Adaptive Data Aggregation for Wireless Sensor Networks

Priya Kasirajan and S. Jagannathan
Department of Electrical and Computer Engineering
Missouri University of Science and Technology
sarangap@mst.edu

***Abstract*— Wireless sensor nodes typically have limited processing capabilities and are powered by batteries. The amount of energy expended in transmitting a single data bit would be several orders of magnitude higher when compared to the energy needed for a 32 bit computation. Thus, to maximize network lifetime in real dynamic environments, data transmissions should be minimized without losing vital information. In this position paper, a novel adaptive data aggregation scheme using nonlinear estimation theory is proposed. Satisfactory performance of the proposed compression scheme in the presence of noise, distortion, and quantization errors is demonstrated. The proposed scheme is contrasted with existing compression schemes using various metrics applicable to wireless sensor networks such as energy efficiency, distortion and compression ratio. Simulation and hardware experimental results demonstrate almost 50% energy savings with very low distortion (less than 5%) and overhead. By iteratively applying the proposed scheme at the cluster head nodes, higher energy savings are obtained with a tolerable level of distortion.**

Keywords- Wireless sensor networks, Energy efficiency, Compression, Distortion, Data Aggregation

I. INTRODUCTION

Recent advancements in embedded processing and wireless networking have led to the development of wireless sensor networks (WSN). A WSN is a multi-hop network of nodes each with a short-range radio, limited sensing and on-board processing capability. Sensor nodes are powered by batteries, which determine their short lifetime. This necessitates development of energy efficient network protocols. For many mission critical applications, transmission of sensor data is highly essential. Under these application scenarios, transmitting the raw data directly results in wastage of valuable network resources including energy.

In this paper, we focus on designing *data aggregation* techniques to conserve energy by reducing the amount of data transmitted. In the literature, the data aggregation process usually involves data at select nodes, called Cluster-heads [1], being combined by computing statistical aggregates such as COUNT, SUM, AVERAGE, MAX or MIN and then sending this data to the observer at the base-station node. In [2], a comprehensive survey of data aggregation schemes applicable for different topologies such as flat, hierarchical, cluster-based and grid networks is presented. As presented in [2], data aggregation methods focus only on reducing the overall amount of data by combining data from spatially separated sensors. In many applications such as monitoring of forest fire, humidity in a building, water level etc., sensors repeatedly report data values, and the amount of data transmitted onto the network can be further reduced if we combine multiple data values from a single sensor over time. This task can be achieved through data compression, through which multiple sensor data points are "compressed" and represented by a smaller number of bits in such a way that we can recreate the original data at the base station from those bits. Since more data is represented using fewer bits, energy required by every node to transmit this compressed data is significantly less. Though compression seems computationally intensive, the energy required to transmit an extra bit is several orders of magnitude higher than the energy required for a single 32 bit computation [3]. Thus compression algorithms with a reasonable level of complexity are certainly worth exploring.

While there are many data compression algorithms [4] available, not many [5] are currently used in wireless sensor networks. The popular entropy encoding schemes [6] work best on correlated data. By contrast, the regression techniques [7] perform well when the data is linearly varying. Our objective is to develop a compression scheme that could be applied on any form of deterministic data. Our motivation comes from Adaptive Differential Pulse Code Modulation (ADPCM) [8] for speech coding, which uses correlation between adjacent samples to reduce bit rate and achieve compression. Instead of quantizing a speech signal directly, only the difference between the actual sample and the predicted sample is quantized. If the prediction is accurate, then the difference between the real and predicted speech samples will have a lower variance than the real speech samples, and will be accurately quantized with fewer bits than what would be needed to quantize the original speech samples. At the decoder the quantized difference signal is added to the predicted signal to give the reconstructed speech signal.

However, real world sensor data does not always boast of good correlation as speech signals. A linear predictor will not always be able to handle fast changing data. We propose to represent the data as a nonlinear relationship and use techniques from adaptive estimation theory to obtain an accurate estimate. A novel nonlinear discrete-time estimator is proposed and its performance is demonstrated using Lyapunov theory. It will be shown that the nonlinear adaptive pulse coded modulation-based compression (NADPCMC) indeed results in better compression ratio, reduced distortion, and higher energy efficiency.

