An Efficient Data Transmission Methodology using Compressive Sensing and Anchor Node in WSN

Shalini U R
M.tech, 4th sem ,Dept. of CSE
T.John institute of technology
Bangalore, India

Nagashree K T
Assistant Professor, Dept. of CSE
T.John institute of technology
Bangalore, India

Abstract**—In wireless sensor network, the sensor node collects the data and forwards it towards the sink. While transferring the data towards the sink, the energy level starts deprecated. To reduce number of transmission, Compressive sensing (CS) is used because, it provide better throughput by balancing the traffic load throughput networks. The number of data transmission to collect the data using pure CS is still large hence, the Hybrid method is proposed. This hybrid method of using CS will reduce the number of transmissions in sensor networks. The sensor field is divided into the number of cluster. The clusters are formed based on the network area. The cluster head (CH) is declared based on the transmission range and the energy level. Sensor nodes transmit data to cluster head without using CS within a cluster. CHs use CS to transmit data to sink .The anchor nodes are deployed in the network which are moving in the network area and equipped with GPS device. Anchors broadcast its current location periodically and the broadcasted messages are received by the sensor nodes. After receiving three messages, a sensor can find the location of the sensor node. While sending the collected information to the sink, a node first send the data towards the boundary node of the next cluster and like this data is forwarded to the sink. Extensive experiments confirm that our method can reduce the number of transmissions significantly.**
.

***Keywords—Wireless sensor networks, compressive sensing, data collection, clustering*.**

I. INTRODUCTION

IN many sensor network applications, such as environment monitoring systems, sensor nodes need to collect data periodically and transmit them to the data sink through multihops. According to field experiments, data communication contributes majority of energy consumption of sensor nodes [1]. It has become an important issue to reduce the amount of data transmissions in sensor networks.

The emerging technology of compressive sensing (CS) [2], [3], [4] opens new frontiers for data collection in sensor networks [5], [6], [7], [8], [9], [10] and target localization in sensor networks. The CS method can substantially reduce the amount of data transmissions and balance the traffic load throughout the entire network. The basic idea of CS works is as follows, as shown in Fig. 1. 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.

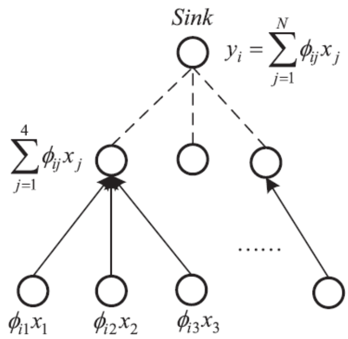

Fig. 1. Data collection with the pure CS method in the tree structure.

In data gathering without using CS, the nodes close to tree leaves relay fewer packets for other nodes, but the nodes close to the sink have to relay much more packets. By using CS in data gathering, every node needs to transmit M packets for a set of N data items. That is, the number of transmissions for collecting data from N nodes is MN, which is still a large number. Hybrid approaches were proposed in [8], [10]. In the hybrid method, the nodes close to the leaf nodes transmit the original data without using the CS method, but the nodes close to the sink transmit data to sink by the CS technique. A hybrid CS in the data collection and proposed an aggregation tree with minimum energy consumption. The previous works use the CS method on routing trees. Since the clustering method has many advantages over the tree method such as fault tolerance and traffic load balancing, we use the CS method on the clustering in sensor networks. The clustering method generally has better traffic load balancing than the tree data gathering method. This is because the number of nodes in clusters can be balanced when we divide clusters. In addition, the previous works ignored the geographic locations and node distribution of the sensor nodes. While in sensor networks, the information of node distribution can help the design of data gathering method that uses less data transmissions. In this paper, we propose a clustering method that uses the hybrid CS for sensor networks. The sensor nodes are organized into clusters. Within a cluster, nodes transmit data to the cluster head (CH) without using CS. A data gathering tree spanning all CHs is constructed to transmit data to the sink by using the CS method. One important issue for the hybrid method is to determine how big a cluster should be. If the cluster size is too big, the number of transmissions required to collect data from sensor nodes within a cluster to the CH will be very high. But if the cluster size is too small, the number of clusters will be large and the data gathering tree for all CHs to transmit their collected data to the sink will be large, which would lead to a large number of transmissions by using the CS method. In this regard, we first propose an analytical model that studies the relationship between the size of clusters and number of transmissions in the hybrid CS method, aiming at finding the optimal size of clusters that can lead to minimum number of transmissions. Then, we propose a centralized clustering algorithm based on the results obtained from the analytical model. Finally, we present a distributed implementation of the clustering method.

