for $\mathcal{A} \subset \{0, 1, \cdots, N-1\}$. $|.|$ denotes the absolute value while $||.||_p$ denotes the l_p norm. Further, $n(\mathcal{S})$ denotes the cardinality of the set $\mathcal{S}$ as defined before.

Algorithm 1 Standard OMP for sparsity pattern estimation at the l-th node

1. Initialize $t = 1, \hat{\mathcal{U}}_l(0) = \emptyset$, residual vector $\mathbf{r}_{l,0} = \mathbf{y}_l$
2. Find the index $\lambda_l(t)$ such that $\lambda_l(t) = \underset{\omega=1,\cdots,N}{arg\ max}\ |\langle \mathbf{r}_{l,t-1}, \theta_{l,\omega}\rangle|$
3. Set $\hat{\mathcal{U}}_l(t) = \hat{\mathcal{U}}_l(t-1) \cup \{\lambda_l(t)\}$
4. Compute the projection operator $\mathbf{P}_l(t) = \Theta_l(\hat{\mathcal{U}}_l(t)) \left(\Theta_l(\hat{\mathcal{U}}_l(t))^T \Theta_l(\hat{\mathcal{U}}_l(t))\right)^{-1} \Theta_l(\hat{\mathcal{U}}_l(t))^T$. Update the residual vector: $\mathbf{r}_{l,t} = (\mathbf{I} - \mathbf{P}_l(t))\mathbf{y}_l$
5. Increment $t = t + 1$ and go to step 2 if $t \leq K$, otherwise, stop and set $\hat{\mathcal{U}}_l = \hat{\mathcal{U}}_l(t)$

The proposed distributed and collaborative OMP (DC-OMP) algorithm is stated in Algorithm 2. Once the l-th node finds an index $\lambda_l(t)$ (which corresponds to the column that is most correlated with the remaining part of $\mathbf{y}_l$) at t-th iteration by performing step 2 in Algorithm 2, it is exchanged among the neighborhood $\mathcal{M}_l$. Subsequently, each node will have the estimated index at every node in its neighborhood at t-th iteration. By fusion, each node selects a set of indices (from $n(\mathcal{M}_l)$ number of indices) such that most of the nodes agree upon (more details of this fusion are provided in Subsection 3.1.1). Note that, in this step several indices can be selected and thus, more than one index of the true support can be estimated at a given iteration. Thus, the number of iterations required to estimate the support at each node can be less than the sparsity index K.

Algorithm 2 Distributed OMP for sparsity pattern estimation

At l-th node:

1. Initialize $t = 1, \hat{\mathcal{U}}_l(0) = \emptyset$, residual vector $\mathbf{r}_{l,0} = \mathbf{y}_l$
2. Find the index $\lambda_l(t)$ such that $\lambda_l(t) = \underset{\omega=1,\cdots,N}{arg\ max}\ |\langle \mathbf{r}_{l,t-1}, \theta_{l,\omega}\rangle|$
3. Update the index set $\lambda_l^*(t)$ via local communication: $\lambda_l^*(t) = f_l(\lambda_l(t), \{\lambda_i(t)\}, i \in \mathcal{M}_l)$, as discussed in subsection 3.1.1
4. Set $\hat{\mathcal{U}}_l(t) = \hat{\mathcal{U}}_l(t-1) \cup \{\lambda_l^*(t)\}$ and $l_t = n(\hat{\mathcal{U}}_l(t))$
5. Compute the projection operator $\mathbf{P}_l(t) = \Theta_l(\hat{\mathcal{U}}_l(t)) \left(\Theta_l(\hat{\mathcal{U}}_l(t))^T \Theta_l(\hat{\mathcal{U}}_l(t))\right)^{-1} \Theta_l(\hat{\mathcal{U}}_l(t))^T$. Update the residual vector: $\mathbf{r}_{l,t} = (\mathbf{I} - \mathbf{P}_l(t))\mathbf{y}_l$
6. Increment $t = t + 1$ and go to step 2 if $l_t \leq K$, otherwise, stop and set $\hat{\mathcal{U}}_l = \hat{\mathcal{U}}_l(t)$

3.1.1. Performing step 3 in Algorithm 2

To perform step 3 in the Algorithm 2 we propose the following procedure.

