

Incremental and diffusion compressive sensing strategies over distributed networks

Ghanbar Azarnia*¹, Mohammad Ali Tinati ², Abbas Ali Sharifi ³, and Hamid Shiri ⁴

¹*Engineering Faculty of Khoy, Urmia University, Urmia, Iran*

²*Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran*

³*Department of Electrical Engineering, University of Bonab, Bonab, Iran*

⁴*Center for Wireless Communications, University of Oulu, Oulu, Finland*

Abstract

Compressive sensing (CS) has been widely used in wireless sensor networks (WSNs). In WSNs, the sensors are battery-powered and hence their communication and processing powers are limited. One of the dominant features of the CS is its complex recovery phase. Thus, great care should be taken into account when designing the CS recovery algorithm for WSNs. In this paper, we propose a distributed and cooperative recovery algorithm for two different cooperation modes of sensor networks including incremental and diffusion. The theoretical performance analysis of the proposed algorithms in both exact and noisy measurements is investigated. The obtained results show the superiority of the proposed method in terms of convergence rate and steady-state error compared with the non-cooperative scenario and the well-known distributed least absolute shrinkage and selection operator (D-LASSO) approach. Furthermore, the proposed structure requires much fewer measurements for exact recovery.

Keywords: Compressive Sensing, Wireless sensor networks, Incremental and diffusion strategy, Sparse signal.

1. Introduction

Compressive sensing (CS) has been emerged into signal processing and received great attention. The CS approach, introduced by Candes, Romberg, Tao [1], and Donoho [2], includes a very simple sensing phase and a complex

recovery phase. In the simple compression phase, the desired signal with a large number of entries is sensed or measured by an appropriate rectangular matrix and results in a short length of probable noisy or inaccurate measurement. In the sophisticated recovery phase, the desired signal is reconstructed from the measured data. When the signal is sparse or equivalently most of its components are zero, there are a lot of practical and efficient algorithms for reconstruction of the signal. The CS approach is beneficial in many applications since many practical signals are compressible in the sense that they can be approximated as a sparse signal by applying an appropriate transformation [3].

In practice, the availability of fast reconstruction algorithms is essential. The reconstruction algorithms are usually based on convex optimization, non-convex optimization, and greedy approaches. In convex optimization, the number of measurements for signal reconstruction is small but the computational complexity is relatively high. Basis pursuit [4], basis pursuit denoising (BPDN) [4], least absolute shrinkage and selection operator (LASSO) [5], and least angle regression (LAR) [6] algorithms are based on convex optimization. The non-convex optimizations, such as iterative re-weighted least squares [7], sparse Bayesian learning algorithms [8], Monte-Carlo based algorithms [9], and focal underdetermined system solver (FOCUSS) classes of algorithms [10]-[14] are hard to exactly solve in a reasonable time. Alternative reconstruction approaches include greedy type methods that have fast reconstruction rate, low complexity of mathematical framework, and simple geometric interpretation. These types of algorithms iteratively solve the reconstruction problem step by step. The basic idea is to find the support (the index set of its nonzero entries) of the unknown signal sequentially. Determining the correct support set, the non-zero signal coefficients are calculated by applying the pseudo-inversion process [15]-[27].

One of the most important areas of CS applications is in wireless sensor networks (WSNs) [28]. The WSNs are used in many applications such as physiological monitoring, environmental monitoring, condition-based maintenance, smart spaces, military, precision agriculture, transportation, factory instrumentation and inventory tracking [29].

All of the described algorithms in [1]-[27] can be used in WSNs with fusion center (FC) where the central processor performs the CS reconstruction tasks. In this scenario, the high aggregate data rates limit the bandwidth availability and the battery-powered devices restrict the communication energy. So, distributed processing is often much desirable for such situations.

In a distributed scenario, without requiring any FC, each sensor communicates only with its closest neighbors and the processing tasks are carried out locally at each sensor node [30]-[32]. Since most of the existing sparse optimization algorithms are performed in a centralized scheme, these algorithms could only provide effective performance for solving the sparse signal recovery problem based on a single measurement vector, which is not suitable for WSNs' applications.

A most valuable joint distributed sparse optimization algorithm is given in [33], where the above-mentioned issues are considered and three decentralized methods are proposed. The first proposed algorithm of [33] is an iterative method and aims to solve a quadratic program in each iteration. So, the computational complexity of this algorithm is not suitable for WSNs. To reduce the computational complexity, the authors proposed a distributed coordinate descent LASSO (DCD-LASSO) algorithm. However, in each iteration, the update of each coordinate is related to the previous coordinate and cannot be performed in parallel. So, this algorithm suffers from a slow convergence rate. To overcome this issue, the authors in [33] developed D-LASSO to update all coordinates in parallel. This leads potentially to fast convergence rate, but this algorithm involves matrix inversion that may be computationally demanding for sufficiently large matrices. Thus, all of the proposed algorithms in [33] are based on convex optimization and are not suitable for WSNs due to their high computational complexities and low convergence rates.

A distributed form of the CS is also considered in [34]-[36]. In [34], each sensor first performs a *local computation* in each iteration to derive an intermediate vector. The sensors then perform a *global computation* on their intermediate vectors to derive the next step in the iteration. Because of the global computation step, this method requires a spanning tree over the network rooted at a special node. This special node also should have some information about other nodes that must be trained using a distributed algorithm. Therefore, the amount of computations and communications of this method is extremely high and it is not proper for WSNs. A BP-based algorithm is proposed in [35]. Each node solves an optimization problem in each iteration. So, the computational complexity of this algorithm is also high and not proper for WSNs. This algorithm requires an additional computation for graph coloring. Another disadvantage of this algorithm is that there is no convergence guarantee in a general network. In [36], an algorithm based on concave penalization is introduced. The proposed method

significantly reduces the data exchange by limiting communications to local communications. The main drawback of the algorithm is its very low convergence rate. This kind of convergence is not suitable for energy-constrained WSNs, because, the speed of operations is crucial for some applications.

In this paper, we propose a greedy-based algorithm that has fast reconstruction and low computational complexity. This algorithm is an extension of hard thresholding pursuit (HTP) [26] assigned to use with a distributed and cooperative scenario. We propose a distributed and cooperative algorithm in two topology structures including incremental and diffusion. In an incremental strategy, the information flows sequentially from one node to the adjacent node. This mode of cooperation requires a cyclic pattern of cooperation among the nodes and it requires fewer communications and power sources [30]. When more communications and energy resources are available, a diffusion cooperative scheme can be applied, where each node communicates with all of their neighbors and no cyclic path is required. In this case, the amount of communications is higher than the incremental scheme. But, nodes can access to more data from their neighbors [31]-[32]. The proposed greedy-based algorithm is intended to adapt for these two modes of cooperation. In the proceeding section, we present the theoretical performance analysis of the proposed algorithms in two cooperative modes. Although some of our analysis are similar to those of [26], we consider the distributive case that is challenging since nodes in each neighborhood interact with each other. In such a scenario, a successful analysis should consider both the temporal and spatial interconnectedness of the data. The suggested approach is quite different from the methods reported in [37]-[40]. The main contribution of [37]-[40] is an adaptive estimation in a distributed scenario, where the sparse property is considered to improve the performance of the distributed adaptive estimation. In these works, there is also no limitation in the number of measurements and they only impose a penalty in the intended cost functions to increase the accuracy and speed of estimation for sparse signals. On the contrary, we encounter an underdetermined system of linear equations and exploit the sparsity of data to solve these equations.

The key contributions of this study are summarized as follows:

- A recovery algorithm based on the incremental and diffusion cooperation modes of the sensor network is proposed. These cooperation modes have lower steady-state error and higher convergence rate than the non-cooperative methods.

- We present the theoretical performance analysis of the proposed algorithms. Compared with the non-distributed case, the theoretical analysis of the proposed approach is challenging since nodes in each neighborhood interact with each other and a successful analysis should consider the temporal and spatial interconnectedness of the data.
- We show that the convergence rate of the proposed method is higher than that of the non-cooperative case.
- Using some theorems, we show sufficient condition on the restricted isometry property (RIP) for the linear system matrix that the proposed incremental and diffusion algorithms converge. This condition is better than the non-distributed counterpart.
- Under some theorems, we present the *stability* of the reconstruction scheme regarding the sparsity defect for both incremental and diffusion modes of cooperation.
- We show the *robustness* of the reconstruction scheme regarding the measurement error.

Notation: For ease of reference, the main symbols used in this paper are listed in Table 1:

2. Preliminaries

In CS algorithms, we find the sparse solution $\mathbf{x} \in \mathbb{C}^N$ of underdetermined linear system $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$, where $\mathbf{A} \in \mathbb{C}^{m \times N}$ ($m < N$) is the sensing matrix, $\mathbf{y} \in \mathbb{C}^m$ is a noisy measurement, and $\mathbf{e} \in \mathbb{C}^m$ is an unknown noise vector. The most common way to reconstruct the sparse signals is solving an optimization problem as follows:

$$\underset{\mathbf{z} \in \mathbb{C}^N}{\text{minimize}} \|\mathbf{z}\|_0 \quad \text{subject to } \mathbf{A}\mathbf{z} = \mathbf{y} \quad (1)$$

where $\|\mathbf{z}\|_0$ is the support cardinality of \mathbf{z} . Finding the solution of the non-convex problem (1) is NP-hard and computationally intractable [41]. Several practical methods were presented in the CS area [1]-[33]. All of these methods are successful when the measurement matrix \mathbf{A} is properly selected. One simple way to measure the suitability of the measurement matrix is to

Table 1: The main symbols

Symbol	Description
$\ \mathbf{A}\ _F$	The Frobenius norm of a matrix \mathbf{A}
$\ \mathbf{A}\ _{2 \rightarrow 2}$	The spectral norm of a matrix \mathbf{A}
$\ \mathbf{x}\ _2$	The ℓ_2 -norm of a vector \mathbf{x}
$\ \mathbf{x}\ _0$	The number of nonzero entries of a vector \mathbf{x}
$ x $	The absolute value of a scalar x
$tr(\mathbf{A})$	The trace (the sum of diagonal elements) of a matrix \mathbf{A}
$supp(\mathbf{x})$	The support of a vector \mathbf{x}
$card(S)$	The cardinality of a set S
$\langle \mathbf{u}, \mathbf{v} \rangle$	The inner product between two vectors \mathbf{u} and \mathbf{v}
$E\{X\}$	The expectation of a random variable X
\mathbf{A}^*	The adjoint (or Hermitian transpose) of a matrix \mathbf{A}
$\mathbf{A} \otimes \mathbf{B}$	The Kronecker product of two matrices \mathbf{A} and \mathbf{B}
\mathbf{I}	The identity matrix
\bar{S}	The complement of a set S
$T \Delta S$	The symmetric difference of the sets T and S
$S \setminus T$	The difference of set T from set S
\mathbf{z}_S	The vector equal to \mathbf{z} on S and to zero on \bar{S}
$L_s(\mathbf{z})$	The index set of s largest absolute entries of a vector \mathbf{z}
$\mathcal{L}_s^J(Z)$	An operator that divides the Z into J blocks and returns the index set of s largest absolute entries of each block
$diag\{.\}$	The block diagonal matrix consisting of the specified matrices
$col\{.\}$	The column vector with the specified entries stacked on top of each other

use coherence [42]. Assuming \mathbf{a}_i is the i th column of \mathbf{A} and all columns of \mathbf{A} are ℓ_2 normalized, the coherence of the matrix \mathbf{A} is defined as

$$\mu = \max_{1 \leq i \neq j \leq N} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle| \quad (2)$$

In general, the smaller coherence results in a better recovery algorithm's performance. The coherence of a matrix with ℓ_2 -normalized columns satisfies the following condition [43]

$$\mu \geq \sqrt{\frac{N-m}{m(N-1)}} \quad (3)$$

The lower bound on the coherence limits the performance of recovery algorithms to the small sparsity levels. The sparsity level is the number of non-zero entries of the signal. Another measure for the quality of the measurement matrix is RIP [44]. The s 'th restricted isometry constant δ_s for a matrix \mathbf{A} is the smallest $\delta \geq 0$ such that

$$(1 - \delta) \|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1 + \delta) \|\mathbf{x}\|_2^2 \quad (4)$$

The above equation is valid for all s -sparse vectors $\mathbf{x} \in \mathbb{R}^N$. The signal is s -sparse when the maximum number of its nonzero entries is s . The matrix \mathbf{A} is said to satisfy the RIP condition if δ_s is small for reasonably large s . Bounds as $\delta_{\kappa s} < \delta_*$ for some integer κ and some specific value δ_* guarantee the recoverability of s -sparse vectors via different algorithms. Testing RIP condition for a matrix is NP-hard and computationally intractable [45]. However, it has been established that certain classes of randomly generated matrices satisfy the RIP with a high probability. It is well known that the random matrices satisfy the condition $\delta_{\kappa s} < \delta_*$ when the number of measurements has the following condition

$$m \geq C \frac{\kappa s}{\delta_*^2} L n \left(\frac{N}{s} \right) \quad (5)$$

for some constant $C > 0$

3. The proposed approach

In this section, the proposed algorithm is investigated. First, we explain the incremental case and then the diffusion strategy is discussed.