The rest of this paper is organized as follows. Section 2 deals with the proposed methodology, discusses theoretical bounds on the stability of the proposed scheme. Results from the simulations and hardware implementation are detailed in Section 3. Section 4 contains the concluding remarks.

II. PROPOSED METHODOLOGY

Fig. 1 depicts the proposed compression-based transmission approach. Two stages are involved at the source node – estimation and quantization. In the first stage, nonlinear adaptive estimation is performed to obtain a close estimate of the current sample based on a few previous samples. In the second stage, the difference between the actual value and the estimated value is quantized. This quantized value is sent to the receiver. At the destination end, a similar estimator is used. A few initial samples are fed directly to the estimator to help it get started. Subsequently, the estimation errors from the first encoder are added to the estimate obtained from the second to reconstruct the signal.

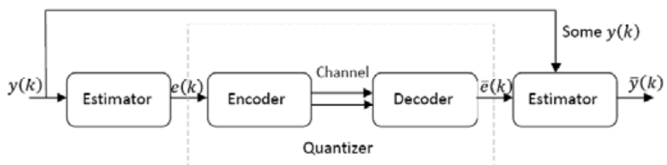

Figure 1. Proposed NADPCMC architecture

A. Adaptive estimation

Adaptive estimation of the data sequence is performed by representing the data as a nonlinear autoregressive moving average sequence as

$$y(k+1) = \theta^T(k)\emptyset(k) + \varepsilon(y) \quad (1)$$

where $\emptyset(k) \epsilon R^{n*1}$ is the basis function, $\theta(k)$ is the unknown parameter vector, and $\varepsilon(y) \epsilon R$ is the reconstruction error such that it is bounded above by $\|\varepsilon(y)\| \leq \varepsilon_N$. The estimated signal is given by

$$\hat{y}(k+1) = \hat{\theta}(k)\emptyset(k) - k_v e(k) \quad (2)$$

where $\hat{\theta}(k)$ is the estimated parameter estimate vector. The estimation error is then given by

$$e(k+1) = y(k+1) - \hat{y}(k+1) \quad (3)$$

Substituting for y and $\hat{y}$ in (3) renders

$$e(k+1) = k_v e(k) + \tilde{\theta}^T(k)\varphi(k) + \varepsilon \quad (4)$$

where the parameter estimation error is defined by

$$\tilde{\theta}^T(k) = \theta(k) - \hat{\theta}(k) \quad (5)$$

Now consider the parameter update as

$$\hat{\theta}(k+1) = \hat{\theta}(k) + \alpha\emptyset(k)e^T(k+1) \quad (6)$$

Here α is called the adaptation gain. The estimation error by using (4) and parameter update equation (6) will be utilized in the next section to show the boundedness of the error. Since the estimation error is related to distortion, subsequently, it will be shown that the distortion is bounded.

B. Analytical Results

The following theorems examine the stability of the estimator and the performance of the method. In the ideal case, when the estimation error $\varepsilon = 0$, equation (4) reduces to

$$e(k+1) = k_v e(k) + \tilde{\theta}^T(k)\varphi(k) \quad (7)$$

where $\bar{e}_i(k) = \tilde{\theta}^T(k)\varphi(k)$. Next the following theorem can be stated in the absence of approximation errors.

Theorem 1 (Estimator-Ideal Performance): Let the proposed nonlinear estimator given in (2) be utilized with the parameter be tuned by (6). In the ideal case with no reconstruction errors and noise present, the estimation error $e(k)$approaches to zero asymptotically while the parameter estimation error vector is bounded.

Proof: Omitted due to space constraints.

In the general non-ideal case, when the reconstruction error is nonzero, the estimation error is as defined in (4) and the following theorem can be stated.

Theorem 1 (Estimator Performance—General Case): Let the hypothesis presented in Theorem 1 hold and if the functional reconstruction error is bounded with $\|\varepsilon(y)\| \leq \varepsilon_N$, then estimation error $e(k)$ is bounded while the parameter errors are also bounded.