Case I: Consider the case where the l-th node can broadcast its estimated index of the sparse support at each iteration to the rest of the nodes in the network. i.e. $\mathcal{M}_l = \bar{\mathcal{M}}$ where $\bar{\mathcal{M}}$ contains the indices of all the nodes in the network. This is a reasonable assumption when there are only a small number of nodes in the network (e.g. cognitive radio networks with $5-10$ cognitive radios). During t-th iteration, the l-th node broadcasts $\lambda_l(t)$. Consequently, the l-th node receives the estimates $\lambda_i(t)$'s for $i \in \bar{\mathcal{M}}$ from the whole network. Further, let $c(t)$ be a L-length vector that contains all the indices estimated at L nodes from step 2 in Algorithm 2 during the t-th iteration (i.e. values of $\lambda_i(t)$'s for $i = 0, 1, \cdots, L-1$). At t-th iteration, $\lambda_l^*(t)$ is computed as follows: Check whether there are indices in $c(t)$ with more than one occurrences.

- If yes, such indices are put in the set $\lambda_l^*(t)$ (such that $\lambda_l^*(t) = \{set\ of\ indices\ in\ c(t)\ which\ occur\ more\ than\ once\}$.
- If no, (i.e. there is no index obtained from step 2 such that any two or more nodes agree with, so that all L indices in $c(t)$ are distinct), then select one index from $c(t)$ randomly and put in $\lambda_l^*(t)$. In this case, to avoid the same index being selected at subsequent iterations, we force all nodes to use the same index.

It is noted that when $L \geq K$,it is more likely that the vector $c(t)$ has at least one set of two indices with the same value, thus, $\lambda_l^*(t)$ is updated appropriately most of the time. Further, since each node has the indices received from all the other nodes in the network, every node has the same estimate for $\mathcal{U}$ when the algorithm terminates.

Case II: Next, we consider the case where $\mathcal{M}_l \subset \bar{\mathcal{M}}$; i.e. each node communicates its estimated index in its neighborhood which has fewer number of nodes compared to all the nodes in the network. Then, similar to the case I, $\lambda_l^*(t)$ is found based on $c_l(t)$ as the indices which have more than one occurrences from $c_l(t)$ where $c_l(t)$ contains the indices received by the l-th node from its neighborhood at t-th iteration. However, in this case, since l-th node does not receive the estimated indices of the sparse support from the whole network at a given iteration, different nodes may agree upon different indices. When two neighboring nodes agree upon two different indices at t-th iterations, there is a possibility that one node selects the same index at a later iteration beyond t. To avoid the l-th node selecting the same index twice, we perform an additional step in updating $\lambda_l^*(t)$ compared to case I; i.e., to check whether $\lambda_l^*(t)$ determined as in case I is in $\hat{\mathcal{U}}_l(t-1)$. If $\lambda_l^*(t) \in \hat{\mathcal{U}}_l(t-1)$, set $\lambda_l^*(t) = \lambda_l(t)$, otherwise update $\lambda_l^*(t)$ similar to the procedure described in Case I.

4. SPARSE SIGNAL DETECTION AND SPARSITY PATTERN ESTIMATION

Next, we consider the case where it is required to detect whether the sparse signal is present and estimate the sparsity pattern only if the signal is present. We consider the following binary hypothesis testing problem,

$$\mathcal{H}_1 : \mathbf{y}_l = \mathbf{A}_l \mathbf{s} + \mathbf{v}_l \ , \ \ \mathcal{H}_0 : \mathbf{y}_l = \mathbf{v}_l. \tag{2}$$

for $l = 0, 1, \cdots, L-1$ and the sparse signal is present under hypothesis $\mathcal{H}_1$. In the following, we extend the collaborative algorithm discussed in Section 3.1 to first detect the sparse signal and then to estimate the sparsity pattern without ever reconstructing the signal. Further, we assume that $L \geq K$.

The proposed algorithm is presented in Algorithm 3. If the signal is not present in the model (2), it is very unlikely that two nodes in the network select the same index of the support set at any given iteration based on the step 2 in the OMP algorithm presented in Algorithm 2. When the signal is present (i.e. hypothesis $\mathcal{H}_1$ is true), the probability that two nodes select the same index at each iteration is higher especially when $L \geq K$. The main difference between Algorithm 3 and Algorithm 2 is the steps 4 and 5 in Algorithm 3 which will be discussed next.

