This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE Globecom 2011 proceedings.

Minimum Transmission Data Gathering Trees for Compressive Sensing in Wireless Sensor Networks

Ruitao Xie*, Xiaohua Jia†
Department of Computer Science
City University of Hong Kong
*ruitaoxie2@student.cityu.edu.hk
†jia@cs.cityu.edu.hk

***Abstract*—Compressive sensing (CS) can reduce the number of data transmissions and balance the traffic load throughout networks. However, the total number of data transmissions required in CS method is still large. It is observed that there are many zero elements in the measurement matrix. In each round of data transmission in CS method, the sensor nodes corresponding to the zero elements in the measurement matrix do not have their own data to transmit. To further reduce the number of data transmissions in the network, we aim to compute a data gathering tree by taking advantages of these zero elements in the measurement matrix, such that the total number of data transmissions is minimized. We formulate the problem as linear programming with boolean variables. The problem is NP-hard. We propose heuristic algorithm to compute the Minimum Transmission Tree (MTT) for data gathering in CS methods. The MTT algorithm constructs a spanning tree by iteratively including the edge whose average incremental transmission cost is minimum. The simulation results demonstrate that our algorithm can reduce the number of transmissions significantly, compared with the methods using minimum spanning tree (MST), shortest path tree and the CS method with nonzero measurement coefficient using MST.**

Keywords: compressive sensing, data gathering, wireless sensor networks

I. INTRODUCTION

Data gathering is a basic function of wireless sensor networks in various applications [1], [2]. The emerging technology of compressive sensing (CS for short) [3]–[5] opens a new frontier for data gathering in sensor networks [6]. CS method can substantially reduce the amount of data transmissions and balance the traffic load throughout the network.

The basic idea of CS works as follows. Suppose the system consists of one sink node and n sensor nodes for collecting data from the field. Let x denote a vector of original data collected from sensors. Vector x has n elements, one for each sensor. In CS, x can be represented by Ψs, i.e., $x = \Psi s$, where Ψ is a $n \times n$ transform basis, and s is a vector of coefficients. If there are at most k nonzero elements in s, x is called k-sparse. When k is small, instead of transmitting n data to the sink, we can send a small number of projections of x to the sink, that is, $y = \Phi x$, where Φ is an $m \times n$ random matrix and y is a vector of m projections. At the sink node, after collecting y, the original data x can be recovered. Some recent research has been done on how to apply CS theory to data gathering in sensor networks, such as [7]–[9].

In data gathering by using CS, each node has to transmit m data projections to the sink, as shown in Fig. 1. The total number of transmissions for collecting n data items is mn, which is still a large number. It is observed that there are many zero elements in matrix Φ. In each round of projections, the sensor nodes corresponding to these zero elements do not have their own data to transmit. To further reduce the number of data transmissions, we need to generate a routing tree that spans only the nodes that participate in this round of data transmission. Literally, for each round of transmission, we need to use a different routing tree that minimizes the data transmissions in this round. However, this would result in the use of m trees for m rounds of data transmissions. By using too many trees, it incurs high overhead for each node to keep and maintain the trees, and it also brings difficulties for synchronization of data transmission in each round. Therefore, the better idea is to construct one data gathering tree and this tree minimizes the total number of data transmissions.

Our problem is NP-hard. We propose a heuristic algorithm to compute Minimum Transmission Tree (MTT for short). The MTT method iteratively includes the edge whose average incremental transmission cost is minimum to the tree. Extensive simulations have been done to evaluate the performance of MTT against the CS methods using shortest path tree (SPT), minimum spanning tree (MST), and the CS method without zero measurement coefficients using MST. The results show that MTT can reduce the number of transmissions significantly.

The rest of the paper is organized as follows. Section II presents the related work on the data gathering of CS. The system model and problem formulation is presented in Section III. Section IV focuses on the discussion of MTT algorithm and its analysis. Section V presents the simulation results, followed by the conclusion in Section VI.

