An Adaptive Distributed Data Aggregation based on RCPC for Wireless Sensor Networks*

Guogang Hua, Chang Wen Chen
Dept. of Electrical and Computer Engineering,
Florida Institute of Technology, Melbourne, FL 32901

ABSTRACT

One of the most important design issues in wireless sensor networks is energy efficiency. Data aggregation has significant impact on the energy efficiency of the wireless sensor networks. With massive deployment of sensor nodes and limited energy supply, data aggregation has been considered as an essential paradigm for data collection in sensor networks. Recently, distributed source coding has been demonstrated to possess several advantages in data aggregation for wireless sensor networks. Distributed source coding is able to encode sensor data with lower bit rate without direct communication among sensor nodes. To ensure reliable and high throughput transmission with the aggregated data, we proposed in this research a progressive transmission and decoding of Rate-Compatible Punctured Convolutional (RCPC) coded data aggregation with distributed source coding. Our proposed 1/2 RSC codes with Viterbi algorithm for distributed source coding are able to guarantee that, even without any correlation between the data, the decoder can always decode the data correctly without wasting energy. The proposed approach achieves two aspects in adaptive data aggregation for wireless sensor networks. First, the RCPC coding facilitates adaptive compression corresponding to the correlation of the sensor data. When the data correlation is high, higher compression ration can be achieved. Otherwise, lower compression ratio will be achieved. Second, the data aggregation is adaptively accumulated. There is no waste of energy in the transmission; even there is no correlation among the data, the energy consumed is at the same level as raw data collection. Experimental results have shown that the proposed distributed data aggregation based on RCPC is able to achieve high throughput and low energy consumption data collection for wireless sensor networks

Keywords: Wireless sensor networks, data aggregation, adaptive, distributed source coding, RCPC

1. CONCLUSION

Networked micro-sensors technology is a key to numerous contemporary surveillance and monitoring applications [1]. In recent years, research in sensor networks has been undergoing a quiet revolution, promising to have a significant impact throughout society that could quite possibly dwarf previous milestones in the information revolution. The MIT Technology Review ranked wireless sensor networks that consist of many tiny, low-power and cheap wireless sensors as the number one emerging technology [2]. As a result, wireless sensor networks attract more and more attentions from various research communities.

Usually, a sensor network consists of one or more "sinks" that gather information of interest. The sensors in the network act as "sources" that detect environmental events which are usually called the readings of sensors, and send these readings to the sinks. The information of interest could be readings from all sensors or readings from only a subset of the sensors. For example, in a river water quality monitoring system, a number of sensors are deployed along the river, and the sink is located in the monitoring center which may be far away from the river. All sensors can send their information periodically to the sink or the sink maybe sends queries to all sensors or some sensors in a particular region to gather information of interest. Those sensors receiving queries send back their readings to the sink in response to the queries.

The most important challenge in design wireless sensor networks is that the key resource, namely energy supply, is very limited. Wireless sensor networks consist of battery-operated sensing devices with computing, data processing and

---

* This research work has been supported by Florida Institute of Technology Allen S. Henry Endowment Fund.
The email and phone correspondences for the authors: Guogang Hua: ghua@fit.edu; phone 1 321 674 7055; Chang Wen Chen: cchen@fit.edu; phone 1 321 674 8769

Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense V, edited by Edward M. Carapezza, Proc. of SPIE Vol. 6201, 62010D, (2006)
0277-786X/06/$15 · doi: 10.1117/12.666548

communicating components. Usually, the lifetime of the sensor networks is expected to be as long as several years. However, battery recharging is often extremely difficult or even impossible. Therefore, to prolong the system lifetime, we need to reduce the unnecessary energy consumption in the networks as much as possible. Most current research has been focused on issues such as energy efficient MAC and routing protocols and a large number of protocols [3-9] have already been proposed.

However, in wireless sensor networks, multiple nodes collectively perform the sensing task and communicate the sensing results. In many cases, sensor nodes are densely deployed and sensing the same physical phenomena. Hence, the information from neighboring sensor nodes is greatly correlated. This implies that the data can be compressed in the networks by exploring the correlation. Actually, most wireless sensor networks are application-orientated, it does not need to send every bit in network (sensed or received for forwarding) as long as the data gathering nodes can extract the correct information from the bitstream they received. That is to say, by exploiting the correlation among data, the nodes can process the information they sensed or received before transmitting in order to reduce the data flow, so as to reduce the energy consumption in communication in the network. This leads to an important area of research in sensor networks: information aggregation. This area of research has attracted more attention recently [2, 10-13]. The basic idea of information aggregation is to combine information from different sources so as to remove redundancy within data and to reduce the transmission and reception via wireless communication, a major source of energy consumption in the wireless sensor networks.

