Compressive networked storage with lazy-encoding

Siwang Zhou, Shuzhen Xiang, Xingting Liu
College of Computer Science and Electrical Engineering,
Hunan University, Changsha, China
{swzhou, shuzhenxiang, xingtingliu}@hnu.edu.cn

Yonghe Liu
Department of Computer Science and Engineering
the University of Texas at Arlington, Arlington, USA
yonghe@cse.uta.edu

***Abstract*—We investigate the problem of distributed networked storage with compressive sensing in wireless sensor networks, and a compressive storage scheme for local data query is proposed. Specifically, we propose a simple but efficient one-step data dissemination strategy, and the dissemination cost is reduced dramatically. We further present a lazy-encoding algorithm, using which the local data are capable of being reconstructed without recovering the global data field if not necessary. Thus the decoding ratio decreases significantly. Experiments using real sensor data show that the proposed scheme achieves far better local data recovery performance compared to the existing ones.**

Keywords: Compressive sensing; Data storage; Sensor network

I. Introduction

Networked systems, such as wireless sensor networks (WSNs) or internet of things (IoT), can involve thousands of independent nodes including sensors, RFID tags, or mobile phone, which are all capable of generating and communicating data [1, 6]. In certain network deployments, one or multiple powered sinks exist, allowing network nodes to send their sensed data to the sink via a routing protocol. However, in remote geographical regions, it may be impractical to deploy static sinks and nodes are required to temporarily store their readings before a mobile sink, such as an unmanned aerial vehicle, visits the network and gathers the data. Since network nodes are energy-constrained and prone to failures, it is critical to energy-efficiently store sensor readings in the network so that the mobile sink can recover the original data field by visiting a portion of nodes at any time.

Data Storage for a WSN with n sensor nodes can be compactly represented as $y = \phi x$ where y is an $m \times 1$ vector of stored data, x is $n \times 1$ sensor readings and ϕ is an $m \times n$ measurement matrix with entries corresponding to the readings that have been combined in the storage nodes [7]. Compressive sensing (CS) [8, 16], an emerging signal sampling approach, can naturally serve as a possible solution to the data storage problem for networked systems. Indeed, a number of approaches applying CS theory to distributed networked data storage systems have been introduced [9, 10, 12–15, 19]. Specifically, in [14, 15], a data dissemination algorithm, called CStorage, is presented for WSNs. CStorage takes advantage of the topology information to select the optimal forwarding nodes, avoiding redundant data transmission related to broadcast. Combing a Metropolis-Hastings random walk algorithm, the authors in [12, 13] propose a distributed CS data storage approach in WSNs. During their encoding process, each node ejects a number of random walks, and the lengthes are set to be sufficiently long for the probability distribution to reach the equilibrium. Generally, these approaches introduce various data disseminating strategies to construct CS measurement matrix, aiming to recover the data field in the entire network. Unfortunately, to recover the global data field, existing approaches require the dissemination of sensor readings throughout the network. This can potentially introduce long convergence time and excessive energy consumption.

At the same time, we note that in the applications of WSNs, users may only need to query relevant regions where monitoring events may occur, and recovering the global data field is actually not necessary. We also note that the movement of the mobile sink can be utilized to generate CS measurement, and sensor readings do not have to be disseminated in the entire network. Our goal hence is to design a compressive data storage strategy with significantly lower dissemination cost while achieving high local data recovery accuracy. Our contributions of the paper are summarized below.

(1) We propose a lazy-encoding algorithm, enabling recovery of a local data field without recovering the global field if not necessary. We show this can significantly reduce the decoding ratio, and thus greatly decrease the number of sensor nodes needed to be queried by the mobile sink.

(2) We present a data dissemination strategy with one step without disseminate readings throughout the networks, and thus dissemination cost declines dramatically. We further provide its mathematical foundation.

The remainder of this paper is organized as follows. Section II describes the proposed compressive networked data storage scheme. Section III provides the theoretical analyse and Section IV demonstrates its effectiveness through simulations. Finally, we conclude the paper in Section V.

II. Compressive networked data storage scheme

In this section, we introduce a compressive networked data storage (CNDS) scheme. We first present a data dissemination strategy with one step, then propose a lazy-encoding algorithm. After that, we make a brief discussion.

