Random Sampling in Collaborative and Distributed Mobile Sensor Networks utilizing Compressive Sensing for Scalar Field Mapping

Minh Tuan Nguyen^{*} and Keith A. Teague School of Electrical and Computer Engineering Oklahoma State University, Stillwater, OK 74078, Emails: [tuanminh.nguyen, keith.teague]@okstate.edu

Abstract—In this paper, we propose an algorithm supporting distributed mobile sensor networks (MSN) for scalar field mapping that has many applications such as environmental monitoring or battle field surveillance, etc. We exploit the integration between compressive sensing (CS) and the collaboration of the mobile sensors. In the algorithm each distributed mobile sensor measures at random positions in the sensing area to create one CS measurement and finally shares the measurement with others by communicating through its neighbors. The convergence time is considered while the sensors exchange their measurements. After all the sensors achieve the number of CS measurements needed, a CS recovery algorithm is applied at each mobile sensor to reconstruct sensory readings from all the positions in the sensing area that need to be observed. The total communication energy consumption is formulated, analyzed and simulated.

Keywords—Compressive sensing, Scalar field mapping, Mobile sensor networks

I. INTRODUCTION

A. Motivation

In recent years, measuring and exploring an unknown field of interest have attracted much attention. The advance networks that are often deployed in the field are mobile sensor networks (MSN) that facilitate many application areas, such as monitoring temperatures, humidity, acoustics, vibrations or detecting events. The network is a combination of sensors, control algorithms and other dynamic factors depending on specific purposes or application scenarios [1], [2]. Sensors are often attached to mobile robots or vehicles. They need to move and to visit the entire of the sensing area in order to collect data from all positions. In scalar field mapping in a distributed fashion, those sensors need to collect or estimate data within the area cover by their sensing ranges. They only need to collaborate with their neighbors to exchange their data and to build the sensing maps themselves [3], [4]. We are looking for an energy-efficient data collection method to prolong the network lifetime.

Compressive sensing (CS) [5], [6], [7], [8] changes the traditional data collection methods either in wireless sensor

networks (WSN) or in MSNs. The technique offers a novel framework to reconstruct all sensor readings from all positions that need to be observed in the sensing area based on a much smaller number of CS measurements compared to the total number of the positions. Each CS measurement can be collected from all sensor nodes or from only some random nodes. We prefer to sample data from some random nodes in order to save both energy for the motion of sensors and their communications. This way of sampling data results in different measurement matrices or different reconstruction errors in the CS recovery process that will be mentioned in the next section.

In our paper, we assume a distributed MSN with all sensors mobile and deployed randomly in a sensing area. The mobile sensors move and measure data from random positions in the area. At a given position each sensor chooses randomly one of positions within its sensing range to visit. It keeps adding new data when it reaches a new position as one scalar value while moving. After a pre-defined number of walking steps, also called random walk length, the mobile sensors stop adding data and they all have one CS measurement themselves. They exchange the measurements to each other through their neighbors. Each mobile sensor can implement a CS recovery algorithm to reconstruct all readings from the sensing area for scalar field mapping. This method is considered energyefficient since all the mobile sensors only have to move into some positions and there are only a certain number of CS measurements to be sent among the sensors. This work shows promise not only in MSNs but also in distributed robot networks or vehicle networks for data monitoring or similar purposes.

B. Related Work

Employing CS in WSNs provides energy-efficient data collection methods [9], [10]. Based on a small certain number of CS measurements required, all the sensory data can be recovered precisely at the sink node that would be a promising point for a lot of applications in such networks. Some data collection methods utilizing CS are proposed as energy efficient algorithms to reduce energy consumption for sensors. In [11], [12] random walk with CS provides distributed routing methods for WSNs. Cluster-based [13], [14] and tree-based [15], [16] data collection methods significantly show the energy reduced based on the combination with CS.

^{*}Minh T. Nguyen is a lecturer with Thai Nguyen University of Technology (TNUT), Thai Nguyen city, Vietnam. This work is supported by the Vietnamese Ministry of Education and Training (MOET) and School of Electrical and Computer Engineering (ECE), Oklahoma State University.