3.1. Incremental compressive sensing strategy

In this section, we propose a compressive sensing recovery method for WSNs based on an incremental strategy. We extend the HTP algorithm for a distributed framework where the information flows sequentially from one node to the adjacent node. The proposed algorithm is called distributed incremental hard thresholding pursuit (DIHTP) through the study. We essentially focus on HTP for many reasons. First, it has a better performance compared with the other algorithms. Second, it is computationally suitable for WSNs, and third, the HTP is compatible with the cyclic patterns of cooperative nodes. In the second part of this section, we analyze the recovery performance of DIHTP in the noise-free scenario and then extend the analysis to the case that the vectors are not exactly sparse and can not be measured with perfect precision. Such an analysis for the incremental topology-based networks is challenging because nodes in each neighborhood interact with each other and consequently, a successful analysis should consider both the temporal and spatial interconnectedness of the data.

3.1.1. DIHTP algorithm

We consider J numbers of sensors are randomly distributed over a region. Let \mathcal{N}_i denote a set of all sensors in the neighborhood of node i . The objective is to collectively recover s -sparse vector $\mathbf{x} \in \mathbb{C}^N$ from their measurements

$$\mathbf{y}_i = \mathbf{A}_i \mathbf{x} + \mathbf{e}_i \in \mathbb{C}^m \quad (6)$$

where $\mathbf{A}_i \in \mathbb{C}^{m \times N}$ is a known matrix and $\mathbf{e}_i \in \mathbb{C}^m$ is an unknown noise vector with $\|\mathbf{e}_i\| < \sigma_i$, for some $\sigma_i > 0$ and some norm on \mathbb{C}^m , usually the ℓ_2 -norm at node i .

A non-cooperative approach, to recover the s -sparse vector \mathbf{x} from the measurement \mathbf{y}_i , proposes running of the reconstruction algorithms for each node separately [1]-[27]. But, such algorithms do not use the spatial correlation between the nodes. If the spatial correlation is used, it increases the speed of convergence. For this purpose, we modify the first stage of HTP [26] as

$$S_j^n = L_s (\mathbf{x}_{j-1}^n + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n)) \quad (7)$$

where $L_s(\mathbf{z})$ returns the index set of s largest absolute entries of \mathbf{z} . $\mathbf{x}_j^n \in \mathbb{C}^N$ and S_j^n are respectively the local estimation of \mathbf{x} and its support set in the n 'th iteration of sensor j . The node j uses the estimation \mathbf{x}_j^{n-1} of its neighbor

in order to update the local estimation of its support set S_j^n . The justification is that the RIP ensures the matrix $\mathbf{A}_j^* \mathbf{A}_j$ behaves as an identity matrix for each node. Thus, the contributions of \mathbf{x}_j^{n-1} and $\mathbf{A}_j^* \mathbf{A}_j \mathbf{x}_j^{n-1}$, despite the use of local estimation of the previous node, is roughly $\mathbf{x}_j^{n-1} - \mathbf{x}_j^{n-1} = 0$ and the term $\mathbf{A}_j^* \mathbf{y}_j$ is equivalent to \mathbf{x}_j , because of $\mathbf{A}_j^* \mathbf{A}_j \mathbf{x}_j = \mathbf{A}_j^* \mathbf{y}_j$.

The suggested DIHTP algorithm can be written as follows:

$$\begin{aligned}
& \text{For each time } n \geq 0, \text{ repeat} \\
& j = 1, \dots, J \\
& \mathbf{x}_0^n = \mathbf{x}^{n-1} \\
& S_j^n = L_s(\mathbf{x}_{j-1}^n + \mathbf{A}_j^*(\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n)) \\
& \mathbf{x}_j^n = \operatorname{argmin}_{\mathbf{z} \in \mathbb{C}^N} \{ \|\mathbf{y}_j - \mathbf{A}_j \mathbf{z}\|_2, \operatorname{supp}(\mathbf{z}) \subset S_j^n \} \\
& \mathbf{x}^n = \mathbf{x}_j^n
\end{aligned} \tag{8}$$

where “*supp*” is the abbreviation of support. The operation of the algorithm (8) is as follows: At each time instant n , each node uses its local data and matrix, $\{\mathbf{y}_j, \mathbf{A}_j\}$, and the estimated vector \mathbf{x}_{j-1}^n , received from its adjacent node, to perform the following tasks:

1. Evaluate the local estimate $\mathbf{x}_{j-1}^n + \mathbf{A}_j^*(\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n)$.
2. Choose the index set of s largest absolute entries of $\mathbf{x}_{j-1}^n + \mathbf{A}_j^*(\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n)$.
3. Update the estimated vector $\mathbf{x}_j^n = \operatorname{argmin}_{\mathbf{z} \in \mathbb{C}^N} \{ \|\mathbf{y}_j - \mathbf{A}_j \mathbf{z}\|_2, \operatorname{supp}(\mathbf{z}) \subset S_j^n \}$.
4. Pass the updated vector \mathbf{x}_j^n to the neighbor node $j + 1$.

In the proceeding section, we show that the DIHTP approach has a better convergence rate than HTP [26]. The HTP is a non-distributed implementation and it is separately performed in each node.

3.1.2. Convergence analysis of DIHTP in the noise-free case

Theorem 1: The DIHTP algorithm generates sequence \mathbf{x}_j^n that satisfies the condition $\|\mathbf{x}_j^n - \mathbf{x}\|_2 \leq \rho^n \|\mathbf{x}_j^0 - \mathbf{x}\|_2$ given $\mathbf{y}_i = \mathbf{A}_i \mathbf{x}$, where $\rho =$

$$\prod_{j=1}^J \sqrt{\frac{2(\delta_{3s}^j)^2}{1 - (\delta_{2s}^j)^2}} \leq 1.$$

Proof of Theorem 1: see Appendix A.

The main finding of this theory is that the sequence \mathbf{x}_j^n converges towards \mathbf{x} in a geometric rate which is almost improved by the power of J compared with the non-cooperative cases. Generally, this Theorem is interpreted from

two points of views. First, if we assume, without loss of generality, that the restricted isometry constants σ_{3s}^j and σ_{2s}^j are equal for all nodes and are chosen such that $\sqrt{\frac{2(\delta_{3s}^j)^2}{1-(\delta_{2s}^j)^2}} < 1$, then $\|\mathbf{x}_j^n - \mathbf{x}\|_2 \leq \left(\sqrt{\frac{2(\delta_{3s}^j)^2}{1-(\delta_{2s}^j)^2}}\right)^J \|\mathbf{x}_j^{n-1} - \mathbf{x}\|_2$ is true (see (A.14) in Appendix A), consequently the convergence rate of DIHTP is better than the conventional non-distributed HTP. Second, note that if for a certain node in the network we choose the restricted isometry constants such that $\frac{2(\delta_{3s}^j)^2}{1-(\delta_{2s}^j)^2} < a$ holds and if for the restricted isometry constants of another node k in the network $\frac{2(\delta_{3s}^k)^2}{1-(\delta_{2s}^k)^2} < \frac{1}{a}$ holds then the products of them will be $\frac{2(\delta_{3s}^j)^2}{1-(\delta_{2s}^j)^2} \frac{2(\delta_{3s}^k)^2}{1-(\delta_{2s}^k)^2} < 1$, which satisfies the convergence condition. On the other hand, since $\delta_{2s}^j \leq \delta_{3s}^j$, these occur as soon as the inequalities $\delta_{3s}^j < \sqrt{\frac{a}{2+a}}$ and $\delta_{3s}^k < \sqrt{\frac{1}{1+2a}}$ hold. If we choose $a = 1$, then the same convergence condition of the single signal case (i.e., $\delta_{3s}^j < \frac{1}{\sqrt{3}}$) will be achieved. But, for example when $a = 2$, we have $\delta_{3s}^j < \frac{1}{\sqrt{2}}$ and $\delta_{3s}^k < \frac{1}{\sqrt{5}}$. The bound $\delta_{3s}^j < \frac{1}{\sqrt{2}}$ offers a substantial improvement over the single signal case in the cost of the more stringent condition for the other node k . So, according to equation (5), the node j requires fewer measurements than the node k . In other words, deficiency of one node's measurements is compensated by another node.

3.1.3. Convergence analysis of DIHTP for approximately sparse vectors measured with some errors

Theorem 2: For any signal $\mathbf{x} \in \mathbb{C}^N$ that is not be exactly sparse with the noisy measurements $\mathbf{y}_j = \mathbf{A}_j \mathbf{x} + \mathbf{e}_j$, the DIHTP algorithm generates sequence \mathbf{x}_j^n which satisfies $\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 \leq (\Pi_{j,1})^n \|\mathbf{x}_j^0 - \mathbf{x}_S\|_2 + (\Pi_{j,1})^{n-1} \mathcal{H}_j$, where S denotes the index set of s largest entries of \mathbf{x} ,

$\Pi_{j,K} = \beta_j \beta_{j-1} \dots \beta_1 \beta_J \beta_{J-1} \dots \beta_{j+K}$, and $\mathcal{H}_j = \Pi_{j,2} \mathcal{F}_{j+1} + \Pi_{j,3} \mathcal{F}_{j+2} + \dots + \Pi_{j,J-1} \mathcal{F}_{j-2} + \Pi_{j,J} \mathcal{F}_{j-1} + \mathcal{F}_j$ with

$$\beta_j = \sqrt{\frac{2(\delta_{3s}^j)^2}{1-(\delta_{2s}^j)^2}} \text{ and } \mathcal{F}_j = \left(\sqrt{\frac{2}{1-\delta_{2s}^j}} + \frac{\sqrt{1+\delta_{2s}^j}}{1-\delta_{2s}^j} \right) \|\mathbf{A}_j \mathbf{x}_{\bar{S}} + \mathbf{e}_j\|_2.$$

Proof of Theorem 2: see Appendix B.