II. DATA GATHERING TREES FOR COMPRESSIVE SENSING

The new technology of CS [3]–[5] motivates the investigations on data gathering with CS [6], [7]. The data gathering with CS in the two dimensional area is conducted along the tree structure in Fig. 1. In the i^{th} round of projection, each node generates a random measurement coefficient ϕ_{ij}, and computes the data term $\phi_{ij} x_j$ for node v_j. Each leaf node transmits its term to the parent node. Once the parent node receives data from all its descendant nodes, it can add its own

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE Globecom 2011 proceedings.

data term and all received data terms together and then send it to its upper parent node or the sink node. When the sink node receives data from all its descendant nodes, it adds them to form the i^{th} projection. When the sink nodes collect m projections in the above way, it can use ℓ_1-norm minimization [10], [11] to recover the n original data.

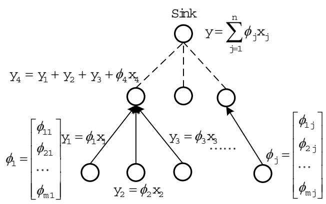


Fig. 1. Data gathering with CS in the tree structure.

Authors in [7] analyzed the network capacity when CS is utilized in data gathering and proved that the capacity gain is proportional to the sparsity level of sensor data. Although researchers stated that the total transmission number can be reduced when the number of projections is low enough. However, if the required number of projections increases, then the total number of transmissions may be larger than the case without CS.

Authors in [12] researched how large throughput can be achieved with or without CS, or with the hybrid scheme, where CS is only applied in the nodes whose incoming traffic is beyond the threshold, the traffic when CS is applied. The authors stated that application of CS in the simple scheme may bring no obvious throughput improvement, but application of CS in hybrid scheme can achieve significant throughput improvement. Authors in [8] designed the data gathering scheme, where in each round of projection m furthest nodes away from the sink send their original data directly to one of the remaining nodes which apply the CS. They proved that this scheme can reduce the transmission cost.

The previous researchers considered the number of transmissions, but they missed the fact that measurement matrix Φ is possible to have many zero elements, such as the one made from the following distribution [5], [13]:

$$\phi_{ij} = \sqrt{3} \times \begin{cases} +1 & \text{with probability} \quad 1/6 \\ 0 & \quad\cdots \quad\quad\quad 2/3 \\ -1 & \quad\cdots \quad\quad\quad 1/6 \end{cases} . \tag{1}$$

They did not consider to save the transmission cost by building an efficient data gathering structure. In this paper, we aim at computing an minimum transmission tree for CS.

III. System Model and Problem Formulation

The system is modeled by a graph $G =< V, E >$, where V consists of the sink node v_0 and n sensor nodes. If node v_i and node v_j in V are within the communication range of each other, then there is an edge between v_i and v_j, i.e. $e = (v_i, v_j) \in E$. Without loss of generality, in system model, we assume all edges have the reliability as 1, and the cost of each edge for one transmission is 1.

Our task is to generate a data gathering tree T that is used for transmissions of all m projections. Given a measurement matrix Φ, when there are many zero elements in Φ, each round of projections only involves a subset of nodes in the network. Let T_i denote the subtree of T which is used for the transmission of i^{th} round of projection, and $L(T_i)$ denote the set of leaf-nodes in T_i. The nodes in $L(T_i)$ correspond to nonzero elements in i^{th} row of Φ, that is

$$\phi_{ij} \neq 0, \quad \forall v_j \in L(T_i), i = 1, 2, \cdots, m.$$

Denote

$$x_e^i = \begin{cases} 1 & e \in T_i \\ 0 & e \notin T_i \end{cases}, \quad \text{and} \quad x_e = \begin{cases} 1 & e \in T \\ 0 & e \notin T \end{cases}.$$

The cost of each edge for m rounds of transmission is defined as

$$c_e = \sum_{i=1}^{m} x_e^i.$$

The transmission cost of data gathering tree T for m projections is $\sum_{e \in T} c_e$, which is sum of the cost of all subtrees:

$$\sum_{e \in T} c_e = \sum_{e \in T} \sum_{i=1}^{m} x_e^i = \sum_{i=1}^{m} \sum_{e \in T_i} x_e^i = \sum_{i=1}^{m} C(T_i),$$

where $C(T_i)$ is the cost of T_i.