The above theorem demonstrated the performance of the estimator. Let us now analyze the overall approach. The proposed scheme (as shown in Fig. 1) involves 2 estimators – one at the transmitter and one at the receiver. The error in estimation from the first is quantized and fed to the second.

The entire NADPCMC scheme can be expressed mathematically as follows: The first estimator continuously produces an estimate $\hat{y}_T$. From (2), the estimated signal can be represented as

$$\hat{y}_T(k+1) = \hat{\theta}_T(k)\emptyset_T(k) - k_v e(k) \quad (8)$$

The error in estimation is obtained from (4) as

$$e(k+1) = y(k+1) - \hat{y}_T(k+1) \quad (9)$$

The parameter $\hat{\theta}_R$ is continuously updated such that the error e which is given by $\hat{y}_T - y$ is minimized. From (6), we have

$$\hat{\theta}_T(k+1) = \hat{\theta}_T(k) + \alpha\emptyset_T(k)e^T(k+1) \quad (10)$$

Then, e(k) is quantized. This stage adds a quantization error ε_Q

$$\bar{e}(k) = Q\big(e(k)\big) = e(k) + \varepsilon_Q \quad (11)$$

The first few samples of $y(k)$ are sent to the receiver side estimator to initialize $\emptyset_R(k)$. This is sufficient for it to start estimating $\hat{y}_R$. As in (2), the estimated signal is given by,

$$\hat{y}_R(k+1) = \hat{\theta}_R(k)\emptyset_R(k) - k_v e_R(k) \quad (12)$$

Now, to obtain the original signal, we simply add the error offset to the estimate. Thus the recovered signal can be expressed as follows

$$\bar{y}(k+1) = \hat{y}_R(k+1) + \bar{e}(k+1) \quad (13)$$

The error in estimation is obtained from (3) as,

$$e_R(k+1) = \bar{y}(k+1) - \hat{y}_R(k+1) = \bar{e}(k+1) \quad (14)$$

The parameter $\hat{\theta}_R$ is updated to account for the error $\bar{e}$ that was incurred at the transmitter side estimator. As in (6), we have

$$\hat{\theta}_R(k+1) = \hat{\theta}_R(k) + \alpha \emptyset_R(k) e_R^{\,T}(k+1) \quad (15)$$

Loss of data can occur at both the estimation and quantization stages. For uniform b bit quantization of a signal that has a dynamic range of (E_{min}, E_{max}), the required step size is given by

$$step\_size = (E_{max} - E_{min})/2^b \quad (16)$$

The quantization error ε_Q is bounded by $step\_size/2$ [4] and thus, resolution of the quantizer has to be chosen based on the permissible level of distortion. Let us now proceed to analyze the maximum distortion introduced by our scheme.

In the proposed scheme, the amount of data lost is dependent on the total error in reconstructing the data at the receiver. The total error after reconstruction is $|y(k) - \bar{y}(k)|$. Now the following theorem can be stated.

Theorem 2 (NADPCMC Distortion): Consider the NADPCMC scheme presented in (2) through (6). If the estimator reconstruction and quantization errors are considered bounded, then the distortion at the destination is bounded. On the other hand in the absence of estimator reconstruction and quantization errors, the distortion is zero.

Remark: The following conclusions can be deduced

1. The total distortion introduced by the proposed scheme is bounded.
2. The distortion is dependent mainly on the quantization error ε_Q.

Theorem 3 (NADPCMC Performance): Consider the NADPCMC scheme presented in (8) through (16). Let us consider $y(k)$ be a N sample vector of x bits each and that the receiver side estimator requires the first K samples to initialize the regression vector. Then the compression ratio, defined as the ratio of the amount of uncompressed data to the amount of compressed data, is greater than one. Moreover, the proposed scheme will render energy savings.