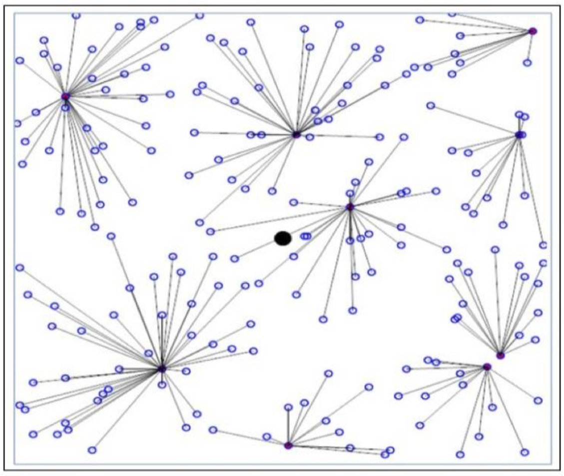

Fig 2. Sensor field divided into cluster.

In existing system each and every sensor node is having attached GPS which will increase the cost of entire system. Here CH selection is not based on energy level, so there is a chance of losing connection after a period of time. Also while sending the information back to the sink is using a path from one CH to other CH. Again this will raise the network traffic.We proposed anchor nodes, which are deployed in the network that are moving in the network area and equipped with GPS device. Anchors broadcast its current location periodically and the broadcasted messages are received by the sensor nodes. After receiving three messages, a sensor can find the location of the sensor node. Here CH selection is done based on Energy level. While sending the collected information to the sink, a node first sends the data towards the boundary node of the next cluster and like this data is forwarded to the sink.

The remainder of this paper is organized as follows: Section2 presents an overview about the related work.Section3 presents how the sensor field is formed into the cluster and the compression of data by using hybrid compressive sensing. Section4 presents overview of centralizes clustering algorithm and finding the cluster center. Section5 presents CH selection and path selection algorithm. The section6 presents the system modules, which contain sink, cluster head and sensor nodes.

II. RELEATED WORK

In paper [2] article considers a particularly salient aspect of network science that revolves around large-scale distributed sources of data and their storage, transmission, and retrieval. The task of transmitting information from one point to another is a common and well-understood exercise. But the problem of efficiently sharing information from and among a vast number of distributed nodes remains a great challenge, primarily because we do not yet have well developed theories and tools for distributed signal processing, communications, and information theory in large-scale networked systems. Compressed sensing provides two key features, universal sampling and decentralized encoding, making it a promising new paradigm for networked data analysis. The paper [3] presents the first complete design to apply compressive sampling theory to sensor data gathering for large scale wireless sensor networks. The proposed compressive data gathering is able to reduce global scale communication cost without introducing intensive computation or complicated transmission control. The proposed scheme can cope with abnormal sensor readings gracefully. A novel scheme for energy efficient data gathering in large scale wireless sensor networks. In paper [5] they proposed two different ways (plain-CS and hybrid-CS) of applying CS to WSNs at the networking layer, in the form of a particular data aggregation mechanism. We formulate three flow-based optimization problems to compute the throughput of the non-CS, plain-CS, and hybrid-CS schemes. Described a naive way of applying CS called plain-CS, then we propose a hybrid-CS scheme that combines conventional data collection (non-CS) with plain-CS. We formulate and solve three flow-based optimization problems that characterize the throughput under the three schemes. Paper [6] present an adaptive data gathering scheme by compressive sensing for wireless sensor networks. By introducing autoregressive (AR) model into the reconstruction of the sensed data, the local correlation in sensed data is exploited and thus local adaptive sparsity is achieved. The paper [10] focuses on nodal and network performance, with an emphasis on lifetime, reliability, and the static and dynamic aspects of single and multi-hop networks. They evaluate the physical design of the sensor node based on deployment experience and a post mortem analysis.