4.1. Steps 4 and 5 in Algorithm 3

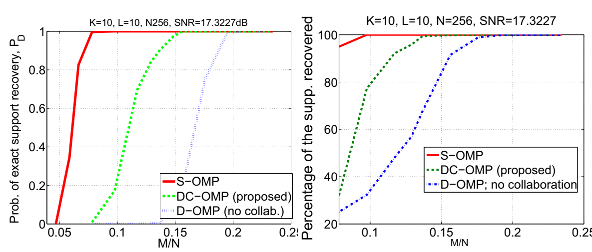
Steps 4 and 5 in Algorithm 3 are performed as follows. At t-th iteration, $c_l(t)$ contains all the indices received by the l-th node from its neighborhood. The function $unique(c_l(t))$ gives the number of distinct indices in $c(t)$. If all the indices in $c_l(t)$ are different from each other, $unique(c_l(t))$ equals to the number of nodes in the neighborhood of the l-th node including itself. If there are any two indices in $c_l(t)$ with the same value, we set the value of $\rho(c_l(t))$ as the number of such indices. After performing K_0 (which is less than K) number of iterations, if i_{index} in step 4 in Algorithm 3 is very small (less than I_0 where $I_0 \ll K$), the algorithm decides that no sparse signal is present and terminate the process resulting in the null set as the estimated support set. If $i_{index} \geq I_0$, it decides that the sparse signal is present and continues estimating the support set similar to Algorithm 2.

Algorithm 3 Distributed OMP for sparse signal detection and sparsity pattern estimation

At l-th node:

1. Initialize $t = 1$, $\hat{\mathcal{U}}_l(0) = \emptyset$, residual vector $\mathbf{r}_{l,0} = \mathbf{y}_l$, $i_{index} = 0$
2. Find the index $\lambda_l(t)$ such that $\lambda_l(t) = \underset{\omega=1,\cdots,N}{arg\ max} |\langle \mathbf{r}_{l,t-1}, \theta_{l,\omega} \rangle|$
3. Update the estimated index set $\lambda_l(t)$ via local communication: $\lambda_l^*(t) = f_l(\lambda_l(t), \{\lambda_i(t)\}, i \in \mathcal{M}_l)$, as discussed in subsection 3.1.1
4. Update i_{index} (as discussed in Subsection 4.1):
   - if $unique(c_l(t)) = n(\mathcal{M}_l)$, $i_{index} = i_{index} + 0$
   - else $(unique(c_l(t)) < n(\mathcal{M}_l))$, $i_{index} = i_{index} + \rho(c_l(t))$
5. Perform signal detection decision when $t = K_0$
   -If $t = K_0$ and $i_{index}(t) \geq I_0$, decide $\mathcal{H}_1$ and go to step 6. Avoid steps 4 and 5 in subsequent iterations
   -If $t = K_0$ and $i_{index}(t) < I_0$ decide H_0, set $\hat{\mathcal{U}}_l(t) = \emptyset$ and go to step 9
6. Set $\hat{\mathcal{U}}_l(t) = \hat{\mathcal{U}}_l(t-1) \cup \{\lambda_l^*(t)\}$, and $l_t = n(\hat{\mathcal{U}}_l(t))$
7. Compute the projection operator $\mathbf{P}_l(t) = \Theta_l(\hat{\mathcal{U}}_l(t)) \left(\Theta_l(\hat{\mathcal{U}}_l(t))^T \Theta_l(\hat{\mathcal{U}}_l(t))\right)^{-1} \Theta_l(\hat{\mathcal{U}}_l(t))^T$. Update the residual vector: $\mathbf{r}_{l,t} = (\mathbf{I} - \mathbf{P}_l(t))\mathbf{y}_l$
8. Increment $t = t + 1$ and go to step 2 if $l_t < K$,
9. set $\hat{\mathcal{U}}_l = \hat{\mathcal{U}}_l(t)$

5. SIMULATION RESULTS



(a) (b)

Fig. 1. Performance of the sparsity pattern recovery with distributed OMP algorithm 2: (a) the probability of correctly recovering the sparse support ($P_D = Pr(\hat{\mathbf{b}} = \mathbf{b})$) vs M/N (b) the percentage of the support correctly recovered vs M/N; $N = 256$, $K = 10$, $L = 10$, $\bar{\gamma} = \frac{||\mathbf{s}||_2^2}{N\sigma_v^2} = 17.3227dB$