The connectivity in the tree T between the sink node v_0 and the other nodes in V can be expressed as follows. Given any $v_0 - v$ cut Q, one of the edges in the crossing set $\delta(Q)$ must be in T. Thus, we express the connectivity as

$$\sum_{e \in \delta(Q)} x_e \geq 1, \quad Q \in F,$$

where $\delta(Q) = \{(u, w) \in E : |\{u, w\} \cap Q| = 1\}$, and F is the set of cuts between the sink and the sensor nodes,

$$F = \{Q : |Q \cap \{v_0, v\}| = 1, v \in V \backslash \{v_0\}\}.$$

Since all nodes in the graph participate in the transmission of projections, our problem is to construct data gathering tree which spans all the sensor nodes and minimizes the total transmission cost. The problem can be formulated by linear programming with boolean variables as follows.

$$\min \sum_{e \in T} c_e$$

subject to:

$$x_e \in \{0, 1\}, \tag{2}$$

$$\sum_{e \in \delta(Q)} x_e \geq 1, \quad Q \in F, \tag{3}$$

$$c_e = \sum_{i=1}^{m} x_e^i. \tag{4}$$

$$\phi_{ij} \neq 0, \quad \forall v_j \in L(T_i), i = 1, \cdots, m. \tag{5}$$

Eq. (2) presents the boolean property of the variable x_e and Eq. (3) states the connectivity between the sink and sensor nodes. Eq. (4) calculates the cost of each edge for m rounds of transmission. Eq. (5) presents the leaf-nodes of subtree T_i correspond to the nonzero elements in the i^{th} row of Φ, where T_i is used for the transmission of i^{th} round of projections. This problem is NP-hard. Therefore, we propose the heuristic algorithm MTT in the next section to solve it.

IV. Heuristic Algorithm

Given $G =< V, E >$ and the measurement matrix Φ, the MTT algorithm iteratively includes one edge into the tree and finally computes a spanning tree T. Let T' denote the subtree constructed in each iteration. Let V' denote the set of nodes in the subtree T', and U' denote the set of nodes unconnected to T', that is $U' = V \backslash V'$. In each iteration, we choose one edge which connects a node in U' to T'.

Our algorithm starts from the sink node, initially T' contains only the sink node, at each iteration we evaluate all edges that connect nodes in unconnected set U' to T'. The adding of new edge would induce the increment of transmission cost. We choose the edge whose average incremental cost is minimum among all edges, and add it to the tree to minimize the transmission cost of data gathering tree.

Given the subtree T' constructed so far, we discuss how to calculate the incremental cost of edges in the following part. Since sensor nodes in the subtree participate in different rounds of transmission, we calculate the incremental cost of edge by adding the incremental cost for each round. Let $C_i^+(T', e)$ denote the incremental cost of subtree T' if edge e is added into T' for i^{th} round of transmission, where $e = (u, v)$ connects the node u in the set U' to the node v in the subtree T'. Let $C^+(T', e)$ denote the incremental cost for all m rounds of transmission. The incremental cost of edge e can be calculated by adding $C_i^+(T', e)$ for all rounds in which node u has own data to transmit, that is

$$C^+(T', e) = \sum_{i \in R(u)} C_i^+(T', e),$$

where $R(u)$ is the set of rounds in which node u has own data to transmit.

The data gathering tree is used to collect all m projections. For each round of projection, subset of nodes in the subtree T' participate in transmission, either have own data to transmit or have received data to relay, the data of new node can be relayed by this subset of nodes without extra cost. While another subset of nodes, that neither have own data to transmit nor have received data to relay, do not participate in this round of transmission. When these nodes relay the data of new node, the extra cost is generated. To calculate the incremental cost, it is needed to get the subset of nodes that do not participate in certain round of transmission. We search the first node that participates in a round of transmission among the upstream nodes of new node, then the downstream nodes of target node have no transmission. The incremental cost can be calculated by it.

As Fig. 2 shows, given the subtree T' constructed so far, we calculate the incremental cost of edge $e = (u, v)$ for i^{th}

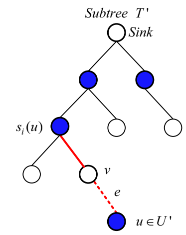


Fig. 2. Calculating incremental cost of edge for one round of transmission.

round of projection. The solid nodes participate in i^{th} round of transmission. Let $s_i(u)$ denote the first node that participates in this round of transmission, along the path from node v to the sink node. All upstream nodes of node $s_i(u)$ participate in this round, while the downstream nodes of node $s_i(u)$ do not participate in. To relay the data of node u, the downstream nodes of node $s_i(u)$ generate the transmission cost. Thus, the incremental cost $C_i^+(T', e)$ of edge e for i^{th} round is the cost of links which are related to this subset of nodes, that is

$$C_i^+(T', e) = C(v, s_i(u)) + 1,$$

where $C(v, s_i(u))$ is the cost of links between node v and node $s_i(u)$ in subtree T'.