III. SENSOR NODES CLUSTERING FOR HYBRID COMPRESSIVE SENSING

We first make the following assumptions: The sensor nodes are uniformly and independently distributed in a sensor field. Such a deployment can be modeled as a Poisson point process. Each sensor node is aware of its own geographic location, which can be obtained by anchor nodes. The location information is used in the distributed implementation. In our method, sensor nodes are organized into clusters, and each cluster has a cluster head, represented by the solid square as shown in Fig. 2. Sensor nodes in each cluster transmit their original data to the CH without using CS. We assume each CH knows the projection vectors (in measurement matrix) of all nodes within its cluster. In real systems, the measurement coefficient Φ_{ij} can be generated using a pseudorandom number generator seeded with the identifier of the node V_j. Thus, given the identifiers of the nodes in the network, the measurement matrix can be easily constructed at CHs or the sink locally. The measurement matrix can be decomposed into submatrices, one for each cluster. Let Hi denote the submatrix for *i*th cluster. For *i*th cluster, let CH_i denote the cluster head and x^{Hi} denote the data vector of the cluster. The CH_i is able to compute the projections of all data x^{Hi} collected from the nodes in it's a vector of M projections from the data within its cluster by using the CS technique. The value of M is determined by the number of nodes N and the sparsity level of the original data [5]. It then forwards them to the sink in M rounds along a backbone tree that connects all CHs to the sink.

$$y = \Phi x = \begin{bmatrix} \Phi^{H_1} & \Phi^{H_2} & \Phi^{H_3} & \Phi^{H_4} \end{bmatrix} \begin{pmatrix} x^{H_1} \\ x^{H_2} \\ x^{H_3} \\ x^{H_4} \end{pmatrix} = \sum_{i=1}^{4} \Phi^{H_i} x^{H_i}. \quad (1)$$

As shown in (1), the projections of all data in the network on the measurement matrix is the sum of the projections generated from the clusters. Thus in each round, the CH aggregates its own projection and the projections received from its children CHs in the same round and forwards it to the sink following the backbone tree. When the sink receives all M rounds of projections from CHs, the original data for all sensor nodes can be recovered. There are two levels of transmissions in our clustering method using the hybrid CS: intracluster transmissions that do not use the CS technique and intercluster transmissions that use the CS technique. The data size in intercluster transmissions is the same as the data in intracluster transmissions. Thus, reducing the number of transmissions can effectively reduce the energy consumption of sensor nodes. For intracluster transmissions, we simply let sensor nodes transmit their data to the CH following the shortest path routing (in terms of number of hops). For intercluster transmissions, the boundary node send the data towards sink. As cluster size increases, the number of intracluster transmissions would increase sharply. But when decreasing the cluster size, the number of clusters would increase and the number of intercluster transmissions would increase. Thus, there exists an optimal cluster size that minimizes the total number of data transmissions in the hybrid CS method. Our task is to determine the optimal cluster size and design a distributed clustering method, such that the total number of transmissions is minimized.

IV. MINIMUM TRANSMISSION CLUSTERING ALGORITHM

A. *Overview of Centralized Clustering Algorithm*

The sensor network is modeled by a graph G = (V, E), where V consists of the sink node v0 and N sensor nodes. If two nodes in V are within the communication range of each other, then there is a link between the two nodes.

As the centralized algorithm, we assume the sink node has the full knowledge of the network topology. That is, it knows the network graph G = (V, E). The sink will divide the sensor nodes into clusters, choose a CH for each cluster, and cluster head send the data to the boundary nodes finally it reaches the sink. After computing the clustering, sink select the CH for each cluster. From the theoretical analysis we can find the optimal cluster size Nc for a given number of N sensor nodes uniformly distributed in a field. Thus, the optimal number of clusters in the system is:

$$C = \left[\frac{N}{N_c^*} \right]. \quad (10)$$

In our method, within a cluster, each sensor node transmits its data to its designated CH via the energy efficient path. The routes that sensor nodes use to send their data to the CH form a energy efficient path in each cluster. The total number of intracluster transmissions is the sum of the distance of all sensor nodes to their CHs. Thus, the clustering problem for minimizing intracluster transmissions becomes a well-known k-median problem, that is to find the locations to place C CHs in the network G = (V ,E) such that the total distance from all sensor nodes to their nearest CHs is minimized. The distance between two nodes is defined as the number of hops of the shortest path between them. Data collected from sensor nodes is compressed by the CS method at the CHs. The data projections generated at each CH are forwarded to the sink in M rounds.