This theorem has an important implication which indicates the *robust-*

ness and *stability* of the algorithm. As it can be seen from theorem 2, the reconstruction error is controlled by the desired signal distance to a s -sparse vector. In the CS literature, it is usually referred to as the *stability* of the reconstruction scheme to sparsity defect. On the other hand, the results of this theorem state that the distance of the reconstructed signal to the original signal is controlled by the measurement error. In the CS literature, it is usually referred to as the *robustness* of the reconstruction scheme to the measurement error. So, theorem 2 indicates the *robustness* and *stability* of the algorithm even in the distributed case. This theorem also shows that on the contrary to the single signal case, in addition to the errors caused by the node, the errors resulting from the inexact sparsity and imperfect precision in the measuring of all other nodes are involved. In addition, in term of the convergence, by restricting $\Pi_{j,1} = \prod_{j=1}^J \sqrt{\frac{2(\delta_{3s}^j)^2}{1-(\delta_{2s}^j)^2}} < 1$, we achieve the same result as the exact case. Therefore, all the issues mentioned in theorem 1 are also confirmed in this section.

3.2. Diffusion compressive sensing strategy

In this section, we propose a CS recovery method for WSNs based on the diffusion mode of cooperation. Then, we analyze the performance of the proposed DDHTP recovery algorithm in the noise-free case and the case with inexact sparsity and imperfect precision setting.

3.2.1. DDHTP algorithm

We assume a network with more communications and available energy resources. In the considered network, each node communicates with all of its neighbors and uses the linear combination of its neighborhood estimates rather than using only the estimation of one node. This process could be expressed as follows:

$$\varphi_j^{n-1} = \sum_{\ell \in \mathcal{N}_j} c_{j,\ell} \mathbf{x}_\ell^{n-1} \quad (9)$$

where the neighborhood parameter \mathcal{N}_j is defined as the set of node j and all its linked nodes. The constant $c_{j,\ell}$ is combination coefficient and it is the element of combination matrix $C = [c_{j,\ell}]$. The matrix C carries the information about the network topology: a nonzero entry $c_{j,\ell}$ means that nodes j and ℓ are connected. We restrict the combining coefficients to be $\sum_{\ell \in \mathcal{N}_j} c_{j,\ell} = 1$. The proposed diffusion strategy is described as

$$\begin{aligned}
\boldsymbol{\varphi}_j^{n-1} &= \sum_{\ell \in \mathcal{N}_j} c_{j,\ell} \mathbf{x}_\ell^{n-1} \\
S_j^n &= L_s (\boldsymbol{\varphi}_j^{n-1} + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \boldsymbol{\varphi}_j^{n-1})) \\
\mathbf{x}_j^n &= \operatorname{argmin}_{\mathbf{z} \in \mathbb{C}^N} \{ \|\mathbf{y}_j - \mathbf{A}_j \mathbf{z}\|_2, \operatorname{supp}(\mathbf{z}) \subset S_j^n \}
\end{aligned} \tag{10}$$

each node uses the linear combination of its neighborhood estimates to accelerate the convergence rate of the algorithm.

3.2.2. Convergence analysis of DDHTP in the noise-free scenario

For the convergence analysis, we first introduce the following quantities:

$$\begin{aligned}
X^n &= \operatorname{col} \{ \mathbf{x}_1^n, \dots, \mathbf{x}_J^n \} & \mathbf{A} &= \operatorname{diag} \{ \mathbf{A}_1, \dots, \mathbf{A}_J \} \\
Y &= \operatorname{col} \{ \mathbf{y}_1, \dots, \mathbf{y}_J \} & \mathcal{S}^n &= \operatorname{col} \{ S_1^n, \dots, S_J^n \} \\
X^\# &= \operatorname{col} \{ \mathbf{x}, \dots, \mathbf{x} \} & \Phi^n &= \operatorname{col} \{ \boldsymbol{\varphi}_1^n, \dots, \boldsymbol{\varphi}_J^n \}
\end{aligned} \tag{11}$$

considering the above-definitions, the equation (10) is rewritten as

$$\begin{aligned}
\Phi^{n-1} &= GX^{n-1} \\
S^n &= \mathcal{L}_s^J (\Phi^{n-1} + \mathbf{A}^* (Y - \mathbf{A} \Phi^{n-1})) \\
X^n &= \operatorname{argmin}_{\mathbf{z} \in \mathbb{C}^{NJ}} \{ \|Y - \mathbf{A}Z\|_2, \operatorname{supp}(Z) \subset \mathcal{S}^n \}
\end{aligned} \tag{12}$$

where $G = \mathbb{C} \otimes I_N$ and the operator \otimes denotes the Kronecker product. The notation $\mathcal{L}_s^J(Z)$ is an operator that divides the Z into J blocks and returns the index set of s largest absolute entries of each block.

Theorem 3: The DDHTP algorithm generates convergence sequence as $\|X^n - X^\#\|_2 \leq \rho^n \|X^0 - X^\#\|_2$ given $\mathbf{y}_j = \mathbf{A}_j \mathbf{x}$, for all j .

where $\rho = \sqrt{\frac{2(\delta_{3s})^2}{1 - (\delta_{2s})^2}} \|G\|_{2 \rightarrow 2} \leq 1$. Here, $\delta_{\kappa s} = \max_j \{ \delta_{\kappa s}^j \}$ with $\kappa = 2, 3$ and $\|G\|_{2 \rightarrow 2}$ is the spectral norm of matrix G .

Proof of theorem 3: see Appendix C.

Note that, the convergence is achieved when $\frac{2(\delta_{3s})^2}{1 - (\delta_{2s})^2} \|G\|_{2 \rightarrow 2}^2 \leq 1$ and is guaranteed if $(\delta_{3s})^2 \leq \frac{1}{1 + 2\|G\|_{2 \rightarrow 2}^2}$. The combination coefficients $c_{j,\ell}$ can be set such that the spectral norm $\|G\|_{2 \rightarrow 2}$ to be unit. So, replacing X^n , $X^\#$ and X^0 by \mathbf{x}_j^n , \mathbf{x} and \mathbf{x}_j^0 , respectively, the same non-cooperative result is achieved. On the other hand, regarding the relation $\|X^n - X^\#\|_2 \leq \sum_{j=1}^J \|\mathbf{x}_j^n - \mathbf{x}\|_2$, it is clear that the convergence rate of this diffusion method is better than that of non-cooperative case.

3.2.3. Convergence analysis of DDHTP for approximately sparse vectors measured with some errors

In addition to quantities that are introduced in (11), we define the following quantity:

$$E = \text{col} \{ \mathbf{e}_1, \dots, \mathbf{e}_J \} \quad (13)$$

so, the equation (6) is rewritten as

$$Y = \mathcal{A}X^\# + E \quad (14)$$

Theorem 4: For any possible inexact sparse signal $\mathbf{x} \in \mathbb{C}^N$ with the noisy measurements $\mathbf{y}_j = \mathbf{A}_j \mathbf{x} + \mathbf{e}_j$, the DDHTP algorithm generates sequence \mathbf{x}_j^n which satisfies the following relation

$$\left\| X^n - X_{\mathcal{S}}^\# \right\|_2 \leq \sqrt{\frac{2(\delta_{3s})^2}{1 - (\delta_{2s})^2}} \|G\|_{2 \rightarrow 2} \left\| X^{n-1} - X_{\mathcal{S}}^\# \right\|_2 + \left(\sqrt{\frac{2}{1 - \delta_{2s}}} + \frac{\sqrt{1 + \delta_s}}{1 - \delta_{2s}} \right) \left\| \hat{E} \right\|_2$$

where \mathcal{S} denotes the index set of J s largest entries of $X^\#$, $\hat{E} = \mathcal{A}X_{\mathcal{S}}^\# + E$, and $\delta_{\kappa s} = \max_j \{ \delta_{\kappa s}^j \}$ with $\kappa = 1, 2, 3$ and $\|G\|_{2 \rightarrow 2}$ denotes the spectral norm of the matrix G .

Proof of Theorem 4: see Appendix D.

Similar to the theorem 2, it is clear that the reconstruction error is controlled by the desired signal distance to a s -sparse vector and by the measurement error. In other words, again the *robustness* and *stability* of the algorithm are stored in this mode of cooperation.

4. Simulation Results

In this section, the proposed methods are evaluated using MATLAB software and the results are obtained through Monte-Carlo simulations over 100 trials. We focus on the signal recovery performance and convergence rates of the proposed algorithms. The proposed approaches are compared with the D-LASSO [33], DJ-IST [36], and a non-cooperative mode. For presenting the algorithms of [33] and [36], we set the simulation parameters according to the specifications provided by the authors to achieve the best results. A network with $J = 20$ sensors is considered in both incremental and diffusion topologies. For the diffusion type of cooperation, we consider a topology as shown in Fig. 1.

A different realization of the sensing matrices \mathbf{A}_i , the solution signal \mathbf{x} , and the noise vectors \mathbf{e}_i is used. In each trial, we construct the sensing

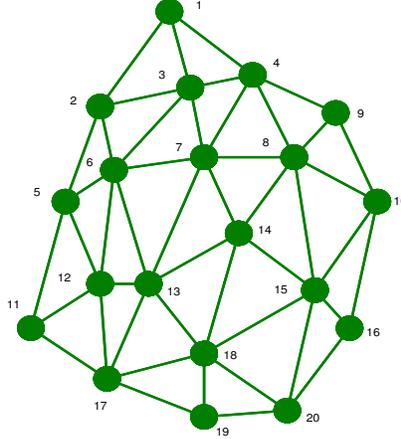


Figure 1: Network Topology

matrices \mathbf{A}_i with independent Gaussian distributed entries $\mathcal{N}(0, \frac{1}{m})$. Moreover, we generate a s -sparse vector \mathbf{x} whose support is generated from a uniform random distribution. The Gaussian sparse signal is considered for \mathbf{x} , i.e. the non-zero components of \mathbf{x} are independently produced from a standard Gaussian distribution. The total average normalized mean-squared error (TAN-MSE) between the original and estimated signals is used as a performance measure which is defined as:

$$TAN - MSE = E \left\{ \frac{\|\mathbf{X}^\# - \mathbf{X}^n\|_F^2}{\|\mathbf{X}^\#\|_F^2} \right\} \quad (15)$$

where $\|\mathbf{X}\|_F$ is the Frobenius norm of matrix \mathbf{X} defined as

$$\|\mathbf{X}\|_F = \sqrt{\text{tr}(\mathbf{X}\mathbf{X}^*)} = \sqrt{\text{tr}(\mathbf{X}^*\mathbf{X})} \quad (16)$$

where $\text{tr}(\mathbf{B})$ is the trace (the sum of diagonal elements) of matrix \mathbf{B} .