A. Data dissemination with one step

Given a sensor s_i as the source node, the next nodes are chosen from its neighbors in the reading dissemination

This work has been supported by the NSFC under Grant No. 61672221.

process. We note that, different from the existing approaches [9, 10, 12–15, 19], the neighbor nodes that received the reading of s_i no longer further forward it. That is to say, data are disseminated only with one step, and do not have to be disseminated in the entire network.

We further design a data packet, denoted by sp_i for node s_i $(0 \leq i \leq n-1)$, to store the reading received. sp_i has two independent components, where the first component $sp_i\{0\}$ contains the original reading x_i and the second one $sp_i\{1\}$ is the corresponding index s_i.

When node s_i receives a reading x_j from one of its neighbors s_j, it stores the received reading by updating its own sp_i as following

$$sp_i : \begin{cases} sp_i\{0\} = sp_i\{0\} \cup \{x_j\} \\ sp_i\{1\} = sp_i\{1\} \cup \{s_j\} \end{cases} \quad (1)$$

Our one-step data dissemination strategy is very simple but efficient. All neighbor nodes hold the information of s_i, and it makes our storage scheme be reliable and have fault-tolerant guarantee. Moreover, the proposed one-step dissemination strategy is capable of generating a measurement matrix with desired property, serving the successful data recovery.

B. Lazy-encoding algorithm

When a reading is received by sensor node s_i, we do not perform encoding operation right away. Instead, the reading is temporarily stored in sp_i . CS encoding is performed only when s_i receives a query instruction from a mobile sink. This is the reason that it is termed lazy-encoding.

1) Determining which data involve in encoding process: Suppose that r nodes are deployed in a local region Re in a WSN with n nodes, and node s_i is queried by a mobile sink. Let those r nodes form a node subset $\mathcal{R}$.

Once a node s_i ($s_i \in \mathcal{M}$) receives the query instruction from the mobile sink, it has to determine which data in its stored data packet sp_i should be involved in the encoding process. s_i checks the sp_i to find out which data have been included in $sp_i\{0\}$. We note that not all those nodes in $sp_i\{0\}$ participate in the encoding process. Only the data corresponding to node set $\mathcal{NS}_i$ are encoded by node s_i, where $\mathcal{NS}_i$ is

$$\mathcal{NS}_i = sp_i\{0\} \cap \mathcal{R} \quad (2)$$

If the mobile sink queries the entire network, that is, $|\mathcal{R}|_0 = r = n$, then $\mathcal{NS}_i = sp_i\{0\} \cap \mathcal{R} = sp_i\{0\}$. In this case, all of data in sp_i will be encoded, and our lazy-encoding is simplified into a general encoding algorithm.

Fig. 1 illustrates our idea with a simple example. The mobile sink has to query the data field in the round region with dotted line, which is monitored by the node set $\mathcal{R} = \{s_2, s_3, s_4, s_8, s_9, s_{10}, s_{15}\}$. The reading x_{15} is sent to its neighbors and s_{15} receives the readings disseminated from those neighbor nodes. Therefore we have $sp_{15}\{0\} = [x_8, x_{10}, x_{13}, x_{14}, x_{15}, x_{16}]$. The sink randomly queries a part

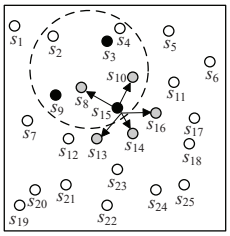


Fig. 1: A simple example. s_{15} has 5 neighbor nodes: s_8, s_{10}, s_{13}, s_{14} and s_{16}. The circle with dotted line represents a local region queried by a mobile sink, which randomly selects s_3, s_9, and s_{15} to gather CS measurements.

of nodes in the local region, and node s_{15} are assumed to be included. Then,

$$\mathcal{NS}_{15} = sp_{15}\{0\} \cap \mathcal{R} = \{s_8, s_{10}, s_{15}\}$$

2) Generating a measurement matrix: We assume that $\mathcal{NS}_i = \{s_i, s_j, s_k, s_l\}$ and use it as an example. Accordingly, s_i generates four CS measurement coefficients, denoted by $\phi_{i,i}$, $\phi_{i,j}$, $\phi_{i,k}$ and $\phi_{i,l}$, respectively.