Data collection problems in MSNs are different from WSNs since some or all the sensors in such networks are mobile. They can collect or evaluate data while moving. There have been many research studies that exploit the combination between the mobility of sensors and CS. In [17], only one mobile sink and some mobile sensors are used to gather sensory data. Wang [18] monitors vehicle networks based on CS. Mostofi builds maps in mobile networks [19] and robot networks [20] while the mobile sensors and robots are deployed outside the sensing areas. In [3], [4] scalar field maps are built in MSNs based on the consensus methods [21], [22]. The mobile sensors are led by the flocking control algorithm [23].

Our paper exploits the random routing methods [11], [12] to collect sensor readings while moving into random positions. All the sensors can move randomly in the sensing area to visit random positions. A random walk length is pre-defined for all the mobile sensors to sample a certain number of random positions. Each sensor creates one CS measurement and shares with others through its neighbors. A scalar field map is created based on the reconstructed data after each sensor achieves all CS measurements from the others.

The main contributions in this work are summarized as follows:

- 1) A new distributed compressive and collaborative sensing algorithm for MSNs to build the map of a scalar field *at each mobile sensor* is proposed.
- All transmission energy consumptions for the network are formulated and analyzed.
- 3) Some important factors such as the number of mobile sensors, the convergence time and the sensor communication range are analyzed and simulated to minimize the network energy consumption.

The rest of the paper is organized as follows. The Background and Problem Formulation are mentioned in Section II. The Random Data Collection Algorithm and the Communication Energy Consumption Analysis are addressed in Section III and Section IV, respectively. Simulation results are shown in Section V. Finally, conclusions and suggestions for future work are presented in Section VI.

II. BACKGROUND AND PROBLEM FORMULATION

A. Compressive Sensing (CS) Overview

1) Sparse presentation of signals: CS [5], [6], [7] offers novel techniques to recover a compressible signal from its undersampled random projections, also called measurements. A signal $\underline{X} = [x_1 x_2 \dots x_N]^T \in \mathbb{R}^N$ is defined to be k-sparse if it has a sparse representation in a proper basis $\Psi = [\Psi_{i,j}] \in \mathbb{R}^{N \times N}$, where $\underline{X} = \Psi \underline{\Theta}$ and $\underline{\Theta}$ has only k nonzero elements. Based on the CS paradigm, a k-sparse signal can be under-sampled and be recovered from only $M \ll N$ random measurements $\underline{Y} = [y_1 y_2 \dots y_M]^T \in \mathbb{R}^M$.

2) Signal sampling: The CS measurements are generated by $\underline{Y} = \Phi \underline{X}$, where $\Phi = [\varphi_{i,j}] \in \mathbb{R}^{M \times N}$ is called the measurement matrix and is often a dense Gaussian matrix or a sparse binary matrix [24]. The *i*th element in the measurement vector \underline{Y} is formed by $y_i = \sum_{j=1}^{N} \varphi_{i,j} x_j$. 3) Signal recovery: : It has been shown that we can reconstruct a k-sparse signal with high probability from only M = O(klogN/k) CS measurements [9] employing the following l_1 optimization problem

$$\underline{\hat{\Theta}} = \arg\min \|\underline{\Theta}\|_1, \ subject \ to \ \underline{Y} = \Phi \Psi \underline{\Theta}, \tag{1}$$

where $\| \Theta \|_1 = \sum_{i=1}^N |\Theta_i|$ and $\underline{\hat{X}} = \Psi \underline{\hat{\Theta}}$. The l_1 optimization problem can be solved with linear programming techniques such as Basis Pursuit (BP) [5].