The data gathering tree constructed in our algorithm is used to transmit all m projections, while the node u added into the subtree T' only participates in several rounds of projection. Each node u transmits different number of rounds, that is $|R(u)|$ is different. The larger number of rounds means that the incremental cost of edges would be smaller in the following iterations. Therefore, we consider the average incremental cost of each edge, that is

$$\bar{C}^+(T', e) = \sum_{i \in R(u)} C_i^+(T', e)/|R(u)|,$$

where $|R(u)|$ presents the number of rounds in which node u transmits its own data. In each iteration, we choose the edge whose average incremental cost is minimum to minimize the total transmission cost.

As shown in Algorithm 1, the MTT algorithm iterates n times to build a spanning tree T. Initially, T' contains only the sink node. At each iteration, we calculate the incremental cost and average incremental cost of all edges that connects nodes in unconnected set U' to T'. We choose the edge e^* whose average incremental cost is minimum among all the edges, and add e^* into T'. It is necessary to update the related variables. The above process is repeated until all sensor nodes are in T'. The final T' is the data gathering tree T with the root of v_0 and connecting the nodes in V.

To calculate the incremental cost of edges, it is necessary to know the subset of nodes that have no transmission in a round. We can get that each node u has own data to transmit in rounds $R(u)$ from Φ at the beginning of algorithm. However, it is possible that the node has no own data to transmit but has received data to relay in a set of rounds. This set changes after each iteration in the MTT algorithm. Since after adding

Algorithm 1 Heuristic MTT Algorithm

Input: $G =< V, E >$ and Φ;

Initialization:
$T' = \{v_0\}$, $V' = \{v_0\}$, $U' = V \backslash \{v_0\}$;
Get $R(u)$ from Φ, for $u \in U'$;
while $U' \neq \emptyset$ **do**
  for $e = (u, v) \in E$, where $u \in U'$, $v \in V'$ **do**
    Calculate incremental cost $C^+(T', e)$;
    Calculate average incremental cost $\bar{C}^+(T', e)$;
  end for
  Choose the edge e^*, whose average incremental cost $\bar{C}^+(T', e)$ is minimum, and add the edge e^* into T';
  Update the related variables U', V';
end while

Output: Data gathering tree $T = T'$ with the root of v_0 and connecting the nodes in V.

new edge $e^* = (u^*, v^*)$ into the tree T', when node u^* has own data to transmit in certain rounds of projection, the nodes along the routing path have to relay these data. Therefore, it is required to record the rounds in which each node relays data after each iteration.

Given the subtree T' constructed so far, the cost of new tree $T' \cup \{e^*\}$ is the sum of incremental cost and the cost of subtree T', that is,

$$C(T' \cup \{e^*\}) = C(T') + C^+(T', e^*),$$

where $C(T')$ is the cost of subtree T'. The cost of data gathering tree T can be calculated by adding the incremental cost of all edges. The following theorem proves this cost is equal to the definition in section III.

Theorem I. The sum of incremental cost $\sum_{e \in T} C^+(T', e)$ is equal to the cost $\sum_{i=1}^{m} c_e$ of tree T, where c_e is the cost of each edge for m rounds of transmission.

Proof. Given $e = (u, v)$, according to the calculation method of incremental cost, we have

$$\begin{aligned} & \sum_{e \in T} C^+(T', e) \\ = & \sum_{e \in T} \sum_{i \in R(u)} (C(v, s_i(u)) + 1). \end{aligned} \tag{6}$$