Proof: From (16), the resolution of the Quantizer is given by

$$b = \log_2\left(\frac{E_{max} - E_{min}}{step\_size}\right) \quad (17)$$

The estimation error has a smaller dynamic range compared to the original data. In other words, $(E_{max} - E_{min}) \ll (Y_{max} - Y_{min})$. Thus $b < x$. The compression ratio is then given by

$$Compression\ ratio = \frac{Nx}{Kx + (N-K)b} \quad (18)$$

Since $N >> K$ and $x > b$, the numerator in (18) is greater than the denominator and hence the compression ratio is greater than one. This metric can be used to calculate the amount of energy savings that can be achieved. Assuming that each bit requires the same amount of energy E_b to be transmitted, the amount of energy required to send the uncompressed data is NxE_b and that required to send the uncompressed data is$(Kx + (N-K)b)E_b$. The total energy savings is given by,

$$Energy\ savings\ (\%) = \frac{(N - K)(x-b)}{Nx} * 100 \quad (19)$$

Again, since $N >> K$ and $x > b$, a finite positive energy saving is achieved.

C. Algorithm

The proposed algorithm for the data compression using the nonlinear adaptive estimator can be summarized as follows:

At the Transmitter:

Step 1: Initialize $\emptyset_T(k)$ with first few data points

Step 2: Calculate estimate $\hat{y}_T(k)$ from (8)

Step 3: Calculate estimation error $e(k)$from (9)

Step 4: Calculate parameter $\hat{\theta}_T(k)$ update from (10)

Step 5: Quantize $e(k)$ and send to receiver

Step 6: Update $\emptyset_T(k)$ and repeat from step 2

At the Destination:

Step 1: Initialize $\emptyset_R(k)$ with first few data points

Step 2: Calculate estimate $\hat{y}_R(k)$ from (12)

Step 3: Add $\bar{e}(k)$ and $\hat{y}_R(k)$ to reconstruct data as in (13)

Step 4: Calculate estimation error as difference between reconstructed and estimated signals as in (14)

Step 5: Calculate parameter $\hat{\theta}_R(k)$ update from (15)

Step 6: Update $\emptyset_R(k)$ and repeat from step 2

III. RESULTS AND DISCUSSION

The performance of compression algorithms in general can be measured by using the following metrics. Quality is an important factor for lossy compression algorithms. It is quantified by percentage of distortion which is measured as the absolute difference between the original data and the reconstructed data at the base station. We calculate it as

$$Distortion = \left|\frac{y(k) - \bar{y}(k)}{y(k)}\right| * 100\%$$

Compression ratio is defined as the ratio of the amount of uncompressed data to the amount of compressed data and the additional overhead needed for reconstruction. The compression ratio is defined as

$$Compression\ ratio = \frac{total\ bits\ in\ y(k)}{total\ bits\ in\ e(k) and\ some\ y(k)}$$

Latency of the compression/decompression process also plays a vital role. The number of machine cycles utilized directly impacts the energy expended in computations. Further, applications such as landslide monitoring and fire detection, cannot tolerate delay in the reception of sensor data at the base station. Thus the computation complexity of the algorithm directly affects the applicability of the algorithm. The memory requirement of the algorithm should also be considered while designing or porting compression algorithms for the sensor node. The code footprint and memory usage should be minimal. However, the most important performance metric in the wireless sensor network case is the energy saving provided by the algorithm. It is calculated as the percentage ratio of difference between the energy required to send the uncompressed data and that required for the compressed data to the energy required to send the uncompressed data.

The algorithm was first implemented in MATLAB and was tested with different data sets. Then it was implemented in C to be tested with the Network Simulator (NS2). The topology is shown in Fig 2. The empty circles represent sensor nodes. The shaded ones indicate the cluster-heads (CH) and the striped one is the Base Station (BS). Each CH aggregates the data and routes it to the BS. TCP agents were used for reliable packet delivery. Simulations results performed with different data sets are now presented. Plain quantization of scaled data is also evaluated to analyze the advantages of quantizing the estimation error instead of the original data. The G.722 sub band linear ADPCM (8 bit) is also evaluated to highlight the improvement provided by the nonlinear estimation scheme

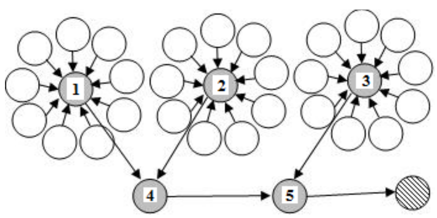

Figure 2. Network topology

A. Synthetic data

This data was generated in MATLAB to resemble data from an explosive sensor. Fig. 3 shows the performance of the estimator. The estimate follows the highly non linear sequence with a minimal delay.

