A COST EFFECTIVE COMPRESSIVE DATA AGGREGATION TECHNIQUE FOR WIRELESS SENSOR NETWORKS

M.Y. Mohamed Yacoab[1], Dr.V.Sundaram[2] and Dr.A.Thajudeen[3]
[1]Research Scholar, Karpagam University, Coimbatore, India
yacoab@yahoo.com
[2]Director, MCA Department, Karpagam College of Engineering, Coimbatore, India
dr_vsundaram@yahoo.com
[3]Director, Measi Institute of Information Technology, Chennai India
thaj_a2001@yahoo.com

ABSTRACT

In wireless sensor network (WSN) there are two main problems in employing conventional compression techniques. The compression performance depends on the organization of the routes for a larger extent. The efficiency of an in-network data compression scheme is not solely determined by the compression ratio, but also depends on the computational and communication overheads. In Compressive Data Aggregation technique, data is gathered at some intermediate node where its size is reduced by applying compression technique without losing any information of complete data. In our previous work, we have developed an adaptive traffic aware aggregation technique in which the aggregation technique can be changed into structured and structure-free adaptively, depending on the load status of the traffic. In this paper, as an extension to our previous work, we provide a cost effective compressive data gathering technique to enhance the traffic load, by using structured data aggregation scheme. We also design a technique that effectively reduces the computation and communication costs involved in the compressive data gathering process. The use of compressive data gathering process provides a compressed sensor reading to reduce global data traffic and distributes energy consumption evenly to prolong the network lifetime. By simulation results, we show that our proposed technique improves the delivery ratio while reducing the energy and delay.

KEYWORDS

Wireless Sensor Network, Data Aggregation And Data Gathering, Compressive Data Gathering

1. INTRODUCTION

1.1 Wireless Sensor Networks

Wireless sensor networks include the emerging technologies which have received major attention from the research community. The sensor network which is self organizing ad hoc system comprises of several small and low cost devices. It observes the physical environment, collect the information and transmit it to one or more sink nodes. Generally, the radio transmission range of the sensor nodes are in the orders of magnitude which is smaller than the geographical extent of the entire network. Therefore, data should be transmitted towards the sink

DOI : 10.5121/ijasuc.2010.1411

node hop-by-hop in a multi-hop manner. By reducing the amount of data which is to be transmitted, the energy consumption of the network can also be reduced [1]. A large number of small electromechanical devices with sensing, computing and communication capabilities are included in the wireless sensor networks. It can be used for collecting sensory information, such as temperature measurements, from an extended geographic area [2].

The possible uses of the sensor networks have been researched actively. Due to the characteristics of the wireless sensor network several challenging issues are created. The following characteristics are mainly focused:

- Sensor nodes tend to fail.
- Sensor nodes utilize a broadcast communication paradigm and have severe bandwidth constraints.
- Sensor nodes have limited resources [3].

1.2 Data Aggregation and Data Gathering

A common function of sensor networks is data gathering. In data gathering the information sampled at sensor nodes has to be transported to the central base station for further processing and analysis. An important topic mentioned by the wireless sensor network community is the in-network data aggregation while focusing on the severe energy constraints of the sensor nodes and the limited transport capacity of multi-hop wireless networks. The basic idea for minimizing the expense of data transmission is to pre-process the sensor data in the network by the sensor nodes [4].

One of the basic distributed data processing procedures in the wireless sensor networks is data aggregation. It is used to save the energy and to reduce the medium access layer contention [5]. The idea is to combine the data coming from different sources, eliminating the redundancy and reduce the number of transmissions, thus saving the energy [6]. By using the in-network data aggregation, the natural redundancy in the raw data collected from the sensors can be eliminated. Moreover, such operations are useful for extracting the specific information from the data. Supporting high frequency of in-network data aggregation is severe for the network in order to conserve energy for a longer network lifetime.

1. 3 Need of Compressive Data Aggregation technique in WSN

In wireless sensor network (WSN) there are two main problems with conventional compression techniques.

- The compression performance relies heavily on how the routes are organized. In order to achieve the highest compression ratio, compression and routing algorithms need to be jointly optimized.
- The efficiency of an in-network data compression scheme is not solely determined by the compression ratio, but also depends on the computational and communication overheads [7].

In this situation, Compressive Data Aggregation technique helps to cope up with these issues. In this technique, data is gathered at some intermediate node where the data size size is reduced by applying compression technique without losing any information of complete data. Compressive Data Aggregation technique requires each node in the WSN to send exactly k packets irrespective of what it has received, which means, compared with traditional techniques , more work/load for the nodes which are far away from the sink and less work/load for the nodes that are close to the sink. Data compression and aggregation technique have the potential to improve WSN energy efficiency and minimize communication [8].

1.4 Adaptive Traffic Aware Data Aggregation Technique

In our previous work [9], we have proposed an adaptive traffic aware aggregation technique for wireless sensor networks. In this work, a multi path structured tree is constructed in which nodes are selected based on their residual energy level. A traffic monitoring agent is used to monitor the load status of the event traffic and each node estimates its traffic load during the data reception. At the sink, it estimates the total traffic load in the system and sends an OVERLOADED packet to the sources if it is greater than a threshold level T. Then the aggregation technique is changed to structure-free lossy aggregation by the sources. If the traffic load is less than the threshold value T, the sink sends UNDERLOADED packet to the sources and then sources change the aggregation mode to the structured lossless aggregation. This technique eventually provides a reliable transmission environment with low energy consumption, by efficiently utilizing the energy availability of the forwarding nodes to gather and distribute the data to sink, according to its requirements.