The parameters used in the simulations are $N = 1000$, $m = 200$, and $s = 50$, unless otherwise stated. First, we consider the noise-free scenario and exact sparse case. Fig. 2 shows TAN-MSE versus iterations for DDHTP, DIHTP, D-LASSO, and DJ-IST. The non-cooperative case is also provided in this figure for comparison. As shown in Fig. 2, for DDHTP, DIHTP, D-LASSO, and DJ-IST, the total numbers of iterations for convergence are respectively 7, 2, 62, and 7077. The steady-state errors are -293 dB, -293 dB, -33 dB, and -56.35 dB, respectively. So, one can easily observe that

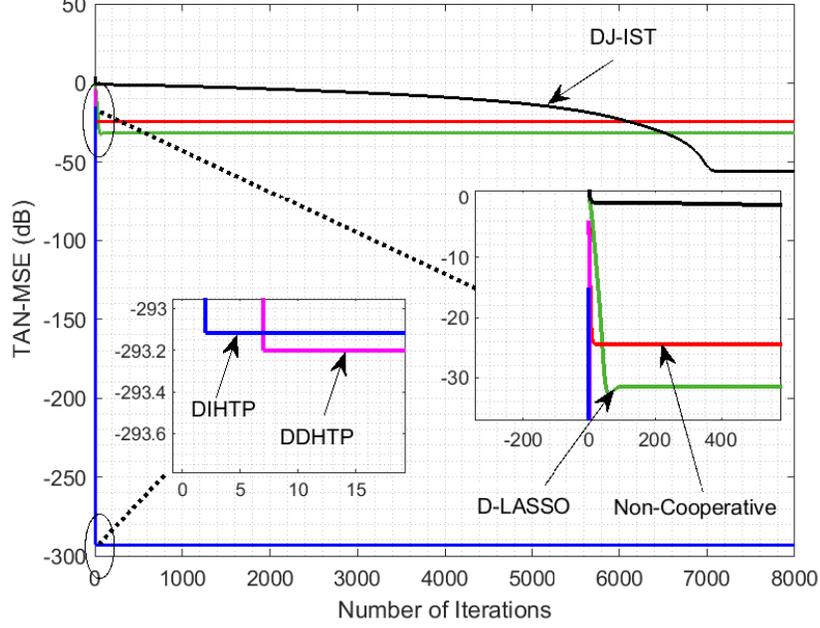


Figure 2: TAN-MSE versus iterations for different algorithms in noise-free scenario

the proposed DDHTP and DIHTP algorithms not only have a lower steady-state error but they also have a fast convergence rate that these issues verify the theoretical analysis. For the non-cooperative case, the total number of iterations for convergence and the steady-state error are respectively 18 and -24.33 dB. The results show the effectiveness of the proposed methods compared with non-cooperative counterparts.

Fig. 3 shows the TAN-MSE versus iterations for DDHTP, DIHTP, and DJ-IST in the noisy case for SNR=30 dB. For each sensor node in each trial, the noise is assumed to be a white Gaussian $\mathcal{N}(0, \sigma^2 \mathbf{I})$ where σ^2 is determined according to the desired SNR as $\sigma^2 = \frac{s}{m} \times 10^{\left(-\frac{SNR}{10}\right)}$. As displayed in Fig. 3, for DDHTP, DIHTP, and DJ-IST the total number of iterations for convergence are respectively 7, 2, and 7796. The steady-state errors are obtained as -33.21 dB, -33.31 dB, and -30.4 dB, respectively. By comparing the obtained results in figures 2 and 3, it is easily seen that the steady-state error of the proposed methods is strongly sensitive to the noise level. On the contrary, the convergence rate, according to the theoretical analysis, is not sensitive to the noise level. For DJ-IST, both the convergence rate and

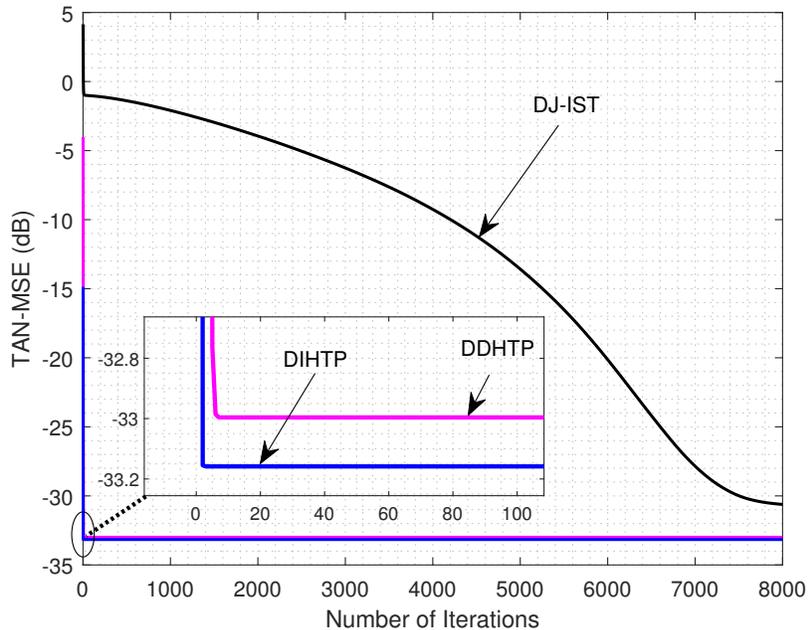


Figure 3: TAN-MSE versus iterations for DDHTP, DIHTP and DJ-IST (SNR=30dB)

steady-state error are sensitive to the noise level. The obtained simulation results showed that the non-cooperative modes and D-LASSO are not significantly sensitive to the noise level, and hence the simulations of these methods are ignored in this figure.

The steady-state TAN-MSE performance of each recovery method as a function of the SNR is presented in Fig. 4. As depicted in Fig. 4, the steady-state performance of the proposed methods is sensitive to the noise level and can be better for the higher SNRs. But, in the case of non-cooperative and D-LASSO methods, almost after 28 number of iterations, the increasing of the SNR can not improve the steady-state performance. Fig. 4 also shows that the performances of non-cooperative and D-LASSO methods are very close to each other.

We also compare the proposed methods with a centralized scenario where each node in the network sends its data to the FC using multi-hop relays. The simulations show that the results of the comparisons depend on the corresponding values of s , m , N , and J . For the given settings in figures 2-4, the TAN-MSE of the centralized strategy is almost the same as the result of the

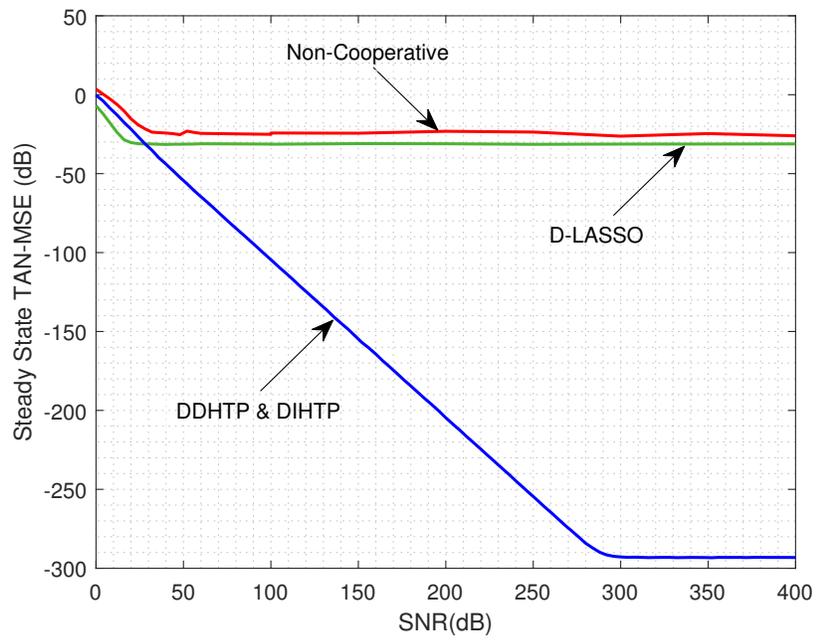


Figure 4: Steady-state TAN-MSE performance of recovery algorithms as a function of SNR

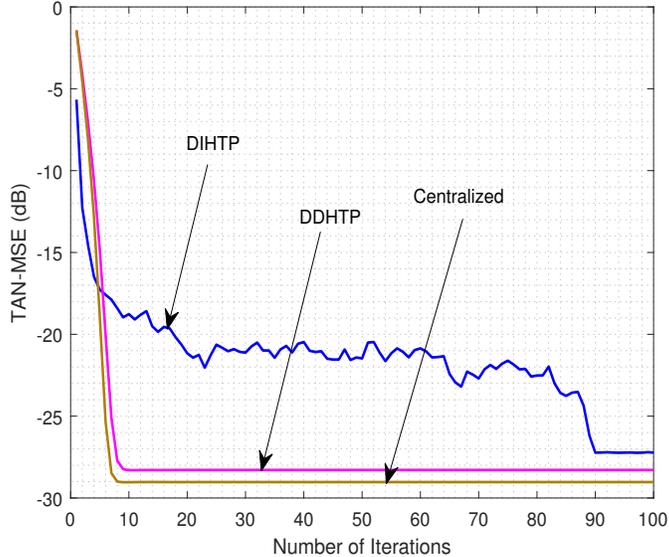


Figure 5: TAN-MSE versus iterations for DDHTP, DIHTP and centralized algorithms (SNR=30dB, $s=105$)

DDHTP algorithm. By increasing the value of s (or accordingly by decreasing the value of m) for the given N and J , the performances of the proposed and centralized schemes are different. To show this issue, we consider the network topology of Fig. 1 with an FC in the neighborhood of sensor 20. The obtained results of TAN-MSE in SNR=30dB for $s=105$ and 150 are respectively shown in figures 5 and 6. As shown in Fig. 5, the convergence rate and steady-state of the centralized method are close to the DDHTP but significantly different from the DIHTP. This is due to access to more data resources in both DDHTP and centralized scenarios compared with DIHTP. By increasing s to 150, as shown in Fig. 6, the difference between DDHTP and centralized becomes even greater. In this case, the DIHTP algorithm diverges. For both Figs 5 and 6 settings, the non-cooperative method also diverges. It is easily seen for the signals with small sparsity rate the proposed algorithms can perform close to the centralized method while they save a part of the required energy for data communications.

The next measure for evaluation of the proposed algorithm is the probability of successful reconstruction. We record a successful recovery when $TAN - MSE \leq 10^{-5}$. For each algorithm, the probability of successful

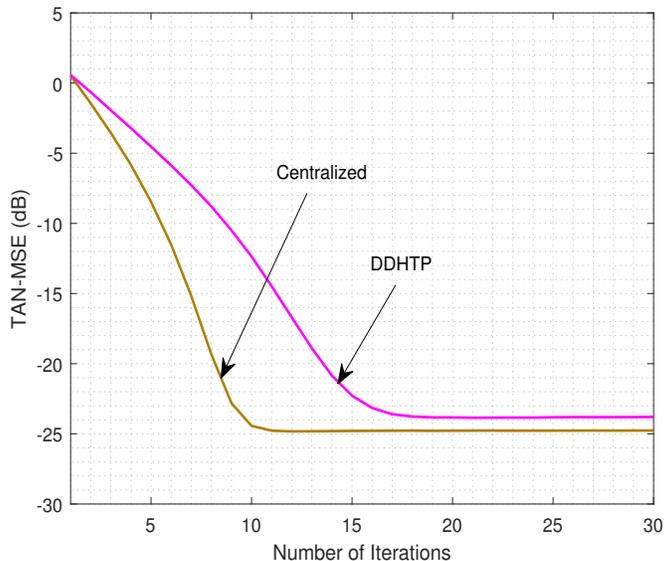


Figure 6: TAN-MSE versus iterations for DDHTP and centralized algorithms (SNR=30dB, $s=150$)

recovery as a function of the sparsity level (the number of non-zero entries of the signal) is shown in Fig. 7. A very surprising result is that, in the distributed and cooperative case, the numbers of required measurements are less than that of the non-cooperative mode (common CS). Besides, in the diffusion scheme, the numbers of required measurements are less than the incremental mode of cooperation. This result does not depend on the type of the algorithm since in the case of D-LASSO we also observed a similar result. Rather it arises from the spatial correlation between sensor nodes. More clearly, the correlation between nodes results in redundancy in the number of measurements.