Distributed source coding has been demonstrated to be a good choice for data aggregation in wireless sensor networks since it can exploit the correlation between the data from neighboring nodes without directly communication among them [13,14] . The basic idea of distributed source coding in wireless sensor networks is to take advantage of the correlation between data and using correlated sensor's data as side information to decode sensor's encoded data. The more correlated the data with the side information; the few bits are needed to encode the data. One problem in distributed source coding for wireless sensor networks is that it is difficult to decide the correlation between data without direct communication among the sensor nodes. Another problem is that distributed source coding cannot guarantee zero decoding error. In some existing schemes, retransmission of raw data is required when a decoding error is detected. In this case, energy is wasted because the previous transmitted data is discarded.

In [13], the authors introduced a framework of using distributed source coding in wireless sensor networks which enable highly effective and efficient compression across a sensor network without the need to establish inter-node communication. However, no in-depth analysis for practical applications has been addressed. In [9], the authors proposed a distributed estimation algorithm that can be applied to a large class of data aggregation problems in wireless sensor networks. However, the aggregation algorithm in sensor networks is actually an energy accuracy trade-off, which means that the algorithm may not be able to gather all data correctly. The error in data gathering may be tolerated for some applications while may not be acceptable for some critical applications in defense and medical fields. In [10], the authors proposed a distributed and adaptive signal processing approach to reducing energy consumption in sensor networks. First, an adaptive correlation-tracking algorithm can continuously track the amount of correlation that exists between the sensor nodes; then, based on the amount of correlation, a distributed source coding algorithm based on tree based codebook partition was used to code each node's reading and send to data gathering node. The advantage of the algorithm is that the encoding and decoding algorithms are both very simple. The disadvantage is that the probability of decoding error is high. As demonstrated in [15], the distributed source coding based on turbo code outperforms the distributed source coding based on trellis codes. Another problem about this scheme is that tracking the amount of correlation takes time, and the correlation is dynamic in the whole data aggregation process.

More recently, we have developed a practical distributed source coding in wireless sensor networks based on convolutional code and turbo code. This scheme is able to ensure that the data can be received correctly and energy consumption in the networks can be reduced [14]. In that research, we assumed that the correlation among the data was constant, which may not be true in practice. In this previous work, retransmission of raw data is required when a decoding error is detected. In this case, energy is wasted because the previous transmitted data is discarded.

In this paper, we proposed an adaptive and distributed data aggregation strategy for wireless sensor networks based on Rate-Compatible Punctured Convolutional (RCPC) [16]. We shall adopt the improved Viterbi Algorithm as shown in [14] which is specifically designed for distributed source coding (VA-DSC). We will show that when the algorithm is applied to a 1/2 RSC encoder, and all the parity bits of the output can be received successfully, no matter how many systematic bits are corrupted, the decoder can always decode the data without error. However, if we use the 1/2 RSC directly, no compression effect will be achieved.

The proposed approach in this paper achieves two aspects of adaptive data aggregation for wireless sensor networks. First, the RCPC coding facilitates adaptive compression corresponding to the correlation of the sensor data. When the data correlation is high, higher compression ration can be achieved. Otherwise, lower compression ratio will be achieved. Furthermore, this adaptive data aggregation does not need to predict the correlation among data. This will save additional energy may be needed for computing the correlation of the data. Second, the data aggregation is adaptively accumulated. In the case of retransmission, the previously received data are used to combine with the newly retransmitted data for the decoding. There is no waste of energy in the transmission; even there is no correlation among the data, the energy consumed is at the same level as raw data collection. Experimental results have shown that the proposed distributed data aggregation based on RCPC is able to achieve high throughput and low energy consumption data collection for wireless sensor networks.