As an extension of our previous work, we provide a compressive data gathering technique to enhance the traffic load, when structured data aggregation is used. The use of compressive data gathering provides a compressed sensor reading to reduce global data traffic and distributes energy consumption evenly to prolong network lifetime. We can also increase the efficiency level if the correlated sensor readings are transmitted jointly rather than separately.

2. RELATED WORK

Marco F. Duarte et al [11] have introduced a new theory for distributed compressed sensing (DCS) that enables new distributed coding algorithms for multi-signal ensembles that exploit both intra- and inter-signal correlation structures. They also proposed algorithms for joint recovery of multiple signals from incoherent projections.

Zainul Charbiwala et al [12] have proposed that if CS is employed for source compression, then Compressive Sensing (CS) can further be exploited as an application layer erasure coding strategy for recovering missing data. They showed that CS erasure encoding (CSEC) with random sampling is efficient for handling missing data in erasure channels, paralleling the performance of BCH codes, with the added benefit of graceful degradation of the reconstruction error even when the amount of missing data far exceeds the designed redundancy.

Wenbo He et al [13] proposed a two privacy-preserving data aggregation schemes for additive aggregation functions. The first scheme – Cluster-based Private Data Aggregation (CPDA) – leverages clustering protocol and algebraic properties of polynomials. The second scheme – Slice-Mix AggRegaTe (SMART) – builds on slicing techniques and the associative property of addition. The goal of this work is to bridge the gap between collaborative data collection by wireless sensor networks and data privacy.

Maarten Ditzel et al [14] presented the results of a study on the effects of data aggregation for multi-target tracking in wireless sensor networks. Wireless sensor networks are normally limited in communication bandwidth. The nodes implementing the wireless sensor network are themselves limited in computing power and usually have a limited battery life. These observations are recognized and combined to come to efficient target tracking approaches.

Steffen Peter et al [15] described and evaluated three algorithms that were reported to suit to the WSN scenario. As result of the evaluation, where emphasize is on the awareness to potential attack scenarios, a brief overview of strengths and weaknesses of the algorithms is presented. Since no algorithm provides all desirable goals, two approaches to cope with the problems are proposed. The first is the successive combination of two algorithms. It increases security, while the additional efforts can be minimized by carefully selected parameters. For the second approach, specific weaknesses are faced and so mechanisms are engineered that solve the particular issues.

3. STRUCTURED TREE CONSTRUCTION

Initially we will describe the structured tree construction algorithm presented in our previous paper.

We consider the wireless sensor network M as a directed graph G (N, E). Let the set of nodes N consists of sensors and (a, b) $\in$ E if a and b are residing inside the transmission range of each other. The fundamental idea of the proposed algorithm is, when a data gathering request is arrived, then using the greedy algorithm a data gathering tree for the request is constructed. The greedy algorithm maximizes the minimum residual energy among the nodes. Then the nodes are included in the tree one by one but in the beginning only the sink node is included. A node b is selected to be included into the tree if the causes to maximize the minimum residual energy among the trees are included.

In our algorithm, we use the following notations

- N is the total number of nodes
- N_T is the set of nodes in the tree,
- stop is a Boolean variable,
- newnode is the node that will be added to the tree.
- q is the size of the sensed data by newnode.
- $w^{\alpha}_{a,b}$ is the weight assigned to the edge.
- R is the set of nodes that are not in the tree.
- RE is the residual energy.
- s is the sink node
- mre_{max} is the maximum value of minimum residual energy at each node of the tree.
- tp is the temporary parent node.
- $P_{a,s}$ is the unique path in T from node a to node s
- p(a) is the parent of a in T
- Let node v $\in$ N - N_T be the considered node.

3. 1. 1. Tree Construction Algorithm

Algorithm.1

1. $N_T = \{s\}$
2. Stop = "false"
3. R = N - N_T
4. RE(s) = ∞
5. $mre_{max} = 0$
6. for each i $\in$ R

6.1 Compute mre_{max} (i) and tp
6.2. If mre_{max} (i) > mre_{max}, then
6.2.1. $mre_{max} = mre_{max}(i)$
6.2.2. Newnode = i
6.3 End if
7. End for
8 If $mre_{max} > 0$, then
8.1. P (newnode) = tp(newnode)

8.2. For each $j \in P_{newnode, s}$ do
8.2.1. RE (j) = RE (j) - $qw^{\alpha}_{j,p(j)}$
8.3 End for
8.4. $N_T = N_T \cup \{newnode\}$
8.5 R = R - newnode
9 Else
9.1 stop = "True"
10. End if
11. If ($R \neq \phi$) or stop="false" then
11.1 repeat from 5
12. End if
13. End

4. ADAPTIVE COMPRESSIVE DATA GATHERING AND RECOVERY

4. 1. Compressive Data Gathering

The intuition behind CDG is that higher efficiency can be achieved if correlated sensor readings are transmitted jointly rather than separately. We have given a simple example in Section I, showing how sensor readings are combined while being relayed along a chain-type topology to the sink. In practice, sensors usually spread in a two-dimensional area, and the ensemble of routing paths presents a tree structure. Fig. 4(a) shows a typical routing tree in which the sink has four children. Each of them leads a sub tree delimited by the dotted lines. Data gathering and reconstruction of CDG are performed on the sub tree basis.