B. *Centralized Clustering Algorithm*

In this section, we present the centralized clustering algorithm. Given the network G =(V ,E), our algorithm has two major steps: 1) select C CHs from the set V of N sensor nodes and divide the sensor nodes into C clusters and 2)elect the CH and send the data to the sink. Our algorithm starts from an initial set of CHs, which is randomly selected. At each iteration, the algorithm proceeds following steps:

1. Connect sensor nodes to their closest CHs. Ties break arbitrarily.
2. For each cluster, choose a new CH, such that the sum of the distances from all nodes in this cluster to the new CH is minimized.
3. Repeat the above two steps until there is no more change of the CHs.

V. DISTRIBUTED IMPLEMENTATION

This section presents a distributed implementation of the clustering method. We assume that (1) every S_{node} knows its geographical location. This location information can be obtained by using anchor node. (2) the sink knows the area of the whole sensor field. In our distributed algorithm, the sink divides the field into C cluster-areas, calculates the geographic central point of each cluster-area. The sensor node that is the closest to the center of a cluster-area is selected to be the CH. The sensor nodes will join their respective clusters.

(a) Cluster head selection

Sink is deploying Sensor nodes in geographical areas, and stores the information about all deployed sensor nodes and the energy level and transmission range of that node. Sensor node will find out its current location by using anchor nodes, which are deployed in the network that are moving in the network area and equipped with GPS device. According to the location got it will maintain the details like which cluster it is falling, and who is its CH. Also the sensor node can get which are the neighbor nodes of that cluster.

ALGORITHM: CH Selection

1. Divide the geographical area into clusters.
2. Calculate the C_{point} of each cluster.
3. If (S_{node} falling nearby to the C_{point}).
4. Check the E_{level} of that S_{node} .
5. If ($E_{level} \geq SE_{level}$).
6. Declare that S_{node} as CH.
7. Else
8. Check other S_{node} which is having higher energy.

Clustering and central point selection will be done by using Minimum clustering algorithm specified in Section 4.

S_{node} will get the cluster which it is falling by using anchor nodes. Sink will check whether the current S_{node} is falling nearby to the central point or not. If it is falling near to the central point, it will check the energy level of that sensor node. If it is having sufficient energy, then sink will declare that node as CH of that cluster. If the energy level is less, it will check for other sensor nodes which are having higher energy.

(b)***Path Selection***

Find out which are the existing paths to reach CH and the energy used in that path by sending Route Request (R_{req}) to other neighbors and from the Route Reply (R_{rep}) which it will get from the neighbor. From the paths, it will select the path which is having highest energy level and maintains that details in its table.

ALGORITHM: Path Selection

1. Sink maintains details of deployed S_{node}.
2. S_{node} maintains the neighbor node and CH details of that cluster.
3. S_{node} find out all the existing paths E_{path} by using R_{req}-R_{rep} method
4. Also calculates the energy of E_{path}.
5. Select a E_{path} which is having higher E_{level} .
6. S_{node} Maintain the S_{path} detail in the table.

In the above algorithm E_{path} represent the path that is having the highest energy, E_{level} represent the energy level of the sensor node, S_{path} represent the selected path, S_{node} represent the sensor node and SE_{level} represent the sufficient energy level.

VI. SYSTEM MODULES

a. Sink:

This module will be having all the information like which all sensor nodes are deployed which is the CH of each node. The compressed data coming from CH will be reached in the sink and it will decompress it and store it.