2. 1/2 RSC FOR DISTRIBUTED SOURCE CODING

To perform data aggregation in wireless sensor networks based on distributed source coding, a practical construction of distributed source coding is needed to take full advantage of its benefits. As is well known, in distributed source coding, the side information Y is correlated with the information to be transmitted, X. Therefore, we can view the side information Y as an output of X being transmitted through a channel and corrupted by the channel noise. Our task is correct the corrupted Y to obtain X. Error control code has the ability to correct transmitted signal corrupted by channel noise. A distributed source coding based on convolutional codes shown as in Fig. 1 was proposed in [17]. In this case, the encoder is a $N/(N+1)$ recursive systematic convolutional (RSC) encoder. Systematic convolutional code means that the output of the encoder includes exactly the input bits and the parity bits. For example, for a 2/3 systematic convolutional encoder, if the input is 00, the output should be 000 or 001. The first two bits are exactly the input, and the last bit is parity bit. In this case, the system will only transmit the parity bits, so the compression ratio is $1/N$. At the decoder, the corresponding bits from the side information Y are inserted into the received parity bits to reconstruct the corrupted version of the encoder output. The Viterbi Algorithm is used to decode reconstructed bit stream to get $\hat{X}$, an estimation of X.

To achieve high compression efficiency, the number N should be much larger than 1. But in [16], the author proposed a method to puncture a low rate $1/N$ convolutional code periodically with period P to obtain a family of codes with rate $P/(P+l)$ where l can be varied between 1 and $(N-1)P$. The decoding algorithm, Viterbi algorithm for high-rate codes can be significantly simplified by puncturing them for low rate $1/N$ codes [18]. In distributed source coding, to achieve compression, the punctured code rate of the convolutional code $P/(P+l)$ must be larger than 1/2.

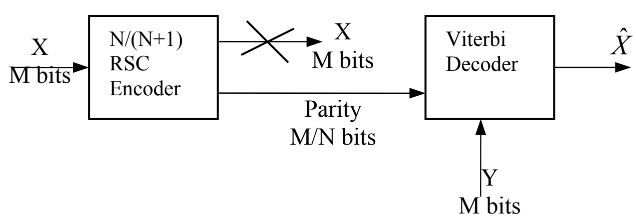

Fig.1 Distributed coding using convolutional code

In our previous work [14], we proposed an improved Viterbi Algorithm for the distributed source coding (VA-DSC) to take the advantage of unchanged parity bits in encoder and decoder. In the general Viterbi Algorithm, the state transition covers all possible state change. We showed in [14] that there is a possibility that the algorithm chooses a wrong survival path and introduces more errors. If we know that some of the transit ion is impossible because of certain constraints, we may delete some impossible paths. In this case, the probability of choosing a wrong path is decreased. In a 1/2 RSC, from each state, there are two possible transitions to the next state. However, if the improved Viterbi Algorithm for the distributed source coding (VA-DSC) is applied to 1/2 RSC codes, we know that the parity bits remain unchanged and we conclude that one of the transition is impossible. Therefore, for each state, there is only one transition to next stage. That is to say that there is only one path from the beginning to the end as shown in Fig. 2. Hence, no matter how many systematic bits are corrupted, the decoder can always decode the data without any error. This is a nice property that guarantees that the maximum number of bits needed to decode the data successfully is exactly the same as the number of raw data in our proposed data aggregation strategy based on distributed source coding. This extreme case occurs when there is no correlation among the data.