The perception behind CDG is that joint transmission of the correlated sensor readings instead of transmission of the readings separately will increase the efficiency to a higher level. The combining of the sensor readings when it is being transmitted to the sink along the chain type topology is shown as an example in section 1. Generally sensors spread in the two dimensional area and the structure represented by the routing paths is a tree structure. A routing tree with four children at the sink is shown in the Fig. 4(a). A sub tree delimited by the dotted lines is lead by each of them. On the sub tree basis data gathering and reconstruction of the CDG are performed.

In order to combine sensor readings while relaying them, every node needs to know its local routing structure. That is, whether or not a given node is a leaf node in the routing tree or how many children the node has if it is an inner node. To facilitate efficient aggregation, we have made a small modification to standard ad-hoc routing protocol: when a node chooses a parent node, it sends a "subscribe notification" to that node; when a node changes parent, it sends an "unsubscribe notification" to the old parent.

Each node should know its local routing structure so as to combine the sensor readings when it is being transmitted. That is, if the given node in the routing tree is a leaf node or not or if the node is an inner node then how many children does it have. To the standard routing protocol, a small modification is

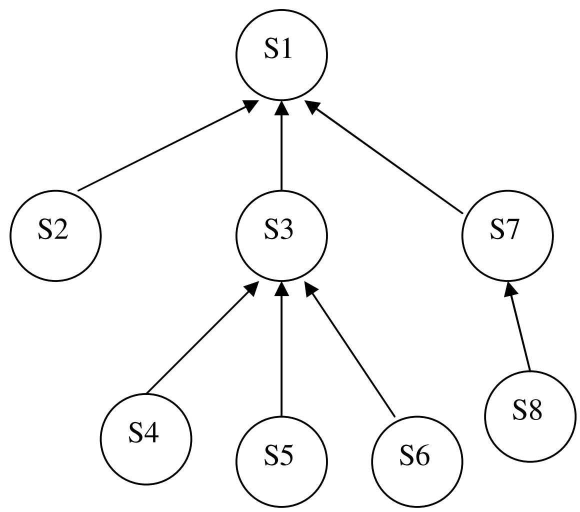

fig. 1

done so as to facilitate proficient aggregation: when a parent node is chosen by the node, it transmits a “subscribe notification” to that node and an “unsubscribe notification” is sent to the old parent, when the node changes the parent.

The data gathering process of CDG is illustrated through an example shown in Fig. 4(b). It is the detailed view of a small fraction of the routing tree marked in Fig. 4(a). After all nodes acquire their readings, leaf nodes initiate the transmission.

The example shown in fig. 1 illustrates the data gathering process of CDG. The leaf nodes will initiate the transmission only after all nodes receive their readings.

In this example, S_2 generates a random number α_{i2}, computes $\alpha_{i2}v_2$, and transmits the value to S_1. The index i denote the i^{th} weighted sum ranging from 1 to M. Similarly, S4, S5 and S6 transmit $\alpha_{i4}v_4$, $\alpha_{i5}v_5$, and $\alpha_{i6}v_6$ to S_3. Once S_3 receives the three values, it computes $\alpha_{i3}v_3$, adds it to the sum of relayed values and transmits $\sum_{j=3}^{6}\alpha_{ij}v_j$ to S_1. Then S_1 computes $\alpha_{i1}v_1$ and transmits $\sum_{j=1}^{8}\alpha_{1j}v_j$. Finally, the message containing the weighted sum of all readings in a sub tree is forwarded to the sink.

In this example, a random number α_{i2} is generated by S2 and it computes $\alpha_{i2}v_2$ and then the value is sent to S1. The ith weighted sum is denoted by the index i which ranges from 1 to M. Likewise $\alpha_{i4}v_4$, $\alpha_{i5}v_5$, and $\alpha_{i6}v_6$ is transmitted to S_3 by S4, S5 and S6. After the three values are received by S3 it will compute the value $\alpha_{i3}v_3$ and then it adds to the sum of the relayed value. It then transmits to S1 the value $\sum_{j=3}^{6}\alpha_{ij}v_j$. Next $\alpha_{i1}v_1$ is computed by the node S1 and

$\sum_{j=1}^{8} \alpha_{1j} v_j$ is transmitted. Lastly, to the sink, the message which contains the weighted sum of all readings in a sub tree is forwarded.

Assume that there are N nodes in a particular tree, and the sink intends to collect M measurements. Then all nodes send the same number of O (M) messages regardless of their hop distance to the sink. The overall message complexity is O (NM). When M << N, CDG transmits less messages than the baseline data collection whose worst case message complexity is O (N^2). More importantly, the transmission load is spread out uniformly so that the lifetime of bottleneck sensors and the entire network is greatly extended.