To compare the performance of the proposed Algorithm 2 with

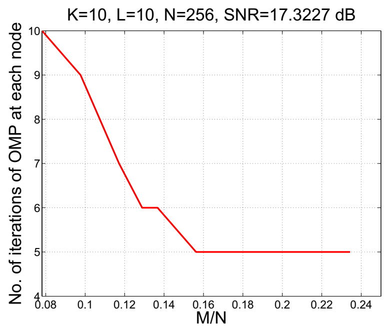
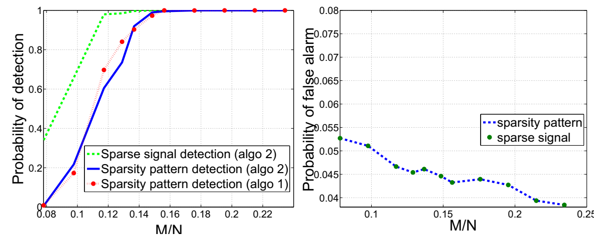


Fig. 2. Number of iterations of OMP at each node with the proposed Algorithm 2 vs M/N; $N = 256$, $K = 10$, $L = 10$, $\bar{\gamma} = \frac{||\mathbf{s}||_2^2}{N\sigma_v^2} = 17.3227dB$,



(a) (b)

Fig. 3. Performance of the sparse signal detection and sparsity pattern recovery with Algorithm 3; $N = 256$, $K = 10$, $L = 10$, $\bar{\gamma} = \frac{||\mathbf{s}||_2^2}{N\sigma_v^2} = 17.3227dB$; (a) Probability of detection, (b) probability of false alarms

other approaches, we consider two existing benchmark cases. (i). Distributed OMP with no collaboration: in this case, each node performs standard OMP in Algorithm 1 independently to obtain the support set estimate $\hat{\mathcal{U}}_l$. To fuse the estimated support sets, $\hat{\mathcal{U}}_l$'s, at individual nodes, each node transmits indices in $\hat{\mathcal{U}}_l$ to a fusion center and performs a majority rule based fusion scheme to obtain a global estimate $\hat{\mathcal{U}}$. (ii). Simultaneous OMP (S-OMP) [32]: S-OMP algorithm is carried out using all the raw observations (as well as the projection matrix $\mathbf{A}_l$) at the fusion center.

In Figures 1-3 we assume that $\mathcal{M}_l = \bar{\mathcal{M}}$ as considered in case I in Subsection 3.1.1. Then the estimated support set at each node based on Algorithm 2 is the same. Entries of each projection matrix $\mathbf{A}_l$ for $l = 0, 1, \cdots, L-1$ are drawn from a Gaussian ensemble with mean zero and variance $\frac{1}{N}$. In Fig. 1, by performing 10^4 runs and averaging over 20 trials, we plot the probability of correctly recovering the full support set, $P_D = Pr(\hat{\mathbf{b}} = \mathbf{b})$ (a) and the percentage of the support set that is estimated correctly (b) vs M/N where M is the number of compressive measurements at each node. It can be seen from Fig. 1 that, at relatively small values of M/N, the proposed algorithm outperforms D-OMP with no collaboration. In resource constrained distributed networks, especially in sensor networks, it is desirable to perform the desired task by employing less measurement data (i.e. with small M) at each node distributing the computational complexity among nodes to save the overall node power, and the proposed algorithm is promising in this case compared to performing OMP at each independently (based on Algorithm 1) and then fusion.

However, as M/N increases, $Pr(\hat{\mathbf{b}} = \mathbf{b})$ from both Algorithms 1 and 2 converge to 1, since when the number of compressive measurements at each node increases, OMP (with or without collaboration) works better and recovers the sparsity pattern almost exactly with a single measurement vector. Moreover, even for small M/N, S-OMP provides a significant performance gain compared to the proposed Algorithm 2. It has been shown in [8, 37] that OMP requires approximately $M \geq (1+\epsilon)4K\log N$, $\epsilon > 0$ measurements for reliable sparsity pattern recovery in the noise free case. Thus, since there are L nodes, this limits is achieved even with small M at a given node. However, S-OMP requires a considerable communication overhead compared to the proposed Algorithm 2. Further, in the proposed algorithm, each node has the same estimator at the end in contrast to the centralized S-OMP.

In Fig. 1, we further plot the percentage of support that is correctly recovered. for example, at $\frac{M}{N} \approx 0.1$, the proposed algorithm correctly recovers approximately 75% of the support while D-OMP with no collaboration recovers only about 30% of the support. To further illustrate the efficiency of the proposed algorithm, in Fig. 2, we plot the number of iterations of the DC-OMP algorithm that each node has to perform in recovering the sparsity pattern. It is observed from Fig. 2 that as M/N increases, the proposed algorithm estimates the sparsity pattern reliably by executing only $\approx K/2$ number of iterations at each node. When M/N increases, as observed from Fig. 1, the performance of both DC-OMP and D-OMP with no collaboration converges to the same level. However, DC-OMP requires very small number of iterations at each node to achieve that performance compared to that with D-OMP with no collaboration which requires K number of iterations at each node irrespective of the value of M/N.