After adding the edge $e = (u, v)$ into the tree, we have $C(v, s_i(u)) + 1 = C(u, s_i(u))$. Furthermore, $i \in R(u)$ is equivalent to $u \in V_i$, where V_i is the set of nodes which involves in the i^{th} round of data transmission. Thus, Eq. (6) is equal to

$$\sum_{i=1}^{m} \sum_{u \in V_i} C(u, s_i(u)). \tag{7}$$

When evaluating edges in our algorithm, $s_i(u)$ is the first node that participates in the i^{th} round of transmission among all relay nodes of node u. For i^{th} round of transmission, when the nodes in V_i are connected to the tree, the cost of links that have no transmission previously, that is $C(u, s_i(u))$, adds up to the cost of subtree T_i, which is used to transmit the i^{th} round of projection. Therefore, the i^{th} term in Eq. (7) is the transmission cost of subtree T_i. Thus, the sum of incremental cost is equal to the sum of the cost of all subtrees.

$$\sum_{e \in T} C^+(T', e) = \sum_{i=1}^{m} C(T_i).$$

Since we get that the cost of tree T is the sum of the cost of all subtrees in section III. It can be induced that the sum of incremental cost is equal to the cost of tree T. That is,

$$\sum_{e \in T} C^+(T', e) = \sum_{i=1}^{m} c_e.$$

V. Simulations

In our simulations, we set the sparsity $k = 0.05n$ and the number of projection $m = 4k = n/5$. Given the $m \times n$ measurement matrix Φ generated as follows,

$$\phi_{ij} = \begin{cases} 1 & \text{with probability} \quad 1/3 \\ 0 & \qquad \cdots \qquad\quad 2/3 \end{cases},$$

we build the data gathering tree for m rounds of transmission in networks with different scale.

We simulate on the lattice topology and the arbitrary topology. The lattice topology contains the same rows and columns, and any nodes not at the border of the network only can communicate with four neighbors. The sink node locates at the center point of the network. In arbitrary topology the nodes are deployed uniformly in a fixed area with sink at the center.

In our simulations, we use the number of transmissions to evaluate the performance of the CS method using MTT, compared with the CS method using MST, SPT and also the CS method with nonzero measurement coefficient using MST (MST-CS for short).

A. Reliable networks

We first consider all links in network have the reliability of 1. Fig. 3 shows the number of transmissions using different data gathering methods in the lattice topology. It is shown that MTT has the smallest number of transmissions, while MST-CS has the largest number of transmissions. Fig. 4 shows the reduction ratio of transmission number of MTT over other methods. The reduction ratio in the case of $m = n/2$ is similar to that in the case of $m = n/5$. It demonstrates that performance improvement of MTT algorithm over MST and SPT is insensitive to the number of projections m, although more transmissions are needed.

In Fig. 4(a), the reduction ratio of MTT over MST and SPT increases when network consists of more nodes. The reduction ratio of MTT over SPT is up to 28% in the lattice network of 400 nodes, and the reduction ratio of MTT over MST is up to 38% in the lattice network of 400 nodes.

It is noticed that when the $m \times (n/3)$ measurement matrix with nonzero element is used in CS method, the number of transmissions is $mn/3$ in reliable networks, and it is 67% less than that of MST-CS with $m \times n$ Φ. Since the measurement

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE Globecom 2011 proceedings.

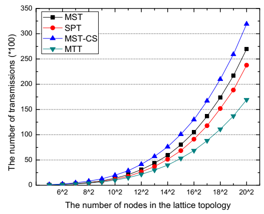
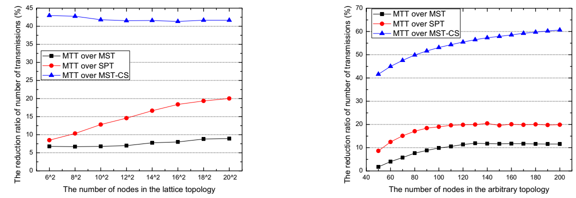
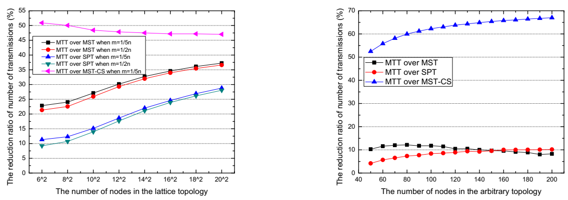


Fig. 3. The number of transmissions in the lattice topology



(a) lattice topology (b) arbitrary topology

Fig. 5. The reduction ratio of transmission number in unreliable networks



(a) lattice topology (b) arbitrary topology

Fig. 4. The reduction ratio of transmission number in reliable networks

matrix used in our simulations has 1/3 nonzero elements at random, the reduction ratio of transmission cost of MTT over MST-CS is always lower than 67%. In Fig. 4(a), the reduction ratio of MTT over MST-CS decreases little when the network consists of more nodes. It is about 47% in the network of 400 nodes, which is less than 67%. It demonstrates that MTT algorithm has significant reduction of the transmission number.