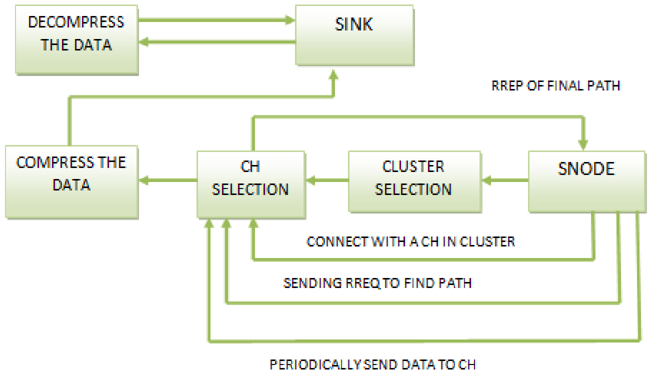

Fig 3.System Design

System Architecture is as shown in fig 3. Basically it is having 3 units. Sink, Cluster head and sensor node. Sensor node will sense the data and periodically send to the CH and CH will compress the data and will send to the boundary nodes of other cluster. Finally from the nearby boundary node from cluster sink will fetch the compressed data, and it will decompress it for fetching the sensed data.

b. SensorNode:

This module will periodically sense the data. It will be having the information about which cluster it's falling, and who is it's CH and neighbor node details of that cluster. Then it will find out a path which is having highest energy to reach the CH.

c. ClusterHead:

CH will get data from connected sensor nodes falling on that cluster and will compress the data and send to sink by using anchor nodes. Whenever a new node is joining it will check which cluster it is falling and will find out the distance with the centre point. If it is near to center make it as CH otherwise simple sensor node. Next sensor node will findout the path which is having highest energy and make it as final path and periodically send data to CH in that path. CH compress the data and send it to Sink. Finally in sink decompress the data and store it.

VII. CONCLUSION AND FUTURE ENHANCEMENT

Sensor nodes collect the data and sends to the storage node. The cluster head is declared based on the transmission range and the energy level. The anchor nodes are moving in the network area and equipped with GPS device and broadcast its current location periodically and the broadcasted messages are received by the sensor nodes. After receiving three messages, a sensor can find the location of the sensor node. While sending the collected information to the sink, a node first sends the data towards the boundary node of the next cluster and like this data is forwarded to the sink. Once a sensor node goes out of the transmission range of the cluster head it joins the cluster and sends the leave and join message to the cluster heads respectively. Then the cluster head update the node details in its cluster. Hence, it reduce the number of data transmission and energy consumption efficiently.

REFERENCES

[1] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler, “An Analysis of a Large Scale Habitat Monitoring Application,” Proc. ACM Second Int’l Conf. Embedded Networked Sensor Systems (SenSys ’04), pp. 214-226, Nov. 2004.
[2] J. Wang, S. Tang, B. Yin, and X.-Y. Li, “Data Gathering in Wireless Sensor Networks through Intelligent Compressive Sensing,” Proc. IEEE INFOCOM, pp. 603-611, Mar. 2012.
[3] R. Baraniuk, “Compressive Sensing [Lecture Notes],” IEEE Signal Processing Magazine, vol. 24, no. 4, pp. 118-121, July 2007.
[4] D. Donoho, “Compressed Sensing,” IEEE Trans. Information Theory, vol. 52, no. 4, pp. 1289-1306, Apr. 2006.
[5] 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.
[6] C. Luo, F. Wu, J. Sun, and C.W. Chen, “Compressive Data Gathering for Large-Scale Wireless Sensor Networks,” Proc. ACM MobiCom, pp. 145-156, Sept. 2009.
[7] S. Lee, S. Pattem, M. Sathiamoorthy, B. Krishnamachari, and A. Ortega, “Spatially-Localized Compressed Sensing and Routing in Multi-Hop Sensor Networks,” Proc. Third Int’l Conf. GeoSensor Networks (GSN ’09), pp. 11-20, 2009.
[8] C. Luo, F. Wu, J. Sun, and C.W. Chen, “Efficient Measurement Generation and Pervasive Sparsity for Compressive Data Gathering,” IEEE Trans. Wireless Comm., vol. 9, no. 12, pp. 3728-3738, Dec. 2010.
[9] J. Luo, L. Xiang, and C. Rosenberg, “Does Compressed Sensing Improve the Throughput of Wireless Sensor Networks?” Proc. IEEE Int’l Conf. Comm (ICC), pp. 1-6, May 2010.
[10] F. Fazel, M. Fazel, and M. Stojanovic, “Random Access Compressed Sensing for Energy-Efficient Underwater Sensor Networks,” IEEE J. Selected Areas Comm., vol. 29, no. 8, pp. 1660-1670, Sept. 2011.