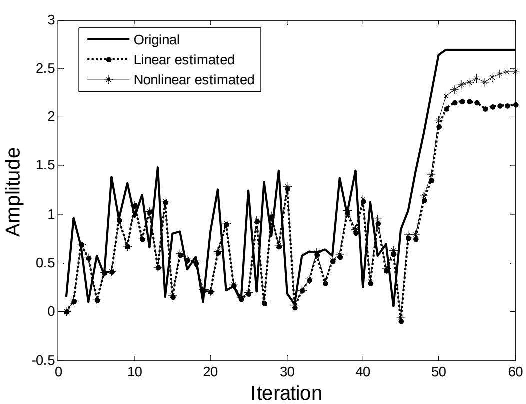

Figure 3. Estimator output.

Fig. 4 illustrates the reconstruction error for different resolutions of the quantizer. The quality improves with the resolution. However, the overhead increases, which in turn causes an increase in the energy expended. These results show that reduced distortion implies higher compression ratio which translates into higher energy efficiency but at the expense of memory and computations.

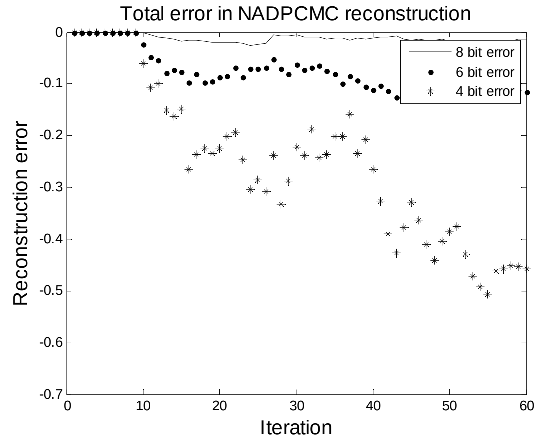

Figure 4. Total reconstruction error with different error encodings

TABLE I. PERFORMANCE METRICS

Method	Compression ratio	Energy savings at nodes	Distortion	Overhead
Scaling and 8 bit quantization	2	50%	2.111%	0
Scaling and 6 bit quantization	2.667	62.5%	13.627%	0
Linear ADPCM	2	50%	18.9%	0
8 bit NADPCMC	1.846	45.83%	1.67%	20 bytes
6 bit NADPCMC	2.342	57.29%	3.64%	20 bytes
4 bit NADPCMC	2.667	62.50%	7.28%	20 bytes

Table I shows a comparison of the proposed performance metrics for different compression schemes on the synthetic data. Direct quantization of scaled data at the nodes provides good compression at the expense of distortion. The linear ADPCM standard loses in terms of distortion. Since the data is very coarse, 8 bit quantization of scaled data is almost as good as 8 bit NADPCMC. However, the proposed scheme offers better performance for lower resolutions of the quantizer. The parameter and regression vectors hold information about 10 previous samples. The overhead of 20 bytes (10 samples of 2 bytes each) corresponds to these first few samples of the original data required by the receiver side estimator.

B. Geophysical data

The proposed algorithm was applied on geophysical data obtained from the Calgary corpus data set [10]. This dataset is widely used in evaluating compression algorithms. The authors mention that the geophysical data is particularly difficult to compress because it contains a wide range of data values. Table 2 summarizes the performance metrics. The distortion was virtually unnoticeable with 8 bit error encoding unlike in direct quantization or linear ADPCM. 10 samples of

2 bytes each were used to initialize the estimator at the receiver. This adds an overhead of 20 bytes per sensor node.