Although for the centralized scenario, the numbers of required measurements are less than all methods, it results in a significant cost of energy consumption. In other words, the non-cooperative method provides a lower bound and the centralized method provides an upper bound on the performance of the proposed methods. More clearly, in the non-cooperative scenario, no energy is consumed to communicate between nodes. On the contrary, in the centralized method, communication power consumption is significantly higher than the proposed methods. As one can see that there is

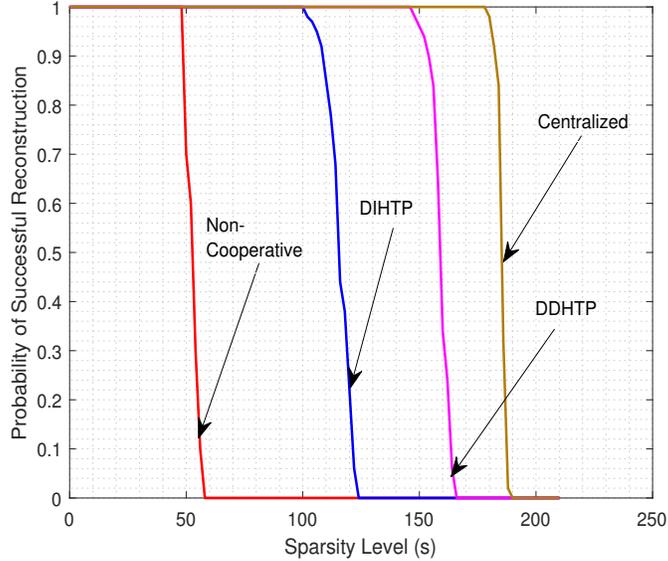


Figure 7: Probability of successful recoveries as a function of sparsity level

a tradeoff between performance and power consumption for these methods.

5. Conclusions

In this paper, we proposed a compressive sensing (CS) recovery algorithm in incremental and diffusion schemes of cooperation in wireless sensor networks (WSNs). We analyzed the performance recovery of the proposed distributed incremental and diffusion hard thresholding pursuit (HTP) algorithms in the noise-free scenario and the case that vectors are not exactly sparse and they can not be measured with perfect precision (i.e. the noisy case). Such a study is challenging since nodes with one-hop distance from each other interact and therefore a successful analysis should take into account the temporal and spatial interconnectedness of the data. These make the analysis more complicated. In our analysis, we derived a closed-form expression for the mean squared deviation between the original and recovered coefficients to evaluate the convergence performance of each sensor node. The results show that, in conflict with the non-cooperative and other methods, the convergence rate of the proposed algorithms is very high. Furthermore, the sensitivity of the suggested methods for the noise of measurements and

inexact sparse cases is very high.

Appendix A. Proof of Theorem 1

The proof of theorem 1 is presented by the following lemma :

Lemma 1 (see e.g. [26]): Given the vectors $\mathbf{u}, \mathbf{v} \in \mathbb{C}^N$, it is obtained that $|\langle \mathbf{u}, (\mathbf{I} - \mathbf{A}^* \mathbf{A}) \mathbf{v} \rangle| \leq \delta_t \|\mathbf{u}\|_2 \|\mathbf{v}\|_2$ if $\text{card}(\text{supp}(\mathbf{u}) \cup \text{supp}(\mathbf{v})) \leq t$ and we have $\|((\mathbf{I} - \mathbf{A}^* \mathbf{A}) \mathbf{v})_S\|_2 \leq \delta_t \|\mathbf{v}\|_2$ if $\text{card}(S \cup \text{supp}(\mathbf{v})) \leq t$.

Proof of Theorem 1: Since the vector $\mathbf{A}_j \mathbf{x}_j^n$ is the projection of \mathbf{y}_j on the space $\{\mathbf{A}_j \mathbf{z}, \text{supp}(\mathbf{z}) \subset S_j^n\}$, it is expected that for all \mathbf{z} with $\text{supp}(\mathbf{z}) \subset S_j^n$ we have $\langle \mathbf{A}_j \mathbf{x}_j^n - \mathbf{y}_j, \mathbf{A}_j \mathbf{z} \rangle = 0$. In the other words, using the equality $\mathbf{y}_j = \mathbf{A}_j \mathbf{x}$, we have $\langle \mathbf{x}_j^n - \mathbf{x}, \mathbf{A}_j^* \mathbf{A}_j \mathbf{z} \rangle = 0$ when $\text{supp}(\mathbf{z}) \subset S_j^n$. The vector in \mathbb{C}^N that coincides with \mathbf{z} on the entries in S and zero on the entries outside S is denoted by \mathbf{z}_S . Then, we have

$$\begin{aligned} \left\| (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\|_2^2 &= \left\langle \mathbf{x}_j^n - \mathbf{x}, (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\rangle \\ &= \left\langle \mathbf{x}_j^n - \mathbf{x}, (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\rangle - \left\langle \mathbf{x}_j^n - \mathbf{x}, \mathbf{A}_j^* \mathbf{A}_j (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\rangle \\ &= \left\langle \mathbf{x}_j^n - \mathbf{x}, (\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j) (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\rangle \end{aligned} \quad (\text{A.1})$$

using Lemma 1, the following equation is achieved:

$$\left\| (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\|_2^2 \leq \delta_{2s}^j \|\mathbf{x}_j^n - \mathbf{x}\|_2 \left\| (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\|_2 \quad (\text{A.2})$$

dividing the both sides to $\left\| (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\|_2$, we have

$$\left\| (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\|_2 \leq \delta_{2s}^j \|\mathbf{x}_j^n - \mathbf{x}\|_2 \quad (\text{A.3})$$

definition of the vector norm, one can easily get that

$$\|\mathbf{x}_j^n - \mathbf{x}\|_2^2 = \left\| (\mathbf{x}_j^n - \mathbf{x})_{\overline{S_j^n}} \right\|_2^2 + \left\| (\mathbf{x}_j^n - \mathbf{x})_{S_j^n} \right\|_2^2 \quad (\text{A.4})$$

where $\overline{S_j^n}$ is the complement of S_j^n . Substituting (A.3) in (A.4):

$$\|\mathbf{x}_j^n - \mathbf{x}\|_2^2 \leq \left\| (\mathbf{x}_j^n - \mathbf{x})_{\overline{S_j^n}} \right\|_2^2 + (\delta_{2s}^j)^2 \|\mathbf{x}_j^n - \mathbf{x}\|_2^2 \quad (\text{A.5})$$

the above-equation can be rewritten as

$$\|\mathbf{x}_j^n - \mathbf{x}\|_2^2 \leq \frac{1}{1 - (\delta_{2s}^j)^2} \left\| (\mathbf{x}_j^n - \mathbf{x})_{\overline{S_j^n}} \right\|_2^2 \quad (\text{A.6})$$

assuming $S = \text{supp}(\mathbf{x})$, we have:

$$\begin{aligned} \left\| (\mathbf{x}_{j-1}^n + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n))_S \right\|_2^2 &\leq \\ \left\| (\mathbf{x}_{j-1}^n + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n))_{S_j^n} \right\|_2^2 & \end{aligned} \quad (\text{A.7})$$

removing the contribution related to $S_j^n \cap S$ from both sides of (A.7), the following equation is obtained

$$\begin{aligned} \left\| (\mathbf{x}_{j-1}^n + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n))_{S \setminus S_j^n} \right\|_2 &\leq \\ \left\| (\mathbf{x}_{j-1}^n + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n))_{S_j^n \setminus S} \right\|_2 & \end{aligned} \quad (\text{A.8})$$

the right hand side of (A.8) can be arranged as

$$\begin{aligned} &\left\| (\mathbf{x}_{j-1}^n + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n))_{S_j^n \setminus S} \right\|_2 \\ &= \left\| (\mathbf{x}_{j-1}^n - \mathbf{x} + \mathbf{A}_j^* \mathbf{A}_j (\mathbf{x} - \mathbf{x}_{j-1}^n))_{S_j^n \setminus S} \right\|_2 \\ &= \left\| ((\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j) (\mathbf{x}_{j-1}^n - \mathbf{x}))_{S_j^n \setminus S} \right\|_2 \end{aligned} \quad (\text{A.9})$$

similarly, for the left hand side of (A.8), we have

$$\begin{aligned} &\left\| (\mathbf{x}_{j-1}^n + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n))_{S \setminus S_j^n} \right\|_2 \\ &= \left\| \left(\mathbf{x} - \mathbf{x} - (\mathbf{x}_j^n)_{\overline{S_j^n}} + \mathbf{x}_{j-1}^n + \mathbf{A}_j^* \mathbf{A}_j (\mathbf{x} - \mathbf{x}_{j-1}^n) \right)_{S \setminus S_j^n} \right\|_2 \\ &= \left\| (\mathbf{x} - \mathbf{x}_j^n)_{\overline{S_j^n}} + ((\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j) (\mathbf{x}_{j-1}^n - \mathbf{x}))_{S \setminus S_j^n} \right\|_2 \end{aligned} \quad (\text{A.10})$$

using the triangle inequality, it is agreed that

$$\begin{aligned} &\left\| (\mathbf{x}_{j-1}^n + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n))_{S \setminus S_j^n} \right\|_2 \geq \\ &\left\| (\mathbf{x} - \mathbf{x}_j^n)_{\overline{S_j^n}} \right\|_2 - \left\| ((\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j) (\mathbf{x}_{j-1}^n - \mathbf{x}))_{S \setminus S_j^n} \right\|_2 \end{aligned} \quad (\text{A.11})$$

substituting (A.9) and (A.11) into (A.8), we get that

$$\begin{aligned} & \left\| (\mathbf{x} - \mathbf{x}_j^n)_{\overline{S_j^n}} \right\|_2 \leq \left\| ((\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j) (\mathbf{x}_{j-1}^n - \mathbf{x}))_{S \setminus S_j^n} \right\|_2 \\ & + \left\| ((\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j) (\mathbf{x}_{j-1}^n - \mathbf{x}))_{S_j^n \setminus S} \right\|_2 \leq \\ & \sqrt{2} \left\| ((\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j) (\mathbf{x}_{j-1}^n - \mathbf{x}))_{S_j^n \Delta S} \right\|_2 \end{aligned} \quad (\text{A.12})$$

exploiting Lemma 1, the equation (A.12) is summarized as

$$\left\| (\mathbf{x} - \mathbf{x}_j^n)_{\overline{S_j^n}} \right\|_2 \leq \sqrt{2} \delta_{3s}^j \|\mathbf{x}_{j-1}^n - \mathbf{x}\|_2 \quad (\text{A.13})$$

substituting (A.6) in (A.13), yields the following equation:

$$\|\mathbf{x}_j^n - \mathbf{x}\|_2 \leq \sqrt{\frac{2(\delta_{3s}^j)^2}{1-(\delta_{2s}^j)^2}} \|\mathbf{x}_{j-1}^n - \mathbf{x}\|_2 \quad (\text{A.14})$$

replacing the term \mathbf{x}_j^{n-1} instead of \mathbf{x}_{j-1}^n , in the right-hand side of (A.14), it gives the same result for the single signal case [26]. But, in each iteration, the error of node j , $\mathbf{x}_j^n - \mathbf{x}$, depends on the previous node's error. To write the error of node j in terms of its previous iteration error, we exploit the incremental topology structure. So, we have

$$\begin{aligned} \|\mathbf{x}_j^n - \mathbf{x}\|_2 & \leq \sqrt{\frac{2(\delta_{3s}^j)^2}{1-(\delta_{2s}^j)^2}} \sqrt{\frac{2(\delta_{3s}^{j-1})^2}{1-(\delta_{2s}^{j-1})^2}} \|\mathbf{x}_{j-2}^n - \mathbf{x}\|_2 \leq \\ & \prod_{k=2}^j \sqrt{\frac{2(\delta_{3s}^k)^2}{1-(\delta_{2s}^k)^2}} \|\mathbf{x}_1^n - \mathbf{x}\|_2 \end{aligned} \quad (\text{A.15})$$

continuing this procedure, one can easily verify that

$$\|\mathbf{x}_j^n - \mathbf{x}\|_2 \leq \left(\prod_{k=2}^j \sqrt{\frac{2(\delta_{3s}^k)^2}{1-(\delta_{2s}^k)^2}} \right) \left(\prod_{k=j+1}^J \sqrt{\frac{2(\delta_{3s}^k)^2}{1-(\delta_{2s}^k)^2}} \right) \sqrt{\frac{2(\delta_{3s}^1)^2}{1-(\delta_{2s}^1)^2}} \|\mathbf{x}_j^{n-1} - \mathbf{x}\|_2 \quad (\text{A.16})$$

which is summarized as the following equation

$$\|\mathbf{x}_j^n - \mathbf{x}\|_2 \leq \prod_{j=1}^J \sqrt{\frac{2(\delta_{3s}^j)^2}{1-(\delta_{2s}^j)^2}} \|\mathbf{x}_j^{n-1} - \mathbf{x}\|_2 \quad (\text{A.17})$$