In a specific tree, if it is assumed to have N nodes and M measurements are intended to be collected by the sink. Then regardless of the hop distance of the node to the sink, all nodes will send the same number of O (M). O (NM) will be the overall message complexity. If M << N, then less messages are transmitted by CDG when compared with the baseline data collection when O (N^2) is the worst case message complexity. More importantly, for the extension of the lifetime of the bottleneck sensors as well as the entire network, the transmission load is spread uniformly.

The i^{th} weighted sum can be represented by:

$$A_i = \sum_{j=1}^{N} \alpha_{ij} v_j \qquad (1)$$

The sink obtains M weighted sums $\{A_i\}$, i = 1, 2 ...M.
Mathematically, we have:

$$\begin{pmatrix} A_1 \\ A_2 \\ . \\ . \\ . \\ A_M \end{pmatrix} = \begin{pmatrix} \alpha_{11}\alpha_{12} \ldots\ldots \alpha_{1N} \\ \alpha_{21}\alpha_{22} \ldots\ldots \alpha_{2N} \\ . \\ . \\ \alpha_{M1}\alpha_{M2} \ldots \alpha_{MN} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ . \\ . \\ . \\ v_N \end{pmatrix} \qquad (2)$$

In this equation, each column of $\{\alpha_{ij}\}$ contains the series of random numbers generated at a corresponding node. In order to avoid transmitting this random matrix from sensors to the sink, we can adopt a simple strategy: before data transmission, the sink broadcasts a random seed to the entire network. Then each sensor generates its own seed using this global seed and its unique identification. With a pre-installed pseudo random number generator, each sensor is able to generate the corresponding series of coefficients. These coefficients can be reproduced at the sink given that the sink knows the identifications of all sensors.
In the above equation, series of random numbers are placed in each column of $\{\alpha_{ij}\}$ which is produced at the corresponding node. A simple strategy is used for preventing the transmission of the random matrix from sensors to the sink: a random seed is broadcasted to the entire network before transmission. Using this global seed and its unique identification, each sensor will generate its own seed. Each sensor generates a corresponding series of coefficients from a pre-installed pseudo random number generator. Given that the sink knows the identifications of all sensors, the coefficients can be reproduced at the sink.
In (2), v_i (i = 1, 2 ...N) is a scalar value. In a practical sensor network, each node is possibly attached with a few sensors of different type, e.g. a temperature sensor and a humidity sensor. Then sensor readings from each node become a multi-dimensional vector. In this case, we may

separate readings of each dimension and process them respectively. Alternatively, since the random coefficients α_{ij} are irrelevant to sensor readings, we may treat v_i as a vector. The weighted sums A_i become vectors of the same dimension too.

In (2), v_i (i = 1, 2 ...N) is a scalar value. Each node is possibly attached with a few sensors of different type, e.g., temperature sensor and a humidity sensor in a practical sensor network. Then from each node, the sensor readings become a multi dimensional vector. In this case, in each dimension we may separate the readings and process them. Alternatively, since for the sensor readings, the random coefficients α_{ij} are irrelevant, v_i is treated as a vector. A_i which is a weighted sum become vectors of the same dimension too.

When M < N, solving a set of M linear equations with N unknown variables is an ill-posed problem. However, sensor readings are not independent variables. In most cases, the sensor field follows a certain structure because of the spatial or temporal correlations. Hence, there exists a transform domain in which the signal is sparse. Under this assumption, we will explain in the following subsection whether the set of linear equations are solvable, what requirements M should meet to solve them, and how these equations can be solved.

When M < N, with N unknown variables, solving a set of M linear equations is an ill-posed problem. But sensor readings are no where independent variables. In most cases, a certain structure is followed by the sensor field due to the spatial or temporal correlations. So, a transform domain is used wherever the signal is sparse. Based on this assumption in the following subsections we explain: whether linear equation set is solvable, to meet them what are the requirements M and these equations can be solved.

4.2 Data recovery

According to compressive sampling theory, a K-sparse signal can be reconstructed from a small number of measurements with a probability close to one. The weighted sums obtained in (2) are a typical type of measurements. Signal sparsity characterizes the correlations within a signal. An N-dimensional signal is considered as a K-sparse signal if there exists a domain in which this signal can be represented by K (K _ N) non-zero coefficients. Fig. 5(a) shows a 100-dimensional signal in its original time domain. Obviously, it is not sparse at all in this domain. Because of the signal correlation, it can be described more compactly in transform domains such as wavelet and DCT.

A K-sampling signal according to the compressive sampling theory can be reconstructed based on the small number of measurements having a probability nearly one. The weighted sum from (2) is measurements of typical type. Within a signal, signal sparsity characterizes the correlations. An N-dimensional signal is called as a K-sparse signal when there is a domain where signal can be presented as K(K_N) non zero coefficients. Fig.5(a) represents a 100-dimensional signal in its real time domain. Due to signal correlation, in transform domains such as wavelet and DCT, it can be described more compactly.

In a densely deployed sensor network, sensors have spatial correlations in their readings.

Sensors have spatial correlations in its readings in a densely deployed sensor networks.