Those coefficients are randomly selected from $\{-1, +1\}$ with equal probabilities or randomly generated with $\mathcal{N}(0,1)$, where $\mathcal{N}(0,1)$ is the zero mean and unit variance Gaussian distribution. Using those coefficients, s_i generates a measurement vector ϕ_i as following

$$\phi_i = \frac{1}{\sqrt{m}} \begin{pmatrix} 0 & \phi_{i,i} & 0 & \phi_{i,j} & 0 & \phi_{i,k} & 0 & \phi_{i,l} & 0 \end{pmatrix} \quad (3)$$

where m is the number of nodes queried by the mobile sink. Factor $1/\sqrt{m}$ will be used for obeying the mutual coherence property between ϕ and a sparse basis ψ. The length of ϕ_i is r because there are r nodes or r data in the region. The numbers of zero in ϕ_i are i, $j-i+1$, $k-j+1$, $l-k+1$ and $r-l+1$, respectively.

The mobile sink randomly queries m nodes in the region, which generate their own measurement vectors, respectively. Those m vectors form a measurement matrix $\phi = (\phi_0, \phi_1, \ldots, \phi_{m-1})^T$.

3) Encoding: For node s_i, it computes its measurement using the measurement vector ϕ_i as $y_i = \phi_i x$, where $x = (x_0, x_1, \cdots, x_{r-1})^T$. In this way, all r data in the region, not merely x_i, x_j, x_k and x_l, are encoded as a CS measurement y_i. In other words, y_i includes all the information of those r readings. m nodes that have received the query instruction independently compute their own measurements, respectively. Therefore, the encoding process in the region is expressed as the following formula

$$y = \phi x \quad (4)$$

where $y = (y_0, y_1, \cdots, y_{m-1})^T$ and ϕ is the corresponding measurement matrix.

When the m measurements $y_0, y_1, \cdots, y_{m-1}$ are received by the mobile sink, it performs decoding and reconstructs the original data field $x = (x_0, x_1, \cdots, x_{r-1})$ in this local region.

Let ψ be an orthogonal basis and $\alpha = \psi^{-1}x$. Then $x = \psi\alpha$, and $y = \phi x = \phi\psi\alpha = \Phi\alpha$, where $\Phi = \phi\psi$. Therefore the original x can be reconstructed by solving the following l_1-norm problem

$$min \parallel \alpha \parallel_1, \quad s.t. \quad y = \Phi\alpha \tag{5}$$

A number of CS reconstruction algorithms, such as basis pursuit [5], orthogonal matching pursuit [18], and iterative thresholding [11], have been introduced to solve the above mentioned l_1 norm minimization equation.

C. Discussion

In this section, we exploit a strategy, called lazy-encoding algorithm, to recover local data. To encode the data field in a region, a sensor node has to temporarily store the readings disseminated from other nodes, and this incurs extra storage overhead. Fortunately, in our scheme, sensor readings do not have to be disseminated in the entire network, and the storage spending is very limited, thanks to the proposed one-step dissemination strategy. Indeed, each node only needs to temporarily store the readings from its neighbor nodes.

In the proposed CNDS scheme, m nodes in a local region are *randomly* queried to gather data with the help of the movement of the mobile sink. At the same time, each node independently generates the coefficients of measurement matrix ϕ. In the following section, we are going to show that, it is the randomness and the independence property that serve the necessary condition of successful data recovery.

III. Theoretical analysis

In this section, we first show that our CNDS scheme satisfies the conditions of successful CS recovery. Then we evaluate its dissemination cost.

The restricted isotropy property (RIP) is considered as the necessary condition for successful CS reconstruction for sparse or compressible signal [3, 8]. Authors in [4] propose an alternative mechanism to relax the RIP condition, and prove that the measurement matrix with the isotropy property and the incoherence property are capable of guaranteeing the successful CS reconstruction [4].