B. Random Walk

Random walk (RW) on a graph can be modeled as a Markov chain mentioned in [25], [26], [27]. A RW at a position chooses one position randomly within a radius to visit consecutively. Such probabilities form a transition probability matrix $P = [p_{ij}]_{N \times N}$, where N is the total number of vertexes. For example, a simple random walk at a step of time k needs to move from vertex i to one of its adjacent vertexes j with a probability p_{ij} . The transition probability is calculated as follows

$$p_{i,j} = P(X_{k+1} = j \mid X_k = i) = \begin{cases} \frac{1}{d(i)}, & \text{if } (i,j) \in E\\ 0, & \text{others,} \end{cases}$$
(2)

where d(i) denotes the degree of vertex *i*. This matrix characterizes the Markov chain that is an important model for random walks on graphs which satisfies

$$0 \le p_{\{ij\}} \le 1; \quad \sum_{j=1}^{N} p_{\{ij\}} = 1; \quad i, j = 1, 2, \dots N.$$
 (3)

In order to visit the positions equally, a RW length for all mobile sensors is chosen as the mixing time which is defined as

Definition: Mixing time is the number of steps before the distribution of a random walk be stationary. It shows how fast a RW converges to its stationary distribution.

The mixing time of RWs has been studied well in [26], [28], [29] showing that it significantly depends on the transmission range or the transition probability p_{ij} . It measures the number of steps for the distribution to reach the stationary distribution. As mentioned in [26], the asymptotic rate of convergence of the Markov chain to the uniform equilibrium distribution is determined by the second largest eigenvalue of the transition probability matrix P as follows

$$\mu(P) = \max_{i=2,\dots,n} |\lambda_i(P)| = \max[\lambda_2(P), \lambda_n(P)].$$
(4)

Since the graph is irreducible and aperiodic, then $\mu(P) < 1$ and the distribution converges to uniform asymptotically. The mixing time τ can be calculated as

$$\tau = 1/\log(1/\mu). \tag{5}$$

C. Problem Formulation

1) System model: In the network model considered here, we assume that M mobile sensors are deployed uniformly at random in a sensing area. We also assume that there are about N random positions that need to be observed in the sensing area. These positions are fixed and are known to the mobile sensors if covered by their sensing ranges. Based on a sensing



Fig. 1. A communication network with 100 mobile sensors randomly deployed in a square sensing area dimensioned $[0, 100]^2$ that has 500 random positions need to be observed.

range, denoted as R_s , there are some nodes within the range that each sensor can choose one of them to visit for sampling data.

If an appropriate sensing range is chosen, all possible routes for the mobile sensors between all the positions can be considered as an undirected connected graph G(V, E), where V represents all the random positions and E represents all the possible routes that the mobile sensors may choose as it visits positions. An appropriate communication range is chosen for all sensors to share their CS measurements with each other.

2) Data collection problem: M mobile sensors start collecting data from M random positions in the sensing area. Based on a sensing range, each mobile sensor has a few choices to visit one of some positions around, as shown in Figure 1. The pre-defined random walk length τ is calculated based on Section II-B. After τ walking steps, each mobile sensor visits and adds τ readings from those positions to create one CS measurement as $y_i = \sum_{j=1}^{\tau} x_j$. Each CS measurement is a linear combination as one scalar value that adds all readings from positions which each mobile sensor visits. Then, the mobile sensor start a communication phase to share the measurements with each other within sensor communication range, denoted as R_c . Finally, each sensor has M measurements ($\underline{Y} = [y_1 y_2 \dots y_M]^T \in \mathbb{R}^M$) to be contributed to the CS data recovery process as mention in Section II-A.

The accuracy of the reconstructed data or the scalar map depends on the number of CS measurements, denoted as M. The greater the number of measurements, the more precise reconstructed data we can achieve from the CS recovery process. We can deploy or add more mobile sensors in the sensing area to build a more accurate map, or in case some mobile sensors are deactivated, and to reduce the reconstruction error.

3) The measurement matrix: Each CS measurement is attached with a record that shows which positions contribute readings to the measurement or which positions are visited by the mobile sensor. Each record contributes to one row of the measurement matrix Φ . The measurement matrix created by collecting M measurements as $\underline{Y} = \Phi \underline{X}$ is a sparse binary

matrix as shown in Equation 6.

$$\Phi = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & \dots & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & \dots & 1 \\ \dots & \dots \\ 1 & 0 & 1 & 0 & 0 & 1 & \dots & 0 \end{bmatrix}_{M \times N}$$
(6)

The restricted isometry property (RIP) of the sparse binary matrix has been studied in [24] and it has been shown that the matrix can satisfy RIP and therefore can be used as an efficient measurement matrix. This matrix ($\Phi_{M \times N}$) can work as well as the full Gaussian matrix for the CS recovery process [24] that will be shown in the simulation section.