Appendix B. Proof of Theorem 2

To prove the theorem 2, we use the following lemma:
 Lemma 2 (see [26]): For the given $\mathbf{e} \in \mathbb{C}^N$ and $S \subset \{1, 2, \dots, N\}$ with $\text{Card}(S) \leq s$, we have

$$\|(\mathbf{A}^* \mathbf{e})_S\|_2 \leq \sqrt{1 + \delta_s} \|\mathbf{e}\|_2$$

Proof of Theorem 2: Since the vector $\mathbf{A}_j \mathbf{x}_j^n$ is the projection of \mathbf{y}_j on to the space $\{\mathbf{A}_j \mathbf{z}, \text{supp}(\mathbf{z}) \subset S_j^n\}$, it is expected that for all \mathbf{z} with $\text{supp}(\mathbf{z}) \subset S_j^n$ we have $\langle \mathbf{A}_j \mathbf{x}_j^n - \mathbf{y}_j, \mathbf{A}_j \mathbf{z} \rangle = 0$ or equivalently $(\mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_j^n))_{S_j^n} = 0$. So, using equation (A.12), we will have

$$\begin{aligned} \|\mathbf{x}_j^n - \mathbf{x}_S\|_2^2 &= \left\| (\mathbf{x}_j^n - \mathbf{x}_S)_{S_j^n} \right\|_2^2 + \left\| (\mathbf{x}_j^n - \mathbf{x}_S)_{\overline{S_j^n}} \right\|_2^2 \leq \\ &\left\| (\mathbf{x}_j^n - \mathbf{x}_S + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_j^n))_{S_j^n} \right\|_2^2 \\ &+ 2 \left\| (\mathbf{x}_{j-1}^n - \mathbf{x}_S + \mathbf{A}_j^* (\mathbf{y}_j - \mathbf{A}_j \mathbf{x}_{j-1}^n))_{S_j^n \Delta S} \right\|_2^2 \end{aligned} \quad (\text{B.1})$$

using the fact that $\mathbf{y}_j = \mathbf{A}_j \mathbf{x} + \mathbf{e}_j = \mathbf{A}_j \mathbf{x}_S + \mathbf{A}_j \mathbf{x}_{\overline{S}} + \mathbf{e}_j = \mathbf{A}_j \mathbf{x}_S + \widehat{\mathbf{e}}_j$ where $\widehat{\mathbf{e}}_j = \mathbf{A}_j \mathbf{x}_{\overline{S}} + \mathbf{e}_j$, we have

$$\begin{aligned} \|\mathbf{x}_j^n - \mathbf{x}_S\|_2^2 &\leq \left\| (\mathbf{x}_j^n - \mathbf{x}_S + \mathbf{A}_j^* \mathbf{A}_j (\mathbf{x}_S - \mathbf{x}_j^n) + \mathbf{A}_j^* \widehat{\mathbf{e}}_j)_{S_j^n} \right\|_2^2 \\ &+ 2 \left\| (\mathbf{x}_{j-1}^n - \mathbf{x}_S + \mathbf{A}_j^* \mathbf{A}_j (\mathbf{x}_S - \mathbf{x}_{j-1}^n) + \mathbf{A}_j^* \widehat{\mathbf{e}}_j)_{S_j^n \Delta S} \right\|_2^2 \leq \\ &\left(\left\| ((\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j) (\mathbf{x}_j^n - \mathbf{x}_S))_{S_j^n} \right\|_2 + \left\| (\mathbf{A}_j^* \widehat{\mathbf{e}}_j)_{S_j^n} \right\|_2 \right)^2 \\ &+ 2 \left(\left\| ((\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j) (\mathbf{x}_{j-1}^n - \mathbf{x}_S))_{S_j^n \Delta S} \right\|_2 + \left\| (\mathbf{A}_j^* \widehat{\mathbf{e}}_j)_{S_j^n \Delta S} \right\|_2 \right)^2 \end{aligned} \quad (\text{B.2})$$

exploiting the Lemmas 1 and 2, we will get

$$\begin{aligned} \|\mathbf{x}_j^n - \mathbf{x}_S\|_2^2 &\leq \left(\delta_{2s}^j \|\mathbf{x}_j^n - \mathbf{x}_S\|_2 + \sqrt{1 + \delta_s^j} \|\widehat{\mathbf{e}}_j\|_2 \right)^2 \\ &+ 2 \left(\delta_{3s}^j \|\mathbf{x}_{j-1}^n - \mathbf{x}_S\|_2 + \sqrt{1 + \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right)^2 \end{aligned} \quad (\text{B.3})$$

factoring the difference of two squares, we have

$$\begin{aligned} & \left[\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 - \left(\delta_{2s}^j \|\mathbf{x}_j^n - \mathbf{x}_S\|_2 + \sqrt{1 + \delta_s^j} \|\widehat{\mathbf{e}}_j\|_2 \right) \right] \times \\ & \left[\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 + \left(\delta_{2s}^j \|\mathbf{x}_j^n - \mathbf{x}_S\|_2 + \sqrt{1 + \delta_s^j} \|\widehat{\mathbf{e}}_j\|_2 \right) \right] \leq \\ & 2 \left(\delta_{3s}^j \|\mathbf{x}_{j-1}^n - \mathbf{x}_S\|_2 + \sqrt{1 + \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right)^2 \end{aligned} \quad (\text{B.4})$$

by manipulation of (B.4), the following equation is achieved

$$\begin{aligned} & \left(1 - (\delta_{2s}^j)^2 \right) \left[\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 - \frac{\sqrt{1 + \delta_s^j}}{1 - \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right] \times \\ & \left[\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 + \frac{\sqrt{1 + \delta_s^j}}{1 + \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right] \leq \\ & 2 \left(\delta_{3s}^j \|\mathbf{x}_{j-1}^n - \mathbf{x}_S\|_2 + \sqrt{1 + \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right)^2 \end{aligned} \quad (\text{B.5})$$

considering $\left[\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 - \frac{\sqrt{1 + \delta_s^j}}{1 - \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right]^2 \leq$
 $\left[\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 - \frac{\sqrt{1 + \delta_s^j}}{1 - \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right] \left[\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 + \frac{\sqrt{1 + \delta_s^j}}{1 + \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right]$

the equation (B.5) is rewritten as

$$\begin{aligned} & \sqrt{1 - (\delta_{2s}^j)^2} \left[\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 - \frac{\sqrt{1 + \delta_s^j}}{1 - \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right] \leq \\ & \sqrt{2} \left(\delta_{3s}^j \|\mathbf{x}_{j-1}^n - \mathbf{x}_S\|_2 + \sqrt{1 + \delta_{2s}^j} \|\widehat{\mathbf{e}}_j\|_2 \right) \end{aligned} \quad (\text{B.6})$$

dividing the both sides of (B.6) to $\sqrt{1 - (\delta_{2s}^j)^2}$ and some manipulations, we get

$$\begin{aligned} & \|\mathbf{x}_j^n - \mathbf{x}_S\|_2 \leq \\ & \sqrt{\frac{2(\delta_{3s}^j)^2}{1 - (\delta_{2s}^j)^2}} \|\mathbf{x}_{j-1}^n - \mathbf{x}_S\|_2 + \left(\sqrt{\frac{2}{1 - \delta_{2s}^j}} + \frac{\sqrt{1 + \delta_s^j}}{1 - \delta_{2s}^j} \right) \|\widehat{\mathbf{e}}_j\|_2 \end{aligned} \quad (\text{B.7})$$

defining $\beta_j = \sqrt{\frac{2(\delta_{3s}^j)^2}{1 - (\delta_{2s}^j)^2}}$ and $\mathcal{F}_j = \left(\sqrt{\frac{2}{1 - \delta_{2s}^j}} + \frac{\sqrt{1 + \delta_s^j}}{1 - \delta_{2s}^j} \right) \|\mathbf{A}_j \mathbf{x}_{\bar{S}} + \mathbf{e}_j\|_2$, the equation (B.7) can be written as

$$\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 \leq \beta_j \|\mathbf{x}_{j-1}^n - \mathbf{x}_S\|_2 + \mathcal{F}_j \quad (\text{B.8})$$

It is observed that the (B.8) is a coupled equation. It involves $\|\mathbf{x}_j^n - \mathbf{x}_S\|_2$ and $\|\mathbf{x}_{j-1}^n - \mathbf{x}_S\|_2$, i.e., information from two spatial locations. To simplify the above equation, we use the advantage of the ring topology (i.e., the availability of the cyclic pattern for cooperation between the nodes) that is inherent in the incremental strategy. Thus, by iterating (B.8), we have

$$\begin{aligned} \|\mathbf{x}_j^n - \mathbf{x}_S\|_2 &\leq \beta_j \|\mathbf{x}_{j-1}^n - \mathbf{x}_S\|_2 + \mathcal{F}_j \\ \|\mathbf{x}_{j-1}^n - \mathbf{x}_S\|_2 &\leq \beta_{j-1} \|\mathbf{x}_{j-2}^n - \mathbf{x}_S\|_2 + \mathcal{F}_{j-1} \\ \|\mathbf{x}_{j-2}^n - \mathbf{x}_S\|_2 &\leq \beta_{j-2} \|\mathbf{x}_{j-3}^n - \mathbf{x}_S\|_2 + \mathcal{F}_{j-2} \end{aligned} \quad (\text{B.9})$$

according to (B.9), $\|\mathbf{x}_j^n - \mathbf{x}_S\|_2$ can be expressed in terms of $\|\mathbf{x}_{j-3}^n - \mathbf{x}_S\|_2$ as

$$\begin{aligned} \|\mathbf{x}_j^n - \mathbf{x}_S\|_2 &\leq \beta_j \beta_{j-1} \beta_{j-2} \|\mathbf{x}_{j-3}^n - \mathbf{x}_S\|_2 \\ &\quad + \beta_j \beta_{j-1} \mathcal{F}_{j-2} + \beta_j \mathcal{F}_{j-1} + \mathcal{F}_j \end{aligned} \quad (\text{B.10})$$