Theorem III.1. *Suppose that an n-length x is a sensing data set and it is sparse in an $n \times n$ orthogonal basis ψ. x is stored in a WSN with n nodes by the proposed one-step dissemination strategy. A mobile sink randomly queries m nodes in a region in the WSN, and the corresponding $m \times n$ measurement matrix ϕ is generated by the proposed lazy-encoding algorithm. Let $\Phi = \phi\psi$. Then the matrix Φ holds both the isotropy property and the incoherence property.*

Proof. Denote ϕ_i and ϕ_j as the i^{th} and the j^{th} column vectors in ϕ, respectively. In the lazy-encoding process, the entries in ϕ_i and ϕ_j are independently generated by m nodes that locate in the dissemination area of the readings originating from node s_i and s_j, respectively. At the same time, m nodes are randomly selected by the mobile sink. Therefore ϕ_i is irrelevant from ϕ_j. Furthermore, the entries in ϕ_i and ϕ_j are randomly generated from selected from $\{-1, +1\}$ or with Gaussian distribution. Thus $\mathbb{E}(\phi_i^T\phi_j) = 0, j \neq i$ and $\mathbb{E}(\phi_i^T\phi_i) = 1$, where $\mathbb{E}(\cdot)$ denotes the mathematical expectation. Therefore we have $\phi\phi^T = I_n$. Then $\mathbb{E}(\Phi^T\Phi) = \mathbb{E}((\phi\psi)^T(\phi\psi)) = \mathbb{E}(\psi^T\phi^T\phi\psi) = I_n$, since ψ is an orthogonal transform basis. It has been illustrated in [4] that an $m \times n$ matrix X satisfies the isotropy property if and only if $\mathbb{E}(X^TX) = I_n$. So our matrix Φ holds the isotropy property.

Let Φ_{ij} be the entry on the i^{th} row and j^{th} column in Φ. Then coherence measure $\mu(\Phi)$ can be expressed as $\mu(\Phi) = \max_{i,j} \left| \sum_{k=1}^{n} \phi_{ik}\psi_{kj} \right|$. The entry ϕ_{ik} is generated by node s_i. If the reading originating from s_k is received by s_i, then s_i assigns a random coefficient to ϕ_{ik}. Otherwise ϕ_{ik} is set to 0. Therefore ϕ is an entirely random sparse matrix, since readings are disseminated with only one step. According to [2, 3], any fixed basis is incoherent with a random matrix with a high probability. Therefore ϕ and ψ have very low mutual coherence measure. i.e., $\mu(\phi\psi) < c$. Here, c is some positive numerical constant. Hence Φ obeys the incoherence property. □

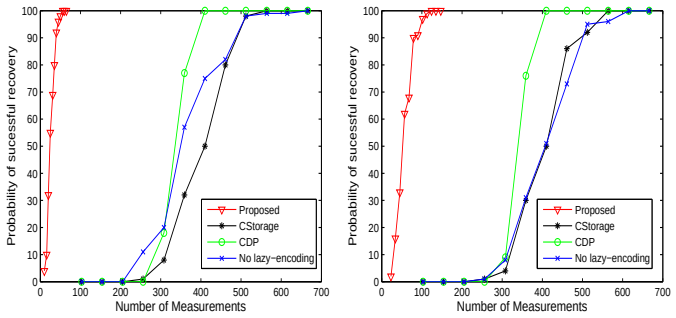
We assume that each data transmission from a node to its neighbor node has the same energy consumption, i.e., the data dissemination cost is proportional to the number of data transmission. Suppose that node s_i has l_i neighbors in a WSN with n nodes. It is clear that the global dissemination complexity is $O(n^2)$. However, for our one-step dissemination strategy, the dissemination cost of a reading is $\sum_{i=0}^{n-1} l_i$, and the complexity is only $O(n)$, since l_i is a constant. Therefore our scheme can guarantee successful data recovery with significantly lower dissemination cost.

IV. Performance evaluation

In this section, we evaluate the performance of the proposed CNDS scheme and compare it with the classical CStorage [15] and the random walk based CDP approach [13].