In Fig. 3, we illustrate the performance of Algorithm 3 for detecting the sparse signal before estimating the sparsity pattern. We plot the performance of sparse signal detection as well as the sparsity pattern estimation in Fig. 3. For sparse signal detection, probability of detection and the false alarm are given by $P_D^s = Prob(\delta = 1|\mathcal{H}_1)$ and $P_F^s = Prob(\delta = 1|\mathcal{H}_0)$, respectively where δ is the binary detection decision. The probability of sparsity pattern detection is given by $P_D^u = Prob(\hat{\mathbf{b}} = \mathbf{b}^1|\mathcal{H}_1)Prob(\mathcal{H}_1) + Prob(\hat{\mathbf{b}} = \mathbf{b}^0|\mathcal{H}_0)Prob(\mathcal{H}_0)$, where we redefine the variables such that $\mathbf{b}^1$ is the binary support of the signal $\mathbf{s}$ (i.e. the support under $\mathcal{H}_1$) while $\mathbf{b}^0$ denotes the vector with all zeros (binary support under $\mathcal{H}_0$). With respect to sparsity pattern recovery, the probability of false alarm is given by $P_F^u = Prob(\hat{\mathbf{b}} = \mathbf{b}^1|\mathcal{H}_0)Prob(\mathcal{H}_1) + Prob(\hat{\mathbf{b}} = \mathbf{b}^0|\mathcal{H}_1)Prob(\mathcal{H}_0)$. In Fig. 3 we use the same values for the parameters N, K, L and $\bar{\gamma}$ as used in Figures 1 and 2 and set $K_0 = 3$ and $I_0 = 2$. From Fig. 3, it is seen that Algorithm 3 reliably detects the sparse signal even with a very small value of M/N, and the performance of the sparsity pattern recovery after detecting the signal has performance that is close to that when the sparsity pattern recovery is done as in Algorithm 2 (where it is known *a priori* that the signal is present).

6. CONCLUSION

We addressed the problem of recovering a common sparsity pattern based on CS measurement vectors collected at distributed nodes in a distributed network. A distributed greedy algorithm based on OMP is proposed to estimate the sparsity pattern via collaboration in which each distributed node is required to perform less number of iterations of the greedy algorithm compared to the sparsity index. When it is not known *a priori* that a sparse signal is present or not, the algorithm was extended to perform detection of the sparse signal with a fewer number of iterations before completely recovering the sparsity pattern. The proposed algorithm is shown to have significant performance gains compared to that with each node performing OMP independently and then fusing the estimated supports to achieve a global estimate. Complete theoretical performance and complexity analysis of the algorithm will be considered in a future work.

7. REFERENCES

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

[2] E. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, pp. 21–30, Mar. 2008.

[3] R. G. Baraniuk, "Compressive sensing," *IEEE Signal Processing Magazine*, pp. 118–120, 124, Jul. 2008.

[4] E. Candès and J. Romberg, "Practical signal recovery from random projections," *Preprint*, Jan. 2004.

[5] E. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, Aug. 2006.

[6] J. K. Romberg, "Sparse signal recovery via l1 minimization," in *40th Annual Conf. on Information Sciences and Systems (CISS)*, Princeton, NJ, Mar. 2006, pp. 213–215.

[7] J. N. Laska, M. A. Davenport, and R. G.Baraniuk, "Exact signal recovery from sparsely corrupted measurements through the pursuit of justice," in *Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, California, Nov. 2009, pp. 1556–1560.

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

[9] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing*, vol. 1, no. 4, 2007.

[10] T. Blumensath and M. E. Davies, "Iterative thresholding for sparse approximations," *Preprint*, 2007.

[11] ——, "Gradient pursuits," *IEEE Trans. Signal Processing*, vol. 56, no. 6, pp. 2370–2382, June 2008.

[12] D. Needell and R. Vershynin, "Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit," *Preprint*, 2007.

[13] D. Malioutov, M. Cetin, and A.Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Trans. Signal Processing*, vol. 53, no. 8, pp. 3010–3022, Aug. 2005.

[14] V. Cevher, P. Indyk, C. Hegde, and R. G. Baraniuk, "Recovery of clustered sparse signals from compressive measurements," in *Int. Conf. Sampling Theory and Applications (SAMPTA 2009)*, Marseille, France, May. 2009, pp. 18–22.