Let N sensor readings form a vector $v = [v_1\ v_2\ ...\ v_N]^T$, then v is a K-sparse signal in a particular domain λ. Denote $\lambda = [\lambda_1\ \lambda_2\ ...\ \lambda_N]$ as the representation basis with vectors $\{\lambda_i\}$ as columns, and $X = [X_1, X_2\ ...XN]^T$ are the corresponding coefficients. Then,

v can be represented in the λ domain as:

$$v = \sum_{i=1}^{N} X_i \lambda_i \quad \text{Or} \quad v = \lambda X \tag{3}$$

Compressive sampling theory tells that a K-sparse signal can be reconstructed from M measurements if M satisfies the following conditions [6]:

$$M \geq \beta.\ \Delta^2 (\alpha, \lambda).\ K.\log N \tag{4}$$

where β is a positive constant, α is the sampling matrix as defined in (2), and $\Delta(\alpha, \lambda)$ is the coherence between sampling basis α and representation basis λ. The coherence metric measures the largest correlation between any two elements of α and λ, and is defined as:

$$\Delta(\alpha, \lambda) = \sqrt{N}. \quad \text{Max } |< \alpha_i, \lambda_j >|,\ 1 \leq i,\ j \geq N \tag{5}$$

From (5), we can see that the smaller the coherence between α and λ is, the lesser measurements are needed to reconstruct the signal. In practice, using random measurement matrix is a convenient choice, since a random basis has been shown to be largely incoherent with any fixed basis and M = 3K ~ 4K is usually sufficient to satisfy (4).

From eq. (5), we get to know that lesser is the coherence in between α and λ, reduced measurements are required for the signal reconstruction. In practice, a convenient choice is to use random measurement matrix, since with any fixed basis a random basis is shown to be largely incoherent and M = 3K – 4K is sufficient to satisfy eq. (4).

With sufficient number of measurements, the sink is able to reconstruct sensor readings by solving an l_1-minimization problem:

$$\min_{X \in \Re^N} \|X\|_{l_1} \quad \text{s.t. } A = \alpha v,\ v = \lambda X \tag{6}$$

In addition, for sparse signals whose random projections are contaminated with noise, reconstruction can be achieved by solving a relaxed l_1-minimization problem, where is a predefined error threshold:

$$\min_{X \in \Re^N} \|X\|_{l_1} \quad \text{s.t. } \|A - \alpha v\|_{l_2} < \varepsilon,\ v = \lambda X \tag{7}$$

Suppose $\tilde{Y}$ is the solution to this convex optimization problem, then the proposed reconstruction of the original signal is $\tilde{u} = \lambda \tilde{Y}$. It has been shown that the above l_1-minimization problem can be solved with linear programming (LP) techniques. Although the reconstruction complexity of LP based decoder is polynomial, it goes pretty high when N is too large. While there is a large body of on-going work looking for low-complexity reconstruction techniques, this topic is beyond the scope of our paper. With the current LP based decoder, we would suggest that the size of N does not exceed one thousand.

Suppose for the convex optimization problem if $\tilde{Y}$ is the solution, then for the original signal the proposed reconstruction is $\tilde{u} = \lambda \tilde{Y}$. The linear programming (LP) techniques can be used for solving the above mentioned l_1-minimization problem. When N is too large the reconstruction complexity of LP based decoder goes pretty high even though initially it is a polynomial function. This topic is beyond the scope of our paper, when there is a large body of on going work looking for low complexity reconstruction technique. We would suggest, for the current LP based decoder that the size of N does not exceed one thousand.

In (6) and (7), the λ matrix describes the correlation pattern among sensor readings. It is utilized only in data recovery process, and is not required to be known to sensors. In this way, most of the computations are shifted from sensors to the sink. Such asymmetry of computation complexity makes CDG an appealing choice for WSNs.

In (6) and (7) the correlation pattern among the sensor reading is described by the λ matrix. It is not required to be known to the sensors since it is utilized in data recovery process. Likewise, most of the calculations are transferred from the sensors to the sink. As a result of the asymmetry of computation complexity, CDG is an appealing choice for WSN's.

5. MINIMIZING COMPRESSION COST

Our objective is to minimize both the computation and communication costs rather than minimizing communication cost only, as done in the prior works. To perform the compression over the data gathering tree (given in the last section), we propose a flow based technique where data from each source is compressed and transmitted as a traffic flow over the corresponding path from the source to the sink.

5. 1. Network Model

The underlying wireless network is modeled as a connected weighted graph

$$G = < N, L, W >,$$

Where the vertex set N represents the set of n sensor nodes; the edge (link) set L represents the wireless connection between nodes and associated with each edge $l_i \in L$, its weight W_{li} is the energy cost of sending a data packet of unit size over l_i. The link weight is determined by the distance between the two adjacent nodes, the radio device, and the communication environment. We also use (u, v) to denote an edge connecting u and v.

Let $s_i \in N$ denote the sink node and $S \subseteq N$, denote the set of source nodes. In each period, each source node generates a raw data of one unit size that needs to be transported to the sink, possibly via multi-hop communication.