Fig. 4(b) presents in arbitrary topology the largest reduction ratio of MTT over the better one between MST and SPT is about 10%. However, the reduction ratio of MTT over MST-CS is up to 67%. According to the above analysis, it demonstrates that MTT algorithm is near to optimal solutions in this situation.

B. Unreliable networks

The algorithm in this paper can be easily extended to the unreliable networks. In our simulations, the reliability of links is chosen uniformly at random from $\{1, 1/2, 1/3\}$. Fig. 5(a) presents that the reduction ratio of MTT in unreliable networks of lattice topology is smaller than that in the reliable networks. The reduction ratio of MTT over SPT is up to 20% in unreliable networks, while it is up to 28% in reliable networks; the reduction ratio of MTT over MST in unreliable networks is about 10%, while it is 38% in reliable networks. The reduction ratio of MTT over MST-CS in unreliable networks is 5% less than that in reliable networks, and it is still above 40%.

Fig. 5(b) shows that in arbitrary networks when the network size is larger than 100 nodes, the reduction ratio of MTT over MST is above 10% and that over SPT is above 20%. The reduction ratio of MTT over MST-CS in unreliable networks is about 10% less than that in reliable networks. The largest reduction ratio is up to 60% in the network of 200 nodes, which is near to the optimal value 67% in reliable networks.

VI. Conclusion

In this paper, we reduce the number of transmissions in data gathering using CS by building the minimum transmission data gathering tree. It should be noted that we consider the scenario where the measurement matrix has many zero elements. We first formulate the problem into linear programming with boolean variables. We further develop the heuristic algorithm MTT, which builds a spanning tree by iteratively including the edge whose average incremental cost is minimum. Simulation results demonstrate that MTT can reduce the number of transmissions significantly compared with the methods using MST, SPT and also the CS method with nonzero measurement coefficient using MST.

References

[1] H. Liu, X. Jia, P.-J. Wan, C.-W. Yi, S. Makki, and N. Pissinou, "Maximizing lifetime of sensor surveillance systems," *Networking, IEEE/ACM Transactions on*, vol. 15, no. 2, pp. 334–345, April 2007.

[2] I. Stojmenovic, D. Liu, and X. Jia, "A scalable quorum-based location service in ad hoc and sensor networks," *Int'l Journal of Communication Networks and Distributed Systems*, vol. 1, no. 1, pp. 71–94, 2008.

[3] D. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[4] R. Baraniuk, "Compressed sensing," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, Jul. 2007.

[5] E. Candes and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[6] J. Haupt, W. Bajwa, M. Rabbat, and R. Nowak, "Compressed sensing for networked data," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 92–101, Mar. 2008.

[7] C. Luo, F. Wu, J. Sun, and C. W. Chen, "Compressive data gathering for large-scale wireless sensor networks," in *Proc. ACM MobiCom'09*, Sep. 2009, pp. 145–156.

[8] ——, "Efficient measurement generation and pervasive sparsity for compressive data gathering," *Wireless Communications, IEEE Transactions on*, vol. 9, no. 12, pp. 3728–3738, Dec. 2010.

[9] S. Lee, S. Pattem, M. Sathiamoorthy, B. Krishnamachari, and A. Ortega, "Spatially-localized compressed sensing and routing in multi-hop sensor networks," in *GSN '09*, 2009, pp. 11–20.

[10] 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, no. 2, pp. 489–509, Feb. 2006.

[11] E. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *Information Theory, IEEE Transactions on*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[12] J. Luo, L. Xiang, and C. Rosenberg, "Does compressed sensing improve the throughput of wireless sensor networks?" in *ICC 2010*, May 2010, pp. 1–6.

[13] D. Achlioptas, "Database-friendly random projections: Johnson-lindenstrauss with binary coins," *Journal of Computer and System Sciences*, vol. 66, pp. 671–687, Jun. 2003.