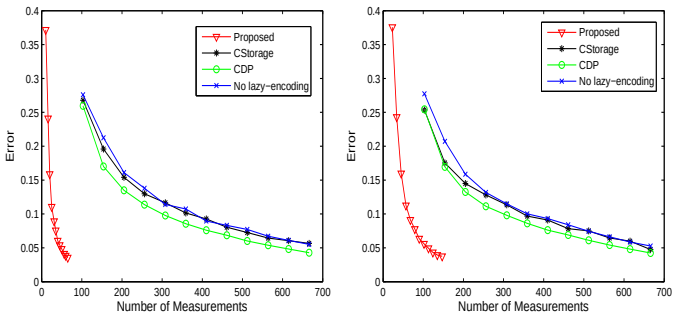
A. Parameter settings

The simulated WSN consists of n nodes, covering an area of V units. Each sensor node is responsible for sensing, forwarding and storing environmental data, which are real data from the sea surface temperatures data set [17]. The communication radius of a sensor node R satisfies $R^2 = CV \log n/n$ where C is a positive constant. The decoding ratio is set as the ratio of the number of measurements m and the number of nodes n, i.e., m/n. Discrete cosine transform is used as the sparse transform on sensing data field. The coefficients in the measurement matrix are randomly selected from $\{+1, -1\}$ with equal probabilities. We use $e(x, x') = \frac{\|x-x'\|_2}{\|x\|_2}$ to evaluate the data reconstruction quality where x and x' represent the original and recovered sensing readings, respectively. Obviously, the smaller e is, the better the reconstruction quality is. In the simulation, we say one reconstruction is successful if $e(x, x') < 0.09$. All results are carried out from a laptop platform with a CPU @ 2.5 GHz of Intel(R) Core (TM) i5-7300HQ using the MATLAB



(a) A region with 100 nodes (b) A region with 225 nodes

Fig. 2: The probabilities of successful local data recovery.



(a) A region with 100 nodes (b) A region with 225 nodes

Fig. 3: The reconstruction error for local regions.

R2015b simulator. Basis pursuit (BP) algorithm [5] is used to reconstruct the original data.

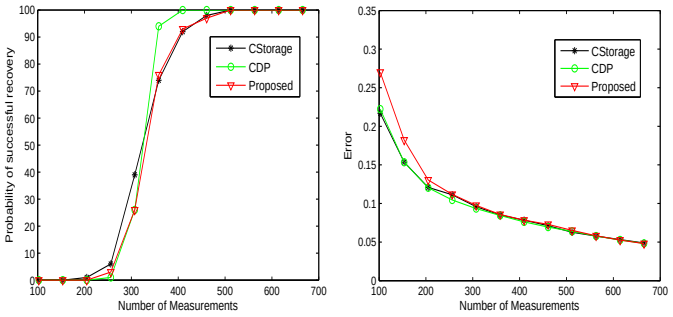
B. Results and analysis

1) Recovering the local data fields: In this subsection, we evaluate the performance of the proposed CNDS strategy by recovering the data fields in specific local regions, and compare them with the existing approaches. In CDP, the random walk instances are fixed at 30, and the random walk step is set to 20. For CStorage, the forwarding probability is set to 0.24 and the number of source nodes is 600. We select those parameters because they obtain almost the best performance in the corresponding approaches.

Here we perform 100 repeated simulations. The probabilities of successful reconstruction along with the increasing of the number of measurements are shown in Fig. 2. The results of recovery accuracy are illustrated in Fig. 3. It is clear that our scheme has far better performances than the existing ones. The decoding ratio decreases significantly with the same recovery probability or accuracy. In other words, the number of sensor nodes that need to be queried by mobile sink for recovering original data in the corresponding regions is reduced significantly. This is because our lazy-coding algorithm does not compute the measurements until the mobile sink launches the query instruction. In this way, we are capable of only encoding the information in the region queried by the sink, and the data fields in the specific local regions can be reconstructed separately without recovering the global data field. This is further illustrated by the experimental finding of "No lazy-encoding" that is our version without lazy-encoding process.

2) Recovering the global data field: This subsection evaluates the performance of our scheme when recovering the global data. The parameter setting is the same as the above subsection. The reconstruction error and the probability of successful recovery are shown in Fig. 4. The experimental results that the proposed CNDS can also recover the whole data field in the network without deceasing the accuracy.

As we illustrate in section I, different dissemination strategies in distributed data storage approaches will generate different measurement matrixes. Theoretically, the measurement matrixes with better properties have better data recovery performance. However, as shown in Fig. 4, the performance gains of the existing approaches by carefully designing dissemination strategy are approximately the same in the context of the networked storage. Our scheme with one-step dissemination also has about the same reconstruction probability and accuracy. Although readings are disseminated with only one step, the movement of the mobile sink can be utilized to randomly and uniformly select nodes being queried, and thus the measurement matrix generated by our CNDS scheme can satisfy the isotropy property and the incoherence property that guarantee the data recovery performance.