A data gathering tree is a sub-tree of G rooted at sink and containing S, denoted as

$$T = < N', L' >,$$

where $S \subseteq N' \subseteq N$ and $L' \subseteq L$. Let M_n denote the number of source nodes in the sub-tree rooted at $n \in N$. Given a data gathering tree, let P_s denote the path in the tree that connects s to s_i.

Also, for two edges l_1, l_2 on the same path, let $l_1 \prec l_2$ denote the fact that l_1 is a predecessor of l_2.

We define a pre-defined system parameter, $cost_{comp} \geq 0$, to represent the energy cost of compressing one unit of data (using (1) and (2) in section 4) normalized by the cost of communicating one unit of data. The energy cost of compressing source information of size z to an output of size o is represented as a function

$$F(o) = Cost_{comp} * z * CR \qquad (8)$$

Where $CR = \frac{z}{o}$ is the compression ratio.

From (1), it can be seen that the energy cost is

- Proportional to the input size z since it has to process the whole input at least once,
- Proportional to the compression ratio CR.

If. $l = (u, v)$ denote a one-hop link, where u generates a data packet of one unit size that needs to be transmitted to v after appropriate compression by u, then z=1 and Eq. (1) can be modified as.

$$F(o) = \frac{Cost_{comp}}{o}$$

Let W_l denote the cost of transmitting one unit of data over the link l. The overall energy costs, denoted as $Cost_{energy}(o)$ can then be modeled as follows

$$Cost_{energy}(o) = \frac{Cost_{comp}}{o} + o.\ W_l \qquad (9)$$

5. 2. Cost Effective Data Gathering

Given a data gathering tree (as described in section 3) over a sensor network, we model data transmission over the tree as a composition of different data flows from each source node to s_i. That is, each path from a source node to s_i in the tree corresponds to a data flow over the path. The flow size may change along its corresponding path due to data compression performed by intermediate nodes. Also, the energy cost of the system is the sum of the computation and communication costs of all paths in the tree.

Consider an arbitrary path P_s in the tree from a source node n to s_i. Let f_l^n denote the flow over $l \in P_s$ and q_s denote the last edge in P_s, i.e., the edge incident on s_i in P_s. We assume that the total energy spent on data compression over the path P_s is determined by the flow on q_s, i.e., the total energy cost for data compression over P_s is calculated as

$$P_s = \frac{Cost_{comp}}{f^{s}_{q_s}} \quad (10)$$

Given a node in the tree, the number of incoming flows equals the number of source nodes in its sub-tree. The output size required for compressing each incoming packet is lower bounded by the joint entropy of these source nodes. We assume that the joint entropy of any i source nodes, E_i is a non-decreasing and concave function of i with $E_1 = E_D$, where $E_D \in (0, 1)$ is the entropy of one unit of data. We assume that the compression of i incoming data flows at node n can be performed in such a way that the lower bound for compressing each data flow equals $LB_i = \frac{E_i}{i}$, with $LB_1 = E_1 = E_D$. In other words, we assume that when maximal compression is performed on i pieces of source information, the fraction of compressible data of each piece is the same.
Thus, for any $l = (a, b) \in P_s$, we impose the constraint on f_l^n such that

$$f_n^l \geq LB_{M_n} = \frac{E_{M_n}}{M_n} \quad (11)$$

Let $\lambda_i = LB_{M_n}$, where M_n is the number of source nodes in the sub-tree rooted at $n \in N$. Also, we have $\lambda_i \geq \lambda_{i+1}$, for $i = 1 \ldots k - 1$.
Given a data gathering tree and an arbitrary source node $s \in S$, consider the path from s to s_i. Let $P_s = \{s_1, s_2 \ldots s_k\}$ denote the path, where $s_1 = s$, $s_k = s_i$, and k is the number of nodes along P_s. We need to compress and transmit a packet of unit size from s_1 to s_i with the minimal computation and communication energy costs.
Let $\vec{f}$ denote a vector with flow along P_s, i.e., $\vec{f} = \{ f^l_{n_1}, \ldots, f^l_{n_{k-1}} \}$.
For any optimal flow $\vec{f}$ over a path P_s, if $f_{i+1} < f_i$, we have $f_i = \lambda_i$.

6. SIMULATION RESULTS

6. 1. Simulation Setup

The performance of our cost effective compressive data aggregation (CECDA) technique is evaluated through NS2 [13] simulation. A random network deployed in an area of 500 X 500 m is considered. We vary the number of nodes as 20, 40….100. Initially the nodes are placed randomly in the specified area. The base station is assumed to be situated 100 meters away from the above specified area. The initial energy of all the nodes is assumed as 3.1 joules. In the simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. The distributed coordination function (DCF) of IEEE 802.11 is used for wireless LANs as the MAC layer protocol. The simulated traffic is CBR with UDP source and sink. The number of sources is varied from 1 to 5.
Table 1 summarizes the simulation parameters used

TABLE 1: SIMULATION PARAMETERS

No. of Nodes	20,40,….100
Area Size	500 X 500
Mac	802.11
Simulation Time	50 sec
Traffic Source	CBR
Packet Size	512
Transmit Power	0.660 w

Receiving Power	0.395 w
Idle Power	0.335 w
Initial Energy	3.1 J
Transmission Range	75m

6. 2. Performance Metrics

The performance of CECDA technique is compared with our previous ATAA [9] protocol**.** The performance is evaluated mainly, according to the following metrics.