4) The convergence time: The convergence time, denoted as C, shows how fast the mobile sensors share their CS measurements with each other. If each sensor connects directly to the others, the convergence time equals one since the sensors have to transmit data to their neighbors only once. In this case, we have to deal with a huge energy consumption due to the long communication distances. We need to choose an appropriate communication range in order to minimize the total energy consumption for the network.

III. RANDOM DATA COLLECTION ALGORITHM

The proposed random data collection in distributed and collaborative MSNs is summarized in the Algorithm 1. In the data collection phase each mobile may visit a position twice due to a random probability. It would not add the data from the position since it already stores the position's index. This does not affect the CS performance in signal recovery.

In other cases, if we cannot deploy enough M mobile sensors in the sensing area, in order to build a map with an accuracy that requires M CS measurements, then each mobile sensor needs to collect more than one CS measurements as

$$M = t * L, \tag{7}$$

where L represents the number of mobile sensors deployed and t is the number of rounds or the number of CS measurements each mobile sensor has to collect. As shown in Equation 7 deploying fewer mobile sensors for scalar field mapping results in more CS measurements that each mobile sensor has to collect with the same RW length τ .

IV. COMMUNICATION ENERGY CONSUMPTION ANALYSIS

In this section, we assume a network modeled as mentioned in Section II-C (L = M and t = 1). After generating one CS measurement at each mobile sensor, all the mobile sensor start the communication phase (phase 2) to share their own measurements with each other. We only consider the noiseless environment throughout this paper and leave the noisy one for future work.

The total consumed energy for sensor communications contains two main elements: the consumed power for communications within all sensor neighborhoods and the convergence time C which represents the number of times sensors keep sending data between neighborhoods until each sensor achieves M measurements. The value of C is changeable and depends on the connections between sensors or the communication Algorithm 1: Random Data Collection Algorithm for Distributed and Collaborative MSNs

Initialization phase:

- $\underline{X} = [x_1 \, x_2 \dots x_N]^T$ represents N unknown values. - $\underline{Y} = [y_1 \, y_2 \dots y_M]^T$ represents CS measurements created at M sensors.

- Appropriate sensing range (R_s) and communication range (R_c) are chosen for all M sensors.

Phase 1: Data collection

- Each sensor collects data while moving through $\boldsymbol{\tau}$ positions

for k = 1 to τ do

- Mobile sensor i^{th} finds node j to visit based on Equation 2.

- The CS measurement y_i is creating by adding new data from node j that mobile sensor i^{th} visited:

$$y_i = y_i + x_j;$$

end

Phase 2: Communications

- The mobile sensors share their own CS measurements with others through their neighbors as

while No new data received do for i = 1 to M do if New data received from any neighbor (j) then - Mobile sensor i^{th} adds new measurement into its own vector \underline{Y} - Mobile sensor i^{th} forwards the new received measurement to its neighbors. - Number of times each sensor sends the data to its neighbors is Counter(i) = Counter(i) + 1; end - Repeat updating new data at each sensor end

- At the end of this phase, each sensor achieves M CS measurements. The convergence time is calculated as C = Counter;

Phase 3: Data recovery

All unknown values (\underline{X}) are reconstructed based on the CS measurements (\underline{Y}) stored at each sensor as $\underline{\hat{X}} = \arg \min \parallel \underline{X} \parallel_1$, subject to $\underline{Y} = \Phi \underline{X}$.

range R_c . M mobile sensors are deployed provide M different neighborhoods and M CS measurements for each sensor. The greater the number of sensors deployed, the more accurate the data reconstruction we can obtain. So, the total energy consumption can be calculated as

$$E_{total} = E_{nei} \times C \times M,\tag{8}$$

where E_{nei} represents the average consumed energy for each sensor to transmit one CS measurement to its neighborhoods that can be calculated as

$$E_{nei} = \omega \times R_c^{\alpha},\tag{9}$$

where ω is the average number of neighbors of each sensor that forms the average number of communications in each neighborhood. R_c is the sensor communication range. α is the path loss exponent that $\alpha = 2$ and $\alpha = 4$ in free space and multipath fading channels, respectively [30]. For simplicity, we assume $\alpha = 2$ throughout this paper. We analyze our problem in both types of network areas: circular and square.