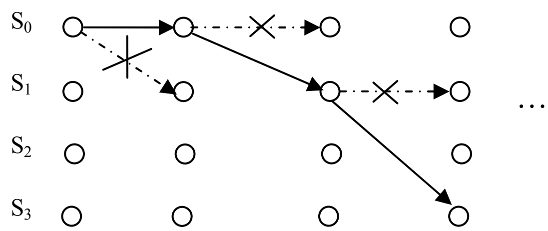

Fig. 2 State transition of the 1/2 RSC coder if parity bits are known

3. RCPC FOR DISTRIBUTED SOURCE CODING

Fig. 3 gives the diagram of Rate-Compatible Punctured Convolutional (RCPC) codes for distributed source coding. The output of the 1/2 RSC encoder can be divided into two parts: systematic bits and parity bits. When used in distributed source coding, the systematic bits outputs of the RSC encoder are discarded, and the parity bits are punctured to achieve different coding rates. First, only a small part of the parity bits are sent, if the data can be decoded correctly from the transmitted data, we achieve a high compression ratio. Otherwise, more parity bits are sent. This new parity bits will be combined the parity bits that have already been received to decode the data. For example, only a small part of the parity bits (say 1/4 of parity bits) are sent. If it can be decoded successfully, we achieve a compression ratio of 1/4; otherwise, more bits are sent (say another 1/12 of parity bits). Now if it can be decoded successfully, we achieve a compression ratio of 1/3; otherwise, more bits are sent (say another 1/6 of parity bits). At this time, the decoder has 1/2 of the parity bits. If it can be decoded successfully, we get a compression ratio of 1/2. Theoretically, the process can continue until the worst case that all parity bits are sent. This will be exactly the same amount as the raw data. In this worst case scenario where there is no correlation among the data, the number of bits needs to be sent to guarantee zero decoding error is the same as the raw data. It is clear that the compression ratio is adaptive to the correlation. When the correlation is higher, a higher compression ration can be achieved; otherwise, a lower compression ratio is achieved. Because of recursive transmission, this adaptive compression scheme does not need to know the correlation in advance.

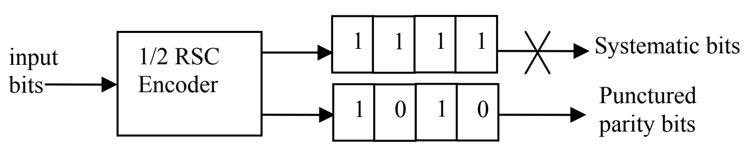

Fig. 3 RCPC for distributed source coding

In our research, we have designed four compression ratios, 1/4, 1/3, 1/2 and 1. To achieve the four compression ratios, we proposed a punctured period of 12 as in Figure 4. First, 1st, 5th, and 9th parity bits of each period are sent to achieve a compression ration of 1/4. If more bits are needed to decode the information, the 7th parity bits of each puncture period are sent to achieve a compression ratio of 1/3. If still more bits are needed, the 3rd and 11th parity bits of each period are sent to achieve a compression ratio of 1/2. If it still can not decode successfully, all the remaining bits are sent, which guarantees the correct decoding. However, we achieve no compression in this extreme case. This is the worst case scenario, since data aggregation based on distributed source coding achieves compression based on the correlation among data. When there is no correlation, the best we can do is to send the same amount of data as the encoded data.

Fig. 4 Punctured period

4. EXPERIMENTAL RESULTS

To demonstrate the performance of our proposed data aggregation scheme for wireless sensor networks, three performance measures are present in this section: (1) the decoding BER performance of different code rate RCPCs with period 12 for distributed source coding, (2) the energy consumption performance in chain-type wireless sensor networks, (3) the energy the energy consumption performances in LEACH architecture.

4.1 Decoding BER performance of RCPC for distributed source coding

The RCPC codes adopted in this research are all derived from 1/2 convolutional code. One benefit of using 1/2 RSC is to guarantee zero decoding error rate. Another benefit of using 1/2 convolutional code is that 1/2 convolutional code simplifies the Viterbi decoder for high-rate codes. To apply RCPC to distributed source coding, the mother code must be recursive systematic convolutional (RSC) codes. Our simulations show that punctured RSC has similar decoding performance with the non-RSC with similar code rate and memory size. However, if the VA-DSC we proposed is applied to the punctured RSC, it performs much better than the traditional RSC codes. Figure 5 give the decoding performance of the proposed period 12 RSC codes with Viterbi Algorithm for distributed source coding (VA-DSC) and the high-rate RCPC from [19], where “HRCPC” represents the high-rate RCPC adopted from [19], “M” is the memory size and “R” is the code rate. It shows that the proposed codes with VA-DSC perform significantly better than the high-rate RCPC with the same rate and memory size.