Average end-to-end Delay: The end-to-end-delay is averaged over all surviving data packets from the sources to the destinations.

Average Packet Delivery Ratio: It is the ratio of the number of packets received successfully and the total number of packets transmitted.

Energy Consumption: It is the average energy consumed by all the nodes in sending, receiving and forwarding operations

The simulation results are presented in the next section.

6. 3. Simulation Results

A. Dense

In our initial experiment, we vary the number of nodes as 20, 40, 60, 80 and 10 in which the sources are densely deployed.

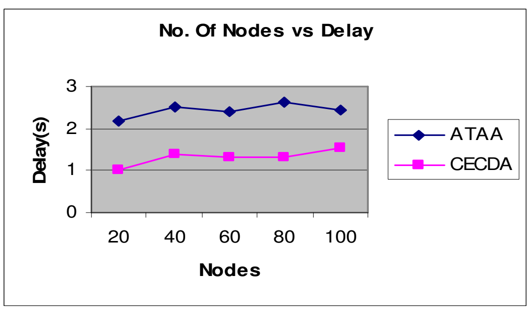

Fig 1: Nodes Vs Delay

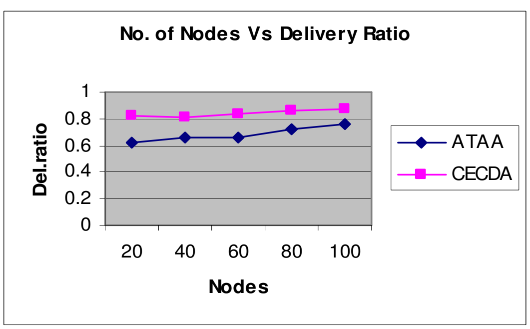

Fig 2: Nodes Vs DelRatio

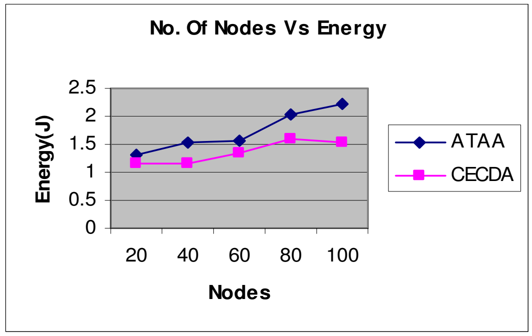

Fig 3: Nodes Vs Energy

Since the aggregation involves compressed data, the delay incurred in sending the data from sensors to the sink, will be significantly reduced. Fig 1 gives the average end-to-end delay when the number of nodes is increased. From the figure, it can be seen that the average end-to-end delay of the proposed CECDA technique is less when compared with ATAA.

Fig 2 presents the packet delivery ratio when the number of nodes is increased. The compressed data aggregation eliminates the packet drops at the intermediate nodes and hence increases the packet delivery ratio. So CECDA achieves good delivery ratio, compared to ATAA.

Compressing the data during data aggregation reduces the number of data packets to be aggregated at the aggregator nodes. Hence the total energy consumption involved in the aggregation process will also be reduced. Fig 3 shows the results of energy consumption when the number of nodes is increased. From the results, we can see that CECDA technique has less energy consumption when compared with ATAA, since it has the energy efficient tree.

B. Sparse

In our second experiment, we vary the number of nodes as 20, 40, 60, 80 and 10 in which the sources are sparsely deployed.

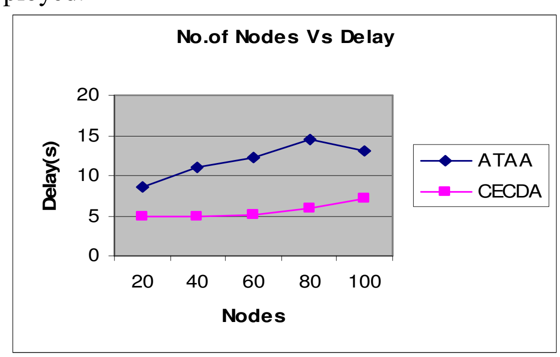

Fig 4: Nodes Vs Delay

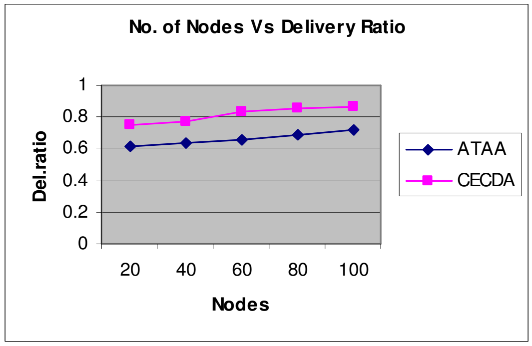

Fig 5: Nodes Vs DelRatio

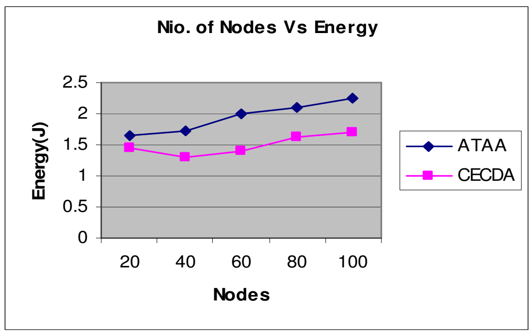

Fig 6: Nodes Vs Energy

Since the aggregation involves compressed data, the delay incurred in sending the data from sensors to the sink, will be significantly reduced. Fig 4 gives the average end-to-end delay when the number of nodes is increased. From the figure, it can be seen that the average end-to-end delay of the proposed CECDA technique is less when compared with ATAA.