iterating in the same manner, the following equation is obtained

$$\begin{aligned} \|\mathbf{x}_j^n - \mathbf{x}_S\|_2 &\leq \prod_{k=1}^J \beta_k \|\mathbf{x}_j^{n-1} - \mathbf{x}_S\|_2 + \prod_{k=1, k \neq j+1}^J \beta_k \mathcal{F}_{j+1} \\ &\quad + \prod_{k=1, k \neq j+1, j+2}^J \beta_k \mathcal{F}_{j+2} + \cdots + \prod_{k=1}^J \beta_k \beta_J \mathcal{F}_{J-1} + \prod_{k=1}^j \beta_k \mathcal{F}_J \\ &\quad + \prod_{k=2}^j \beta_k \mathcal{F}_1 + \prod_{k=3}^j \beta_k \mathcal{F}_2 + \cdots + \beta_j \beta_{j-1} \mathcal{F}_{j-2} + \beta_j \mathcal{F}_{j-1} + \mathcal{F}_j \end{aligned} \quad (\text{B.11})$$

we define a set of J quantities for each node j as

$$\Pi_{j,\ell} \triangleq \beta_j \beta_{j-1} \cdots \beta_1 \beta_J \beta_{J-1} \cdots \beta_{j+\ell}, \quad \ell = 1, \dots, J \quad (\text{B.12})$$

$$\mathcal{H}_j = \Pi_{j,2} \mathcal{F}_{j+1} + \Pi_{j,3} \mathcal{F}_{j+2} + \cdots + \Pi_{j,J-1} \mathcal{F}_{j-2} + \Pi_{j,J} \mathcal{F}_{j-1} + \mathcal{F}_j \quad (\text{B.13})$$

from the equations (B.11)-(B.13), we have

$$\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 \leq \Pi_{j,1} \|\mathbf{x}_j^{n-1} - \mathbf{x}_S\|_2 + \mathcal{H}_j \quad (\text{B.14})$$

using the mathematical induction, we conclude that

$$\|\mathbf{x}_j^n - \mathbf{x}_S\|_2 \leq (\Pi_{j,1})^n \|\mathbf{x}_j^0 - \mathbf{x}_S\|_2 + (\Pi_{j,1})^{n-1} \mathcal{H}_j \quad (\text{B.15})$$

Appendix C. Proof of Theorem 3

Similar to proof of theorem 1, we conclude that $\langle X^n - X^\#, \mathcal{A}^* \mathcal{A} Z \rangle = 0$ when $\text{supp}(Z) \subset \mathcal{S}^n$. So, we have

$$\|(X^n - X^\#)_{\mathcal{S}^n}\|_2^2 = \langle X^n - X^\#, (\mathbf{I} - \mathcal{A}^* \mathcal{A}) (X^n - X^\#)_{\mathcal{S}^n} \rangle \quad (\text{C.1})$$

using the fact that $\|\mathbf{I} - \mathcal{A}^* \mathcal{A}\|_{2 \rightarrow 2} = \max_j \left\{ \|\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j\|_{2 \rightarrow 2} \right\}$ and exploiting Lemma 1, we conclude

$$\|(X^n - X^\#)_{\mathcal{S}^n}\|_2 \leq \max_j \{ \delta_{2s}^j \} \|X^n - X^\#\|_2 \quad (\text{C.2})$$

and

$$\|X^n - X^\#\|_2^2 = \|(X^n - X^\#)_{\overline{\mathcal{S}^n}}\|_2^2 + \|(X^n - X^\#)_{\mathcal{S}^n}\|_2^2 \quad (\text{C.3})$$

by substituting (C.2) into (C.3), we get

$$\|X^n - X^\#\|_2^2 \leq \frac{1}{1 - (\delta_{2s})^2} \|(X^n - X^\#)_{\overline{\mathcal{S}^n}}\|_2^2 \quad (\text{C.4})$$

where $\delta_{2s} = \max_j \{ \delta_{2s}^j \}$. We rewrite the first two terms of (10) in a compact form such as

$$\mathcal{S}^n = \mathcal{L}_s^J (GX^{n-1} + \mathcal{A}^* (Y - \mathcal{A}GX^{n-1})) \quad (\text{C.5})$$

assuming $\mathcal{S} = \text{supp}(X)$, we will have

$$\left\| \begin{array}{l} (GX^{n-1} + \mathcal{A}^* (Y - \mathcal{A}GX^{n-1}))_{\mathcal{S} \setminus \mathcal{S}^n} \\ (GX^{n-1} + \mathcal{A}^* (Y - \mathcal{A}GX^{n-1}))_{\mathcal{S}^n \setminus \mathcal{S}} \end{array} \right\|_2 \leq \quad (\text{C.6})$$

considering the right hand side of (C.6), we get

$$\begin{aligned} & \left\| (GX^{n-1} + \mathcal{A}^* (Y - \mathcal{A}GX^{n-1}))_{\mathcal{S}^n \setminus \mathcal{S}} \right\|_2 \\ &= \left\| (GX^{n-1} - X^\# + \mathcal{A}^* \mathcal{A} (X - GX^{n-1}))_{\mathcal{S}^n \setminus \mathcal{S}} \right\|_2 \\ &= \left\| ((\mathbf{I} - \mathcal{A}^* \mathcal{A}) (GX^{n-1} - X^\#))_{\mathcal{S}^n \setminus \mathcal{S}} \right\|_2 \end{aligned} \quad (\text{C.7})$$

for the left hand side of (C.6), we have

$$\begin{aligned}
& \left\| (GX^{n-1} + \mathcal{A}^*(Y - \mathcal{A}GX^{n-1}))_{\mathcal{S} \setminus \mathcal{S}^n} \right\|_2 \\
&= \left\| (X^\# - X^n - (X^n)_{\overline{\mathcal{S}^n}} + GX^{n-1} + \mathcal{A}^* \mathcal{A} (X^\# - GX^{n-1}))_{\mathcal{S} \setminus \mathcal{S}^n} \right\|_2 \geq \\
& \left\| (X^\# - X^n)_{\overline{\mathcal{S}^n}} \right\|_2 - \left\| ((\mathbf{I} - \mathcal{A}^* \mathcal{A}) (GX^{n-1} - X^\#))_{\mathcal{S} \setminus \mathcal{S}^n} \right\|_2
\end{aligned} \tag{C.8}$$

with regard to (C.6)-(C.8), we get

$$\begin{aligned}
& \left\| (X^\# - X^n)_{\overline{\mathcal{S}^n}} \right\|_2 \leq \left\| ((\mathbf{I} - \mathcal{A}^* \mathcal{A}) (GX^{n-1} - X^\#))_{\mathcal{S} \setminus \mathcal{S}^n} \right\|_2 \\
& + \left\| ((\mathbf{I} - \mathcal{A}^* \mathcal{A}) (GX^{n-1} - X^\#))_{\mathcal{S}^n \setminus \mathcal{S}} \right\|_2 \leq \\
& \sqrt{2} \left\| ((\mathbf{I} - \mathcal{A}^* \mathcal{A}) (GX^{n-1} - X^\#))_{\mathcal{S} \Delta \mathcal{S}^n} \right\|_2
\end{aligned} \tag{C.9}$$

considering $\|\mathbf{I} - \mathcal{A}^* \mathcal{A}\|_{2 \rightarrow 2} = \max_j \left\{ \|\mathbf{I} - \mathbf{A}_j^* \mathbf{A}_j\|_{2 \rightarrow 2} \right\}$ and Lemma 1, we take

$$\left\| (X^\# - X^n)_{\overline{\mathcal{S}^n}} \right\|_2 \leq \sqrt{2} \max_j \{ \delta_{3s}^j \} \|GX^{n-1} - X^\#\|_2 \tag{C.10}$$

By denoting $\delta_{3s} = \max_j \{ \delta_{3s}^j \}$ and substituting (C.10) in (C.4), we get

$$\|X^n - X^\#\|_2 \leq \sqrt{\frac{2(\delta_{3s})^2}{1 - (\delta_{2s})^2}} \|GX^{n-1} - X^\#\|_2 \tag{C.11}$$

finally, rewriting (C.11) as

$$\|X^n - X^\#\|_2 \leq \sqrt{\frac{2(\delta_{3s})^2}{1 - (\delta_{2s})^2}} \|G\|_{2 \rightarrow 2} \|X^{n-1} - X^\#\|_2 \tag{C.12}$$

completes the proof.

It should be noted that in (C.12), we used the fact that $GX^\# = X^\#$ due to $\sum_{\ell \in \mathcal{N}_j} c_{j,\ell} = 1$.

Appendix D. Proof of Theorem 4

According to the proof of theorem 2, we conclude $(\mathcal{A}^*(Y - \mathcal{A}X^n))_{\mathcal{S}^n} = 0$ when $\text{supp}(Z) \subset \mathcal{S}^n$. So, using the equation (C.9), we have

$$\begin{aligned}
\|X^n - X_S^\#\|_2^2 &= \|(X^n - X_S^\#)_{S^n}\|_2^2 + \|(X^n - X_S^\#)_{S \setminus S^n}\|_2^2 \leq \\
&\|(X^n - X_S^\# + \mathcal{A}^*(Y - \mathcal{A}X^n))_{S^n}\|_2^2 + \\
&2\|(GX^{n-1} - X_S^\# + \mathcal{A}^*(Y - \mathcal{A}GX^{n-1}))_{S\Delta S^n}\|_2^2
\end{aligned} \tag{D.1}$$

regarding $Y = \mathcal{A}X^\# + E = \mathcal{A}X_S^\# + \mathcal{A}X_{S^c}^\# + E = \mathcal{A}X_S^\# + \hat{E}$ where $\hat{E} = \mathcal{A}X_{S^c}^\# + E$, we get

$$\begin{aligned}
\|X^n - X_S^\#\|_2^2 &\leq \|(X^n - X_S^\# + \mathcal{A}^*\mathcal{A}(X_S^\# - X^n) + \mathcal{A}^*\hat{E})_{S^n}\|_2^2 + \\
&2\|(GX^{n-1} - X_S^\# + \mathcal{A}^*\mathcal{A}(X_S^\# - GX^{n-1}) + \mathcal{A}^*\hat{E})_{S\Delta S^n}\|_2^2 \leq \\
&\left(\|(I - \mathcal{A}^*\mathcal{A})(X^n - X_S^\#)\|_{S^n} + \|\mathcal{A}^*\hat{E}\|_{S^n}\right)^2 + \\
&2\left(\|(I - \mathcal{A}^*\mathcal{A})(GX^{n-1} - X_S^\#)\|_{S\Delta S^n} + \|\mathcal{A}^*\hat{E}\|_{S\Delta S^n}\right)^2
\end{aligned} \tag{D.2}$$

considering two Lemmas 1 and 2 and using the fact that $\|I - \mathcal{A}^*\mathcal{A}\|_{2 \rightarrow 2} = \max_j \{\|I - \mathbf{A}_j^*\mathbf{A}_j\|_{2 \rightarrow 2}\}$, the following equation is achieved

$$\begin{aligned}
\|X^n - X_S^\#\|_2^2 &\leq \left(\max_j \{\delta_{2s}^j\} \|X^n - X_S^\#\|_2 + \max_j \sqrt{1 + \delta_s^j} \|\hat{E}\|_2\right)^2 \\
&+ 2\left(\max_j \{\delta_{3s}^j\} \|GX^{n-1} - X_S^\#\|_2 + \max_j \sqrt{1 + \delta_{2s}^j} \|\hat{E}\|_2\right)^2
\end{aligned} \tag{D.3}$$