[15] Z. Tian and G. Giannakis, "Compressed sensing for wideband cognitive radios," in *Proc. Acoust., Speech, Signal Processing (ICASSP)*, Honolulu, HI, Apr. 2007, pp. IV–1357–IV–1360.

[16] Y. Jin, Y.-H. Kim, and B. D. Rao, "Support recovery of sparse signals," *arXiv:1003.0888v1[cs.IT]*, 2010.

[17] S. Baillet, J. C. Mosher, and R. M. Leahy, "Electromagnetic brain mapping," *IEEE Signal Process. Mag.*, pp. 14–30, 2001.

[18] D. Wipf and S. Nagarajan, "A unified bayesian framework for meg/eeg source imaging," *NeuroImage*, pp. 947–966, 2008.

[19] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Computing*, vol. 24, no. 2, pp. 227–234, 1995.

[20] A. J. Miller, *Subset Selection in Regression*. New York, NY: Chapman-Hall, 1990.

[21] E. G. Larsson and Y. Seln, "Linear regression with a sparse parameter vector," *IEEE Trans. Signal Processing*, vol. 55, no. 2, pp. 451–460, Feb.. 2007.

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

[23] M. J. Wainwright, "Information-theoretic limits on sparsity recovery in the high-dimensional and noisy setting," *Univ. of California, Berkeley, Dept. of Statistics, Tech. Rep*, Jan. 2007.

[24] ——, "Information-theoretic limits on sparsity recovery in the high-dimensional and noisy setting," *IEEE Trans. Inform. Theory*, vol. 55, no. 12, pp. 5728–5741, Dec. 2009.

[25] W. Wang, M. J. Wainwright, and K. Ramachandran, "Information-theoretic limits on sparse signal recovery: Dense versus sparse measurement matrices," *IEEE Trans. Inform. Theory*, vol. 56, no. 6, pp. 2967–2979, Jun. 2010.

[26] A. K. Fletcher, S. Rangan, and V. K. Goyal, "Necessary and sufficient conditions for sparsity pattern recovery," *IEEE Trans. Inform. Theory*, vol. 55, no. 12, pp. 5758–5772, Dec. 2009.

[27] M. M. Akcakaya and V. Tarokh, "Shannon-theoretic limits on noisy compressive sampling," *IEEE Trans. Inform. Theory*, vol. 56, no. 1, pp. 492–504, Jan. 2010.

[28] G. Reeves and M. Gastpar, "Sampling bounds for sparse support recovery in the presence of noise," in *IEEE Int. Symp. on Information Theory (ISIT)*, Toronto, ON, Jul. 2008, pp. 2187–2191.

[29] G. Tang and A. Nehorai, "Performance analysis for sparse support recovery," *IEEE Trans. Inform. Theory*, To appear 2010.

[30] V. K. Goyal, A. K. Fletcher, and S. Rangan, "Compressive sampling and lossy compression," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 48–56, Mar. 2008.

[31] T. Wimalajeewa and P. K. Varshney, "Performance bounds for sparsity pattern recovery with quantized noisy random projections," *IEEE Journal of Selected Topics in Signal Processing, Special Issue on Robust Measures and Tests Using Sparse Data for Detection and Estimation*, vol. 6, no. 1, pp. 43 – 57, Feb. 2012.

[32] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Simultaneous sparse approximation via greedy pursuit," in *Proc. Acoust., Speech, Signal Processing (ICASSP)*, 2005, pp. V–721–V–724.

[33] D. Baron, M. Duarte, S. Sarvotham, M. B. Wakin, and R. G. Baraniuk, "Distributed compressed sensing," *preprint*, 2009.

[34] Q. Ling and Z. Tian, "Decentralized support detection of multiple measurement vectors with joint sparsity," in *Proc. Acoust., Speech, Signal Processing (ICASSP)*, 2011, pp. 2996–2999.

[35] F. Zeng, C. Li, and Z. Tian, "Distributed compressive spectrum sensing in cooperative multihop cognitive networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 1, pp. 37–48, Feb. 2011.

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

[37] A. Fletcher and S. Rangan, "Orthogonal matching pursuit from noisy random measurements: A new analysis," in *NIPS'09*, 2009.

[38] M. J. Wainwright, "Sharp thresholds for high-dimensional and noisy recovery of sparsity," in *Technical Report*, Univ. of California, Berkeley, Dept. of Statistics, May 2006.