TABLE II. PERFORMANCE METRICS

Method	Compression ratio	Energy savings at nodes	Distortion	Overhead
Scaling and 8 bit quantization	2	50%	4.36%	0
Scaling and 6 bit quantization	2.667	62.5%	13.42%	0
Linear ADPCM	2	50%	35.87%	0
8 bit NADPCMC	2	50%	1.02%	20 bytes
6 bit NADPCMC	2.667	62.5%	4.22%	20 bytes

C. Performance as an aggregation scheme

This section examines the performance of the NADPCMC scheme when it is applied at a cluster-head for aggregation instead of compression performed at the source node only.

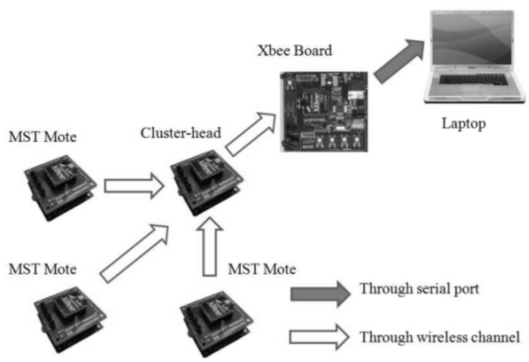

Figure 5. Hardware architecture

Fig. 5 shows the topology considered. The NADPCMC algorithm was implemented on a low-cost, fan-less single board computer called Beagle Board [11] running Ubuntu Linux and interfaced with Missouri S&T F1 motes. These motes provide a common platform for sensing, networking and data processing. The platform consists of an 8051 processor and an 802.15.4 (XBee) radio with micro Smart Digital (SD™) flash storage, USB and RS-232 connectivity and an assortment of sensors. More information can be found in [12]. The motes form a network and use a static routing protocol to deliver data over multiple hops to a base station.

Initially, uncompressed synthetic data (generated using MATLAB) is packetized and sent over the network and the energy expended is calculated. Then the data is compressed online using the proposed algorithm, packetized and then routed to the base station. Once again the expended energy is calculated. It is important to note that the type of routing protocol is not relevant since proposed aggregation scheme is independent of routing. The data is recovered at the base station and the performance metrics are applied. The cluster-head aggregates the data received from three nodes and forward it to the base station. 8 bit NADPCMC was performed at the source nodes and 6 bit NADPCMC was applied at the Cluster-heads.

With compression only at the source level, all nodes reported an energy savings of 45.83% when compared with no compression. The average compression ratio at the source nodes was 1.846. By repeating compression implemented on the MST Motes at the Cluster-head, the amount of data and the energy expended in transmitting it are reduced further, however at the expense of distortion. The compression ratio at the Cluster-head level was 2.526 amounting to an energy savings of 60.42%. There is an overhead of 20 bytes added by the NADPCMC scheme for every level of aggregation. The total distortion has increased from 1.67% to 4.60% with an additional level of aggregation. These results clearly demonstrate that with repeated compression, the distortion increases while energy savings and compression improve.

The encoding scheme consisted of 7050 floating point operations per second. Thus, the OMAP 3530 processor on the Beagleboard expends 1.224µJ per second on compression. This is very small compared to the energy required to send the uncompressed data over the network (20.32µJ per second).

To test scalability, networks with 50 to 250 nodes were created in NS2 and multiple data flows were applied. All source nodes use 8 bit NADPCMC. First level cluster-heads use 6 bit NADPCMC and are one hop away from the sensor nodes. A second level of aggregation with 4 bit NADPCMC is performed at a cluster-head that is closest to the base station. Fig. 6 shows the compression ratio at the first cluster-head level. Each application of the NADPCMC scheme adds an overhead of 20 bytes. With increase in the network size, more clusters were created. This led to an increase in the number of times the compression scheme was applied. Since there is a small overhead associated with the scheme, the compression ratio decreases slightly with increase in network size for the same number of flows. Also with increase in number of flows, the amount of data (N) flowing through the network is higher and the overhead (K) appears smaller. This leads to a slight increase in compression ratio.