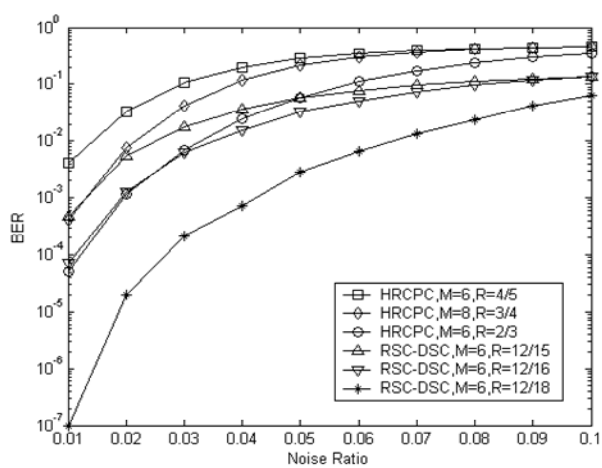

Fig.5. Period 12 RSC RCPC decoding performance

4.2 Energy consumption in chain-type wireless sensor networks

When the scheme is applied to chain-type wireless sensor networks [14], we get the energy consumption per bit received at CHN nodes as figure 6. The 2/3 RSC coding with VA-DSC is the method adopted in [14]. We can see that when the correlation is low, the energy consumption is even more than that of uncoded data transmission. However, for the new scheme proposed in this research, when the correlation is very low, the energy consumption is about the same as that of uncoded data transmission. When the correlation becomes higher, the energy consumptions in both schemes decrease. However, RCPC always consume less energy than scheme developed in [14]. When the correlation is very high (when C-SNR is 20dB), the energy consumption in RCPC is about 1/4 of that of uncoded data transmission and the energy consumption in 2/3 RSC is about 1/2 of that of uncoded data transmission. This is because, at this time, the correlation is so high that almost all data can be decoded correctly in the fist transmission. For the new scheme, the best compression ratio is 1/4, while in the scheme developed in [14], the best compression ratio is 1/2. From these results, it clear that the new scheme proposed in this research is adaptive to the data correlation.

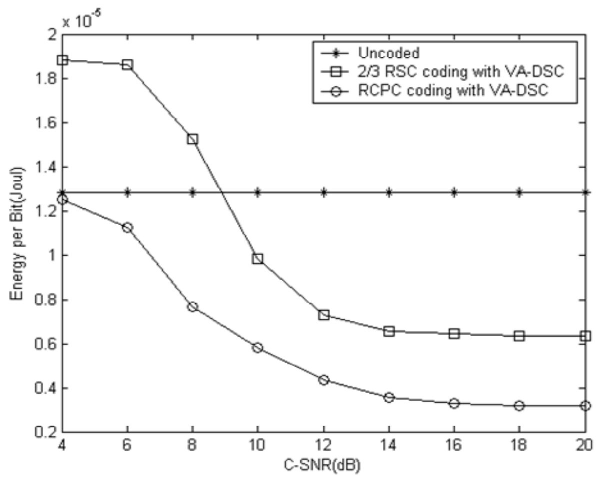

Fig.6. Energy consumption in Chain-type wireless sensor

4.3 Energy consumption performances in LEACH architecture

LEACH [19] is a protocol architecture where computation is performed locally to reduce the amount of transmitted data and save energy in wireless sensor networks. We combine our data aggregation with the LEACH architecture to demonstrate the performance of our proposed data aggregation strategy in general wireless sensor networks. In each round, the set-up phase is the same with the set-up phase in [19]. However, in the steady-state phase, the proposed data aggregation strategy based on RCPC for distributed source coding is applied.

In our simulation, the data of each node is generated as Gaussian random sequence. The data of cluster head X_{chn} is a zero mean unit variance Gaussian random sequence, and the data of other nodes in the cluster is generated as:

$$X_i = X_{chn} + \sigma_i Z_i$$

where Z_i is another zero mean unit variance Gaussian random sequence, σ_i is a random number based on CSNR, and $CSNR = 10\log_{10}(\frac{1}{\sigma^2})$. Before actually encoding and decoding, the sequences are quantized using 4-level Lloyd-Max scalar quantizer.

In our simulation, two types of CSNR are conducted. First, CSNR is a uniform random number from 0 to 20, which means that the correlation among data is from very low to very high. Second, CSNR is a uniform random number from

10 to 20, which means high correlation among all data. All the simulations are conducted under the same condition: the same nodes' position and same initial energy. For LEACH, no data aggregation is carried and therefore no data aggregation energy is consumed. That is, $E_{DA} = 0$. For LEACH-DSC, $E_{DA} = 5nJ / bit / signal$. That is the energy for decoding one bit data. We use the same energy model of [19]. 100 nodes were randomly distributed between (0, 0) and (100,100), and the BS is located at (50,175). Each round last for 20s, in each second, each node generates 960 bits data, and the cluster heads collect data one time. Part of the parity bits of output of the encoder with the checksum of original data is sent.