Fig 5 presents the packet delivery ratio when the number of nodes is increased. CECDA achieves good delivery ratio, compared to ATAA. The compressed data aggregation eliminates the packet drops at the intermediate nodes and hence increases the packet delivery ratio.

Fig 6 shows the results of energy consumption when the number of nodes is increased. Compressing the data during data aggregation reduces the number of data packets to be aggregated at the aggregator nodes. Hence the total energy consumption involved in the aggregation process will also be reduced. From the results, we can see that CECDA technique has less energy consumption when compared with ATAA, since it has the energy efficient tree.

7. CONCLUSION

Compressive Data Aggregation technique helps to solve the issues of traditional compression techniques. In this technique data is gathered at some intermediate node where size of the data need to be sent is reduced by applying compression technique without losing any knowledge of complete data. In our previous work, we have developed an adaptive traffic aware aggregation technique in which the aggregation technique is adaptively changed to structured and structure-free, depending on the load status of the traffic. In this paper, as an extension of our previous work, we have provided a compressive data gathering technique to enhance the traffic load, when structured data aggregation is used. We have also designed a technique that effectively reduces the computation and communication costs involved in the compressive data gathering process. The use of compressive data gathering provides a compressed sensor reading to reduce global data traffic and distributes the energy consumption evenly to prolong network lifetime.

By simulation results, we have shown that our proposed technique improves the delivery ratio while reducing the energy and delay.

References

[1] Dorottya Vass and Attila Vidacs, “Distributed Data Aggregation with Geographical Routing in Wireless Sensor Networks”, Proc., IEEE International Conference on Pervasive Services (ICPS), Istanbul, Turkey, 15-20 July, 2007.

[2] Jukka Kohonen, “Data Gathering in Sensor Networks”, Helsinki Institute for Information Technology in Finland, Nov-2004.

[3] Gregory Hartl, Baochun Li, “Loss Inference in Wireless Sensor Networks Based on Data Aggregation”, IPSN 2004.

[4] Tao Cui, Lijun Chen, Tracey Ho, Steven H. Low, and Lachlan L. H. Andrew, “Opportunistic Source Coding for Data Gathering in Wireless Sensor Networks”, Proceedings of the 2007 conference on Diversity in computing Orlando, Florida, 2007.

[5] Zhenzhen Ye, Alhussein A. Abouzeid and Jing Ai, “Optimal Policies for Distributed Data Aggregation in Wireless Sensor Networks”, Draft Infocom2007 Paper.

[6] Bhaskar Krishnamachari, Deborah Estrin and Stephen Wicker, “The Impact of Data Aggregation in Wireless Sensor Networks”, Proceedings of the 22nd International Conference on Distributed Computing Systems – 2002.

[7] Chong Luo and Feng Wu, “Compressive Data Gathering for Large-Scale Wireless Sensor Networks”, Proceedings of the 15th annual international conference on Mobile computing and networking, Beijing, China, 2009.

[8] Hengyu Long et al, “Energy-Efficient Spatially-Adaptive Clustering and Routing in Wireless Sensor Networks”, Proceedings of Design, Automation & Test in Europe Conf, 2009.

[9] M.Y. Mohamed Yacoab and Dr. V. Sundaram, “An Adaptive Traffic Aware Data Aggregation Technique for Wireless Sensor Networks”, American Journal of Scientific Research Issue 10, 2010.

[10] Riccardo Masiero et al, “A Bayesian Analysis of Compressive Sensing Data Recovery in Wireless Sensor Networks”, IEEE SASN 2009, Saint Petersburg, Russia, Oct. 12-14, 2009.

[11] Dror Baron et al, “An Information-Theoretic Approach to Distributed Compressed Sensing”, Proceedings of the 43d Allerton Conference on Communication, Control, and Computing, Monticello, IL, September, 2005.

[12] Zainul Charbiwala et al, “Compressive Oversampling for Robust Data Transmission in Sensor Networks”, IEEE INFOCOM, San Diego, CA, March 2010.

[13] Wenbo He, Xue Liu, Hoang Nguyen, Klara Nahrstedt and Tarek Abdelzaher, “PDA: Privacy-preserving Data Aggregation in Wireless Sensor Networks”, 2007.

[14] Maarten Ditzel, Caspar Lageweg, Johan Janssen and Arne Theil, “Multi-target Data Aggregation and Tracking in Wireless Sensor Networks”, JOURNAL OF NETWORKS, VOL. 3, NO. 1, JANUARY 2008.

[15] Steffen Peter, Krzysztof Piotrowski and Peter Langendoerfer, “On Concealed Data Aggregation for Wireless Sensor Networks”, 2007.