(a) Recovery probability (b) Reconstruction error

Fig. 4: The global data recovery performances.

V. Conclusion

In this paper, we exploit the data storage problem in networked systems, a compressive data storage scheme with lazy-encoding, called CNDS, is proposed. CNDS utilizes the lazy-encoding algorithm to compute measurements, and local data fields are thus capable of being recovered by querying far less sensor nodes. The energy cost of the proposed one-step disseminating process decreases significantly since the sensor readings do not have to be disseminated throughout the network. Theoretically and experimentally, for local data fields, our proposed CNDS scheme has far better recovery performance. At the same time, our scheme can recover the global data field as well if necessary, without decreasing accuracy.

REFERENCES

[1] Ian F Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A survey on sensor networks. *IEEE Communications magazine*, 40(8):102–114, 2002.

[2] Emmanuel Candes, Yaniv Plan, et al. Near-ideal model selection by $l1$ minimization. *The Annals of Statistics*, 37(5A):2145–2177, 2009.

[3] Emmanuel Candes and Justin Romberg. Sparsity and incoherence in compressive sampling. *Inverse Problems*, 23(3):969, 2007.

[4] Emmanuel J Candes and Yaniv Plan. A probabilistic and ripless theory of compressed sensing. *IEEE Transactions on Information Theory*, 57(11):7235–7254, 2011.

[5] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.

[6] Xiaoyi Cui. The internet of things. In *Ethical Ripples of Creativity and Innovation*, pages 61–68. Springer, 2016.

[7] Alexandros G Dimakis, Vinod Prabhakaran, and Kannan Ramchandran. Distributed fountain codes for networked storage. In *Proceedinf of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 5, pages V–V. IEEE, 2006.

[8] David L Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[9] Bo Gong, Peng Cheng, Zhuo Chen, Ning Liu, Lin Gui, and Frank de Hoog. Spatiotemporal compressive network coding for energy-efficient distributed data storage in wireless sensor networks. *IEEE Communications Letters*, 19(5):803–806, 2015.

[10] Jarvis Haupt, Waheed U Bajwa, Michael Rabbat, and Robert Nowak. Compressed sensing for networked data. *IEEE Signal Processing Magazine*, 25(2):92–101, 2008.

[11] Shidong Li, Yulong Liu, and Tiebin Mi. Fast thresholding algorithms with feedbacks for sparse signal recovery. *Applied and Computational Harmonic Analysis*, 37(1):69–88, 2014.

[12] Mu Lin, Chong Luo, Feng Liu, and Feng Wu. Compressive data persistence in large-scale wireless sensor networks. In *Proceeding of IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–5. IEEE, 2010.

[13] Feng Liu, Mu Lin, Yusuo Hu, Chong Luo, and Feng Wu. Design and analysis of compressive data persistence in large-scale wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 26(10):2685–2698, 2015.

[14] Ali Talari and Nazanin Rahnavard. Cstorage: Distributed data storage in wireless sensor networks employing compressive sensing. In *Proceeding of IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–5. IEEE, 2011.

[15] Ali Talari and Nazanin Rahnavard. Cstorage: Decentralized compressive data storage in wireless sensor networks. *Ad Hoc Networks*, 37:475–485, 2016.

[16] Weiyu Xu, Enrique Mallada, and Ao Tang. Compressive sensing over graphs. In *Proceeding of IEEE International Conference on Computer Communications(INFOCOM)*, pages 2087–2095. IEEE, 2011.

[17] Xi Xu, Rashid Ansari, Ashfaq Khokhar, and Athanasios V Vasilakos. Hierarchical data aggregation using compressive sensing (hdacs) in wsns. *ACM Transactions on Sensor Networks (TOSN)*, 11(3):45, 2015.

[18] Mingrui Yang and Frank de Hoog. Orthogonal matching pursuit with thresholding and its application in compressive sensing. *IEEE Transactions on Signal Processing*, 63(20):5479–5486, 2015.

[19] Xianjun Yang, Xiaofeng Tao, Eryk Dutkiewicz, Xiaojing Huang, Y Jay Guo, and Qimei Cui. Energy-efficient distributed data storage for wireless sensor networks based on compressed sensing and network coding. *IEEE Transactions on Wireless Communications*, 12(10):5087–5099, 2013.