Fig.7 gives the number of nodes that remain alive in LEACH architecture over time. The life of network in LEACH-RAW is much shorter than that in LEACH-DSC. When the correlation among the data is higher (10-20dB), the life of the network is longer. This is because in high correlation cases, fewer data are transmitted each round, and the energy consumption is lower.

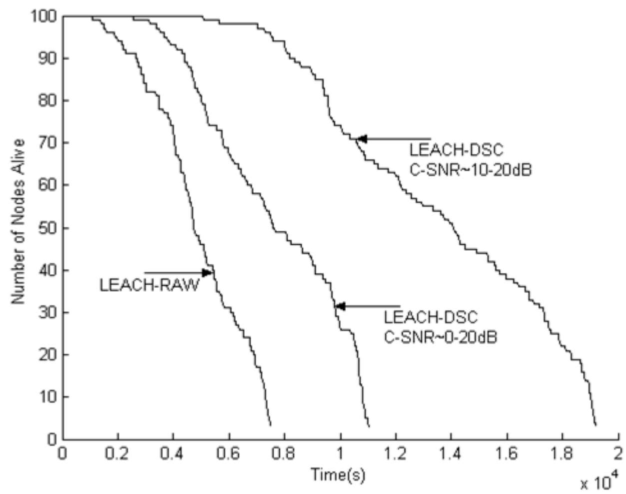

Fig.7. Number of nodes alive in LEACH architecture networks

Fig.8 shows the total energy dissipation in LEACH architecture over time. We can see that the nodes consume energy at a faster rate in LEACH-RAW. While in LEACH-DSC, the higher correlation case consumes energy slower, because in this case, few data are transmitted. Here few transmitted data does not mean that the base station get few information. Actually, to get same information in base station, the LEACH-DSC transmits fewer bits than LEACH-RAW, which can be viewed from Fig. 9. Fig. 9 gives the number of data received at BS (after decoding) over total energy consumed in the network. Given the same energy, the LEACH-RAW collects fewest data, while the LEACH-DSC (CSNR~10-20dB) collects the most data.

5. CONCLUSION

In this paper, we have described an adaptive distributed data aggregation strategy based on RCPC for wireless sensor networks. This strategy takes the advantage of our previously developed Viterbi Algorithm for distributed source coding as it applied to 1/2 RSC and RCPC to enable us to exploit the correlation among the sensors' data as much as possible without explicit estimation of the data correlation. When the data correlation is high, a higher compression ratio can be achieved. Otherwise, lower compression ratio can still be achieved. Furthermore, the data aggregation is adaptively accumulated. There is no waste of energy in such an adaptive transmission. Even when there is no correlation among the data at all, the energy consumption of RCPC based scheme is in the same as transmitting raw data. Several experiments have been conducted to verify the performance of the proposed data aggregation scheme. Experimental results have shown that the proposed distributed data aggregation based on RCPC is able to achieve high throughput and low energy consumption data collection for wireless sensor networks.

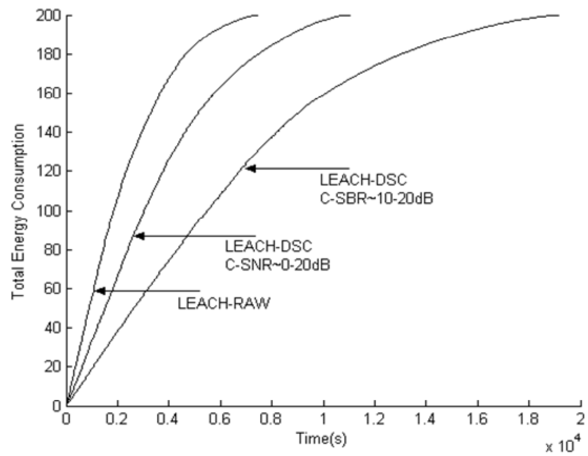

Fig.8. Total Energy dissipation in LEACH architecture

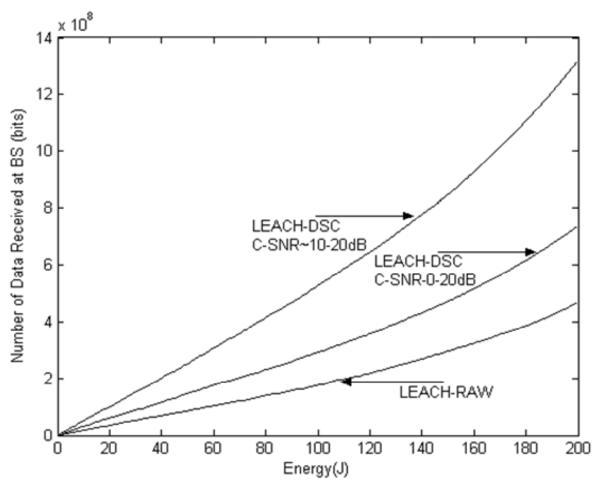

Fig.9. Total amount of data received at BS over total energy consumption in the network