A. Working on circular sensing area

We assume that the mobile sensors are deployed randomly in a circular area with radius R_0 , we can find the average number of sensors deployed in the area covered by each sensor communication range R_c as

$$\beta = \frac{M}{\pi R_0^2} \times \pi R_c^2, \tag{10}$$

where $\frac{M}{\pi R_0^2}$ is the sensor density. As mentioned in [31], we can approximately obtain the mean value of ω as

$$\omega = (\beta - 1) = (M \frac{R_c^2}{R_0^2} - 1). \tag{11}$$

Hence, the total consumed energy as mentioned in Equation 8 for gathering data within M neighborhoods is calculated as

$$E_{total} = (M \frac{R_c^2}{R_0^2} - 1) R_c^2 C M.$$
(12)

B. Working on square sensing area

We assume to have a square sensing area dimensioned $H \times H$. As mentioned in Equation (10), the average number of sensors deployed in the area covered by each sensor communication range R_c as $\beta = \frac{M}{H^2} \times \pi R_c^2$. Hence

$$\omega = \left(\frac{M\pi R_c^2}{H^2} - 1\right).$$
 (13)

Finally, the total energy consumption for sharing M CS measurements in the square sensing area is

$$E_{total} = \left(\frac{M\pi R_c^2}{H^2} - 1\right) R_c^2 CM.$$
 (14)

The convergence time C depends on both the sensor density and the connections between them.

As mentioned in Section III, deploying more or less the sensors results the reconstruction error and also the cost for the network. The communication range R_c decides all the connections. It means that increasing or reducing R_c could increase or reduce the number of connections among the sensors in each neighborhood, respectively. In other words, increasing the connections among sensors reduces the convergence time C. There should be a trade-off at this point.

V. SIMULATION RESULTS

In this section, we consider the square sensing area dimensioned 100×100 (H = 100). We deploy a certain number of mobile sensors randomly in the sensing area. We also assume that there are about 500 positions need to be observed. It means that there are 500 unknown values need to be collected at each mobile sensor to build a scalar map. The real sensor readings we use in our simulations collected from Sensorscope: Sensor Networks for Environmental Monitoring [32].

We chose the random walk length τ from our previous paper [11] as $\tau = 48$ with $R_c = 14$ that provides the lowest energy consumption for the mobile sensors to visit all positions equally. We first check if the measurement matrix achieved from the random walk gathering data works as well as the full dense Gaussian matrix in the CS recovery algorithm. We consider normalized reconstruction error as $\frac{||\underline{X} - \underline{X}||_2}{||\underline{X}||_2}$.



Fig. 2. Comparison between two measurement matrices, the sparse binary matrix obtained from our collecting method and the full dense Gaussian matrix corresponding to full sampling all positions for one CS measurement.

Figure 2 depicts that the sparse binary matrix created by our mobile sensors that randomly sample the sensing area can work as well as the full dense Gaussian matrix in the CS recovery algorithm with different number of CS measurements collected at each mobile sensor.



Fig. 3. The scalar field map with 500 random positions which is reconstructed by 120 CS measurements and the true map. The normalized reconstruction error equals 0.1

The scalar field map is built by the reconstructed readings that are compared with the true readings in the 2-D Figure 3. 500 sensor readings are reconstructed at each mobile sensor out of 120 sensors with the reconstruction error 0.1.

The convergence time as analyzed in Section II-C is calculated with different communication ranges. As analyzed, it reduces almost linearly as the communication range R_c increases.

Figure 5 depicts the total energy consumption for communications between 100 mobile sensors with different communication ranges. As we increase R_c , the gaps between two analysis and simulation lines increase. This can be explained as, when the areas covered by R_c increase as we increase R_c , sensors close by the boundary of the sensing area do not have as many neighbors as the ones close by the center of the sensing area. These sensors lose neighbors since their radio ranges could reach outside the sensing area. In analysis



Fig. 4. The convergence time with different communication ranges.