Note that, in order to apply Lemma 2, for obtaining (D.3) from (D.2), we modify it as $\|(\mathcal{A}^*E)_{S^n}\|_2^2 = \sum_{j=1}^J \|(\mathbf{A}_j^*e_j)_{S_j^n}\|_2^2 \leq \sum_{j=1}^J (1 + \delta_s^j) \|e_j\|_2^2 \leq \sum_{k=1}^J \max_j (1 + \delta_s^j) \|e_k\|_2^2 = \max_j (1 + \delta_s^j) \|E\|_2^2$. Considering $\delta_{\kappa s} = \max_j \{\delta_{\kappa s}^j\}$ ($\kappa = 1, 2, 3$), we can rewrite (D.3) as

$$\begin{aligned}
&\left[\left\| X^n - X_S^\# \right\|_2 - \left(\delta_{2s} \left\| X^n - X_S^\# \right\|_2 + (\sqrt{1 + \delta_s}) \left\| \hat{E} \right\|_2 \right) \right] \times \\
&\left[\left\| X^n - X_S^\# \right\|_2 + \left(\delta_{2s} \left\| X^n - X_S^\# \right\|_2 + (\sqrt{1 + \delta_s}) \left\| \hat{E} \right\|_2 \right) \right] \leq \\
&2 \left(\delta_{3s} \left\| GX^{n-1} - X_S^\# \right\|_2 + \sqrt{1 + \delta_{2s}} \left\| \hat{E} \right\|_2 \right)^2
\end{aligned} \tag{D.4}$$

the equation (D.4) is rearranged as

$$\sqrt{1 - (\delta_{2s})^2} \left[\left\| X^n - X_S^\# \right\|_2 - \frac{\sqrt{1+\delta_s}}{1-\delta_{2s}} \left\| \hat{E} \right\|_2 \right] \leq \sqrt{2} \left(\delta_{3s} \left\| GX^{n-1} - X_S^\# \right\|_2 + \sqrt{1 + \delta_{2s}} \left\| \hat{E} \right\|_2 \right) \quad (\text{D.5})$$

dividing both sides of (D.5) to $\sqrt{1 - (\delta_{2s})^2}$ and performing some calculations, we have

$$\left\| X^n - X_S^\# \right\|_2 \leq \sqrt{\frac{2(\delta_{3s})^2}{1-(\delta_{2s})^2}} \left\| GX^{n-1} - X_S^\# \right\|_2 + \left(\sqrt{\frac{2}{1-\delta_{2s}}} + \frac{\sqrt{1+\delta_s}}{1-\delta_{2s}} \right) \left\| \hat{E} \right\|_2 \quad (\text{D.6})$$

using $GX^\# = X^\#$, the above equation is rewritten as

$$\left\| X^n - X_S^\# \right\|_2 \leq \sqrt{\frac{2(\delta_{3s})^2}{1-(\delta_{2s})^2}} \|G\|_{2 \rightarrow 2} \left\| X^{n-1} - X_S^\# \right\|_2 + \left(\sqrt{\frac{2}{1-\delta_{2s}}} + \frac{\sqrt{1+\delta_s}}{1-\delta_{2s}} \right) \left\| \hat{E} \right\|_2 \quad (\text{D.7})$$

References

- [1] E. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information,” *IEEE Trans. Inform. Theory*, 52 (2006), 489–509.
- [2] D. Donoho, “Compressed sensing,” *IEEE Trans. Inform. Theory*, 52 (2006), 1289–1306.
- [3] S. Mallat, “A Wavelet Tour of Signal Processing”. San Diego: Academic Press, (1999).
- [4] S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM J. Scientif. Comput.*, 20 (1) (1998), 33–61.
- [5] R. Tibshirani, “Regression Shrinkage and Selection Via the LASSO”. *Journal of the Royal Statistical Society, Series B* 58 (1996), 267–288.
- [6] B. Efron, T. Hastie, I. Johnston, and R. Tibshirani, “Least angle regression,” *Ann. Statist.*, 32 (2) (2004), 407–499.

- [7] R. Chartrand, W. Yin, “Iteratively reweighted algorithms for compressive sensing” In Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (2008), 3869-3872.
- [8] D. Wipf and B. D. Rao, “Sparse Bayesian learning for basis selection,” IEEE Trans. Signal Process., 52 (8) (2004) 2153–2164.
- [9] S.J. Godsill, A.T. Cemgil, C. Fevotte, P. J. Wolfe, “Bayesian computational methods for sparse audio and music processing”. In: European Signal Processing Conference, (2007) 345-349.
- [10] I. F. Gorodnitsky, J. S. George, and B. D. Rao, “Neuromagnetic source imaging with FOCUSS: A recursive weighted minimum norm algorithm,” J. Electroencephalog. Clinical Neurophysiol., 95 (4) (1995) 231–251.
- [11] I. F. Gorodnitsky and B. D. Rao, “Sparse signal reconstructions from limited data using FOCUSS: A re-weighted minimum norm algorithm,” IEEE Trans. Signal Process., 45 (3) (1997) 600–616.
- [12] B. D. Rao and K. Kreutz-Delgado, “An affine scaling methodology for best basis selection,” IEEE Trans. Signal Process., 47 (1) (1999) 187–200.
- [13] K. Kreutz-Delgado and B. D. Rao, “Sparse basis selection, ICA, and majorization : Toward a unified perspective,” in Proc. ICASSP, Phoenix, AZ, (1999).
- [14] J. F. Murray and K. Kreutz-Delgado, “An improved FOCUSS-based learning algorithm for solving sparse linear inverse problems,” in Proc. Conf. Rec. 35th Asilomar Conf. Signals, Systems Computers, (2001) 347–351.
- [15] S. Mallat and Z. Zhang, “Matching pursuit in a time-frequency dictionary,” IEEE Trans. Signal Process., 41 (12) (1993) 3397–3415.
- [16] Joel A. Tropp, Anna C. Gilbert, “Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit”, IEEE Trans. Inf. Theory 53 (12) (2007) 4655-4666.

- [17] M. Yang and F. “Orthogonal Matching Pursuit With Thresholding and its Application in Compressive Sensing”, *IEEE Trans. On Signal Processing*, 63 (20) (2015) 5479 – 5486.
- [18] D. L. Donoho, Y. Tsaig, I. Drori, and J. L. Starck, “Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit,” *IEEE Trans. Inf. Theory*, 58 (2) (2012) 1094– 1121.
- [19] D. Needell and R. Vershynin, “Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit,” *IEEE J. Sel. Topics Signal Process.*, 4 (2) (2010) 310–316.
- [20] J. Wang, S. Kwon, and B. Shim, “Generalized orthogonal matching pursuit,” *IEEE Trans. Signal Process.* 60 (12) (2012) 6202–6216.
- [21] P. Jain, A. Tewari and I. S. Dhillon, “Orthogonal Matching Pursuit with Replacement,” In *Proc. Neural Inform. Process. Systems Conf. (NIPS)*, (2011).
- [22] S. Chatterjee, D. Sundman and M. Skoglund, “Look ahead orthogonal matching pursuit,” *Proc. ICASSP*, (2011) 4024-4027.
- [23] N. B. Karahanoglu and H. Erdogan, “Compressed sensing signal recovery via forwardbackward pursuit,” *Digit. Signal Process.*, 23 (5) (2013) 1539-1548.
- [24] D. Needell and J. A. Tropp, “CoSaMP: Iterative signal recovery from incomplete and inaccurate samples,” *Appl. Comput. Harmon. Anal.*, 26 (3) (2008) 301–321.
- [25] W. Dai and O. Milenkovic, “Subspace pursuit for compressive sensing signal reconstruction,” *IEEE Trans. Info. Theory*, 55 (5) (2009) 2230–2249.
- [26] S. Foucart, “Hard thresholding pursuit: an algorithm for compressive sensing,” *SIAM J. Numer. Anal.*, 49 (6) (2011) 2543–2563.
- [27] T. Blumensath and M.E. Davies “Iterative Hard Thresholding for Compressed Sensing,” *Applied and Computational Harmonic Analysis*, 27 (3) (2009) 265-274.

- [28] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, “Wireless Sensor Networks: A Survey,” *Computer Networks*, Elsevier Publishers, 38 (4) (2002) 393–422.
- [29] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, “Instrumenting the world with wireless sensor networks,” in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, (2001) 2033-2036.
- [30] G. Azarnia and M. A. Tinati, ‘Steady-state analysis of the deficient length incremental LMS adaptive networks with noisy links’, *AEU-Int. J. Electron. Commun.*, 69 (1) (2015) 153–162.
- [31] H. Shiri, M. A. Tinati, M. Codreanu, S. Daneshvar, “Distributed sparse diffusion estimation based on set membership and affine projection algorithm,” *Digital Signal Processing*, (73) (2018) 47-61. <https://doi.org/10.1016/j.dsp.2017.10.022>.
- [32] H. Shiri, M. A. Tinati, M. Codreanu, G. Azarnia, “Distributed sparse diffusion estimation with reduced communication cost,” *IET Signal Processing*, 12 (8) (2018) 1043-1052. DOI: 10.1049/iet-spr.2017.0377.
- [33] G. Mateos, J. A. Bazerque, and G. B. Giannakis, “Distributed sparse linear regression,” *IEEE Transactions on Signal Processing*, 58 (10) (2010) 5262–5276.
- [34] S. Patterson, Y.C. Eldar, and I. Keidar, “Distributed compressed sensing for static and time-varying networks’. *IEEE Transactions on Signal Processing*, 62 (19) (2014) 4931-4946.
- [35] J.F. Mota, J.M. Xavier, P.M. Aguiar, and M. Puschel, “Distributed basis pursuit”. *IEEE Transactions on Signal Processing*, 60 (4) (2012) 1942-1956.
- [36] S. M. Fossou, J. Matamoros, C. A. Haro and E. Magli “Distributed recovery of jointly sparse signals under communication constraints.” *IEEE Trans Signal Process* 64(13) (2016) 3470–3482.
- [37] P. Di Lorenzo, and A. H. Sayed. ”Sparse distributed learning based on diffusion adaptation.” *IEEE Transactions on signal processing* 61 (6) (2013). 1419-1433.

- [38] Z. Liu, Y. Liu, and C. Li. "Distributed Sparse Recursive Least-Squares Over Networks." *IEEE Trans. Signal Process.* 62 (6) (2014) 1386-1395.
- [39] M. O. Sayin, and S. S. Kozat. "Compressive diffusion strategies over distributed networks for reduced communication load." *IEEE Transactions on Signal Process.* 62 (20) (2014). 5308-5323.
- [40] S. Xu, R. C. de Lamare, and H. V. Poor. "Distributed compressed estimation based on compressive sensing." *IEEE Signal Processing Letters*, 22 (9) (2015) 1311-1315.
- [41] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, 24 (2) (1995) 227–234.
- [42] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via l1 minimization," *Proc. Nat. Acad. Sci. USA*, 100 (5) (2003) 2197–2202.
- [43] S. Foucart, H. Rauhut, "A Mathematical Introduction to Compressive Sensing", Springer, New York, (2013).
- [44] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *C. R. Math. Acad. Sci. Paris, Ser. I*, 346 (9) (2008) 589–592.
- [45] A. S. Bandeira, E. Dobriban, D. G. Mixon, W. F. Sawin, "Certifying the Restricted Isometry Property is Hard", *IEEE Trans. Inf. Theory*, 59 (6) (2013) 3448-3450.