REFERENCES

[1] C. Chong, S. P. Kumar, "Sensor Networks: Evolution, Opportunities, and Challenges," pp. 1247–1256, *Proceedings of the IEEE*, vol. 91, no. 8, Aug. 2003.

[2] Z. Xiong, A. Liveris, and S. Cheng, "Distributed Source Coding for Sensor Networks," *IEEE Signal Processing Magazine,* vol. 21, Sept. 2004.

[3] K. Sohrabi, J. Gao, V. Ailawadhi, and G.J. Pottie. "Protocols for self-organization of a wireless sensor network,"*IEEE Personal Communications,* Vol. 7, No. 5*, Oct. 2000.*

[4] G. Kulkarni, C. Schurgers, and M. Srivastava, "Dynamic link labels for energy efficient MAC headers in wireless sensor networks," *Proceedings of IEEE, Volume: 2*, 12-14 June 2002.

[5] T.V. Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," *The First ACM Conference on Embedded Networked Sensor Systems (Sensys'03)*, Los Angeles, CA, USA. November, 2003.

[6] G. Lu, B. Krishnamachari, and C. Raghavendra, “An adaptive energy-efficient and low-latency MAC for data gathering in sensor networks,” *The 4th International Workshop on Algorithms for Wireless, Mobile, Ad Hoc and Sensor Networks (WMAN 04)*, April 2004.
[7] W. Ye, J. Heidemann and D. Estrin, “Medium access control with coordinated adaptive sleeping for wireless sensor networks,” *IEEE/ACM Trans. on Networks,* Vol. 12, No. 3, June 2004.
[8] M. Ettus, “System capacity, latency, and power consumption in multihop-routed SS-CDMA wireless networks,” in Proc. Radio and Wireless Conf. (RAWCON), Colorado Springs, CO, Aug. 1998, pp. 55-58.
[9] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” *IEEE/ACM Trans. on Wireless Communications,* vol. 1, No. 4 ,pp.660-670, Oct. 2002.
[10] J. Chou, D. Petrovic, and K. Ramchandran, “A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks,” *Proc. IEEE INFOCOM,* March 2003.
[11] A. Boulis, S. Ganeriwal, M. B. Srivastava, “Aggregation in sensor networks: an energy-accuracy tradeoff,” *Elsevier Ad-hoc Networks Journal (special issue on sensor network protocols and applications)*, 2003.
[12] S. S. Pradhan and K. Ramchandran, “Distributed source coding using syndromes (DISCUS): Design and construction,” *Proceedings of the Data Compression Conference (DCC)*, March 1999.
[13] S. S. Pradhan, J. Kusuma, and K. Ramchandran, “Distributed compression in a dense microsensor network,” *IEEE Signal Processing Magazine*, vol. 19, issue 2, pp.51-60, March, 2002.
[14] G. Hua and C.W. Chen, "Distributed Source Coding in Wireless Sensor Networks", *The Second International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine 2005)*, Orlando, USA, August, 2005
[15] A. Aaron and B. Girod, "Compression with side information using turbo codes," *Proc. IEEE Data Compression Conference, DCC-2002*, Snowbird, UT, April 2002.
[16] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Communications.*, vol. 36, no. 4, pp. 389--400, Apr. 1998.
[17] A. Aaron, “Distributed source coding,” [http://www.stanford.edu/%7Eamaaron/dsc/index.htm]
[18] Y.Yasuda, K. Kashiki and Y. Hirata, "High-Rate Punctured Convolutional Codes for Soft Decision Viterbi Decoding", *IEEE Transactions on Communications*, No. 32, March, 1984, pp. 315 – 319.
[19] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An Application-Specific Protocol Architecture for Wireless Microsensor Networks,” *IEEE Tran. on Wireless Comm. Vol. 1, No. 4, October 2002.*