Fig. 5. Total communication energy consumption with different communication ranges; 100 mobile sensors are deployed in a square sensing area dimensioned 100×100 .

case, we assume that all sensors have the same number of neighbors that would make the difference as we increase R_c . As shown in Figure 5 the optimal communication range R_c^* is the smallest one to minimize the network energy consumption for communications ($R_c^* = 15$).



Fig. 6. Total communication energy consumption with different number of mobile sensors deployed in a square sensing area dimensioned 100×100 ; The convergence time increases corresponding to the number of sensors as follows. C = [74.6 79 84 88.4 93 97.7 102.5 107.4 111.9].

Figure 6 depicts the total communication energy consumption that increases as we increase the number of mobile sensors (L) deploying in the network. Note that the convergence time increases as L increases.

Figure 7 depicts the percentage of sampling the sensing area with 500 random positions need to be observed. It is shown that only 82 mobile sensors with the random walk length as $\tau = 48$ can sample 100% the sensing area.



Fig. 7. Percentage of sampling the random positions with different number of mobile sensors deployed in a square sensing area dimensioned 100×100 .

VI. CONCLUSIONS

In the paper, we proposed an algorithm that deploys a certain number of distributed mobile sensors to build scalar field maps utilizing compressive sensing. Each mobile sensor moves or walks in a sensing area to visit some random positions to create CS measurements. The measurements created at each mobile are shared with others through sensor neighborhoods. After each mobile sensor obtains all the CS measurements required, it implements the CS recovery algorithm to reconstruct all sensor readings from the sensing area and build the scalar map itself. All the energy consumption for communications in the network are analyzed and formulated. The trade-off between three factors (R_c , C and L) is analyzed. The possible smallest communication range is suggested for the network to consume the least energy in order to increase the lifetime.

REFERENCES

- B. Liu, O. Dousse, P. Nain, and D. Towsley, "Dynamic coverage of mobile sensor networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 24, pp. 301–311, Feb 2013.
- [2] C. Tunca, S. Isik, M. Donmez, and C. Ersoy, "Distributed mobile sink routing for wireless sensor networks: A survey," *Communications Surveys Tutorials, IEEE*, vol. 16, pp. 877–897, Second 2014.
- [3] H. M. La and W. Sheng, "Distributed sensor fusion for scalar field mapping using mobile sensor networks," *Cybernetics, IEEE Transactions on*, vol. 43, pp. 766–778, April 2013.
- [4] H. La, W. Sheng, and J. Chen, "Cooperative and active sensing in mobile sensor networks for scalar field mapping," *Systems, Man, and Cybernetics: Systems, IEEE Transactions on*, vol. PP, pp. 1–1, May 2014.
- [5] D.L.Donoho, "Compressed sensing," Information Theory, IEEE Transactions on, vol. 52, pp. 1289 – 1306, 2006.
- [6] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *Information Theory, IEEE Transactions on*, vol. 52, pp. 489 – 509, Feb. 2006.
- [7] R. Baraniuk, "Compressive sensing [lecture notes]," Signal Processing Magazine, IEEE, vol. 24, pp. 118 –121, July 2007.
- [8] E. Candes and M. Wakin, "An introduction to compressive sampling," Signal Processing Magazine, IEEE, vol. 25, no. 2, pp. 21–30, 2008.
- [9] J. Haupt, W. Bajwa, M. Rabbat, and R. Nowak, "Compressed sensing for networked data," *Signal Processing Magazine*, *IEEE*, vol. 25, no. 2, pp. 92–101, 2008.
- [10] G. Quer, R. Masiero, G. Pillonetto, M. Rossi, and M. Zorzi, "Sensing, compression, and recovery for wsns: Sparse signal modeling and monitoring framework," *Wireless Communications, IEEE Transactions* on, vol. 11, pp. 3447–3461, October 2012.