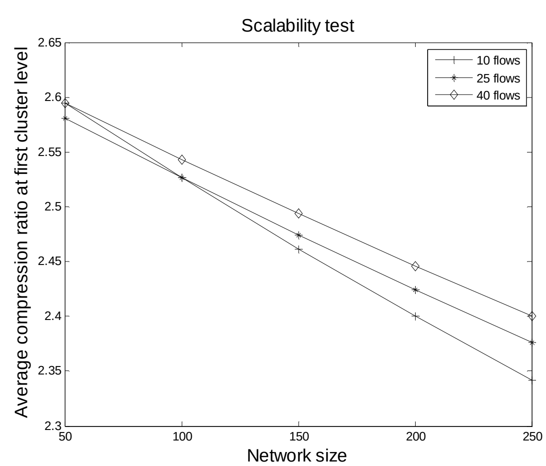

Figure 6. Average compression ratio at first cluster-head level

The distortion suffered by each data flow was constant and independent of the traffic scenario and network size. This indicates that the performance of the NADPCMC scheme is only dependent on the number of aggregation levels and not on the network size.

IV. Conclusions

In this position paper, a novel data aggregation for WSN is introduced by using the compression scheme based on adaptive estimation and quantization. Theoretical bounds on estimation error are derived and shown to be bounded when reconstruction error and quantization errors are bounded. Subsequently, distortion has been proven to be bounded and small. The scheme was tested using multiple data sets. This proposed scheme is shown to offer energy savings of approximately 50% at each source node at the cost of around 2-3% distortion. Though direct quantization works well on coarse data, it fails with data with fine resolution. However the NADPCMC scheme works fairly well on all these data sets. Then data aggregation through iterative compression was examined. Hardware and simulation results demonstrate that aggregation can improve the over-all energy savings with a small level of distortion which depends on the resolution of the Quantizer and the number of aggregation levels and not on the network size.

References

[1] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks", *Proc. Hawaii Intl. Conf. System Sciences*, pp. 3005-3014, Jan 2000.

[2] R Rajagopalan and P K Varshney, "Data-Aggregation Techniques in Sensor Networks – A Survey", *IEEE Communications Surveys & Tutorials*, vol. 8, no. 4, pp. 48-63, 2006.

[3] C M. Sadler and M Martonosi, "Data Compression Algorithms for Energy-Constrained Devices in Delay Tolerant Networks", *Proc. of the 4th Int'l conference on Embedded networked sensor systems*, pp. 265-278, 2006.

[4] K Sayood, "*Introduction to Data compression*", Third edition, Morgan Kaufmann Series in Multimedia Information and Systems, Dec 2005.

[5] Kimura N and Latifi S, "A survey on data compression in wireless sensor networks", *Int'l Conference on Information Technology: Coding and Computing*, vol. 2, pp. 8 -13, 2005.

[6] C Alippi, R Camplani, and C Galperti, "Lossless Compression Techniques in Wireless Sensor Networks: Monitoring Microacoustic Emissions", *Int'l Workshop on Robotic and Sensors Envs*, pp. 1-5, Oct 2007.

[7] A Deligiannakis, YKotidis and N Roussopoulos, "Compressing Historical Information in Sensor Networks," *Proc. of the 2004 ACM SIGMOD Int'l conference on Management of data*, pp. 527-538, 2004.

[8] P. Cummiskey, N. S. Jayant, and J. L. Flanagan, "Adaptive quantization in differential PCM coding of speech", *Bell Syst. Tech. J.*, vol. 52, pp. 1105-1118, Sept. 1973.

[9] S.Jagannathan, "*Wireless Ad hoc and Sensor Networks: Protocols, Performance and Control"*, CRC Press, 2007.

[10] Geophysical data from Calgary corpus – available online - http://corpus.canterbury.ac.nz/descriptions/#calgary

[11] Beagleboard Technical Specifications – available online - http://beagleboard.org/static/BBSRM_latest.pdf

[12] J Fonda, S Watkins, S. Jagannathan, and M Zawodniok, "Embeddable Sensor Mote for Structural Monitoring", *SPIE 15th Annual Int'l Symposium on Smart Structures/NDE: Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2008*, vol. 6932, pp. 69322V.1-69322V.11, 2008.