- [11] M. T. Nguyen, "Minimizing energy consumption in random walk routing for wireless sensor networks utilizing compressed sensing," in System of Systems Engineering (SoSE), 2013 8th International Conference on, pp. 297–301, June 2013.
- [12] M. T. Nguyen and K. Teague, "Compressive sensing based energyefficient random routing in wireless sensor networks," in Advanced Technologies for Communications (ATC), 2014 International Conference on, pp. 187–192, Oct 2014.
- [13] M. T. Nguyen and N. Rahnavard, "Cluster-based energy-efficient data collection in wireless sensor networks utilizing compressive sensing," in *Military Communications Conference, MILCOM 2013 - 2013 IEEE*, pp. 1708–1713, Nov 2013.
- [14] M. T. Nguyen and K. Teague, "Compressive sensing based data gathering in clustered wireless sensor networks," in *Distributed Computing* in Sensor Systems (DCOSS), 2014 IEEE International Conference on, pp. 187–192, May 2014.
- [15] R. Xie and X. Jia, "Minimum transmission data gathering trees for compressive sensing in wireless sensor networks," in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pp. 1–5, Dec 2011.
- [16] M. T. Nguyen and K. Teague, "Tree-based energy-efficient data gathering in wireless sensor networks deploying compressive sensing," in *Wireless and Optical Communication Conference (WOCC)*, 2014 23rd, pp. 1–6, May 2014.
- [17] L. Guo, R. Beyah, and Y. Li, "Smite: A stochastic compressive data collection protocol for mobile wireless sensor networks," in *INFOCOM*, 2011 Proceedings IEEE, pp. 1611–1619, April 2011.
- [18] H. Wang, Y. Zhu, and Q. Zhang, "Compressive sensing based monitoring with vehicular networks," in *INFOCOM*, 2013 Proceedings IEEE, pp. 2823–2831, April 2013.
- [19] Y. Mostofi, "Compressive cooperative sensing and mapping in mobile networks," *Mobile Computing, IEEE Transactions on*, vol. 10, pp. 1769– 1784, Dec 2011.
- [20] Y. Mostofi, "Cooperative wireless-based obstacle/object mapping and see-through capabilities in robotic networks," *Mobile Computing, IEEE Transactions on*, vol. 12, pp. 817–829, May 2013.
- [21] W. Ren, R. Beard, and E. Atkins, "A survey of consensus problems in multi-agent coordination," in *American Control Conference*, 2005. *Proceedings of the 2005*, pp. 1859–1864 vol. 3, June 2005.
- [22] E. Kokiopoulou and P. Frossard, "Distributed classification of multiple observation sets by consensus," *Signal Processing, IEEE Transactions* on, vol. 59, pp. 104–114, Jan 2011.
- [23] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. on Automatic Control*, vol. 51, no. 3, pp. 401– 420, 2006.
- [24] R. Berinde and P. Indyk, "Sparse recovery using sparse random matrices," MIT-CSAIL Technical Report, 2008.
- [25] L. Lovsz, "Random walks on graphs: A survey," 1993.
- [26] S. Boyd, P. Diaconis, and L. Xiao, "Fastest mixing markov chain on a graph," *SIAM REVIEW*, vol. 46, pp. 667–689, 2003.
- [27] C. Avin and G. Ercal, "On the cover time and mixing time of random geometric graphs," *Theoretical Computer Science*, vol. 380, no. 12, pp. 2 – 22, 2007. Automata, Languages and Programming.
- [28] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Mixing times for random walks on geometric random graphs," in *the proceedings of SIAM ANALCO*, pp. 240–249, SIAM, 2005.
- [29] D. Vukobratovic, C. Stefanovic, V. Crnojevic, F. Chiti, and R. Fantacci, "A packet-centric approach to distributed rateless coding in wireless sensor networks," in *Sensor, Mesh and Ad Hoc Communications and Networks, 2009. SECON '09. 6th Annual IEEE Communications Society Conference on*, pp. 1–8, June 2009.
- [30] T. S. Rappaport, Wireless Communications: Principles and Practice (2nd Edition). Prentice Hall, 2 ed., Jan. 2002.
- [31] M. T. Nguyen and K. Teague, "Neighborhood based data collection in wireless sensor networks employing compressive sensing," in Advanced Technologies for Communications (ATC), 2014 International Conference on, pp. 198–203, Oct 2014.
- [32] http://lcav.epfl.ch/op/edit/sensorscope en.