Collaborative Broadcasting and Compression in Cluster-based Wireless Sensor Networks

ANH TUAN HOANG and MEHUL MOTANI†
Electrical and Computer Engineering Department,
Institute for Infocomm Research,
National University of Singapore, Singapore 119260.
Email: {engp1629, motani}@nus.edu.sg

Achieving energy efficiency to prolong the network lifetime is an important design criterion for wireless sensor networks. In this paper, we propose a novel approach that exploits the broadcast nature of the wireless medium for energy conservation in spatially correlated wireless sensor networks. Since wireless transmission is inherently broadcast, when one sensor node transmits, other nodes in its coverage area can receive the transmitted data. When data collected by different sensors are correlated, each sensor can utilize the data it overhears from other sensors to compress its own data and conserve energy in its own transmissions. We apply this idea to a class of cluster-based wireless sensor networks in which each sensing node transmits collected data directly to its cluster head using time division multiple access (TDMA). We formulate the problem in which sensors in each cluster collaborate their transmitting, receiving, and compressing activities to optimize their lifetimes. We show that this lifetime optimization problem can be solved by a sequence of linear programming problems. We also propose a heuristic scheme, which has low complexity and achieves near optimal performance. Important characteristics of wireless sensor networks such as node startup cost and packet loss due to transmission errors are also considered. Numerical results show that by exploiting the broadcast nature of the wireless medium, our control schemes achieve significant improvement in the sensors' lifetimes.

Categories and Subject Descriptors: E.4 [**CODING AND INFORMATION THEORY**]: —*Data compaction and compression*; C.2.1 [**COMPUTER-COMMUNICATION NETWORKS**]: Network Architecture and Design—*Packet networks*

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Sensor networks, clustering, spatial correlation, data gathering, broadcast communication, data compression, lifetime optimization.

†The corresponding author is MEHUL MOTANI. He can be contacted at 10 Kent Ridge Crescent, National University of Singapore, Electrical & Computer Eng, Singapore 119260; Tel: (65) 6874 6918, Fax: (65) 6779 1103, Email: motani@nus.edu.sg.

The material in this paper was presented in part at the Second European Workshop on Wireless Sensor Networks (EWSN), January 2005 and at the International Conference on Communications (ICC), May 2005.

1. INTRODUCTION

Recent advances in wireless communication and microelectronics have enabled the possibility of wireless sensor networks (WSNs) [Akyildiz et al. 2002], which can consist of hundreds or even thousands of low-cost, low-power, and small-in-size sensors. As these cheap and tiny sensors can only be equipped with small batteries, and in many applications, battery recharging/replacing is not desirable, achieving energy-efficiency to increase sensors' lifetimes is an important design criterion for WSNs.

In many sensing networks, a high degree of spatial correlation exists among the readings of different sensors. By allowing nodes to cooperate to carry out joint data compression and aggregation, the amount of data communicated within the network can be reduced. This can help conserve energy and extend sensors' lifetimes.

This paper deals with removing the redundancy due to spatial correlation among nodes in WSNs. The novelty of our work lies in the fact that we exploit the inherent broadcast nature of the wireless medium for nodes to share and jointly compress their data. The core idea is that, when one node broadcasts data, other nodes within the transmission range can receive and utilize this data in compressing their own data.

We note that most of the works concerning data compression and aggregation in WSNs adopt a common model for the wireless medium, in which a wireless channel is abstracted as a single point-to-point link between a pair of nodes, e.g., [Pradhan and Ramchandran 1999; Chou et al. 2003; Scaglione and Servetto 2002; Cristescu et al. 2004; Intanagonwiwat et al. 2000; Krishnamachari et al. 2002; Goel and Estrin 2003]. This point-to-point link model, while simplifying design and analysis, ignores important advantages that come with the inherent broadcast nature of the wireless medium. We contend that the wireless broadcast property offers nodes in a WSN much more freedom in carrying out joint data compression and achieving energy efficiency.

To illustrate our point, let us consider a simple system of four wireless sensor nodes (A), (B), (C), and (D) depicted in Figs. 1 and 2. Nodes (A) and (B) need to transmit collected data to (C), who then relays the data toward (D). Note that both Figs. 1 and 2 represent the same network. The only difference is that in Fig. 1, the wireless broadcast property is not considered, while this is taken into account in Fig. 2.

In Fig. 1, when either (A) or (B) transmits to (C), the other node does not receive and decode the transmitted data. With this point-to-point link model, the only way for (A) and (B) to jointly compress their data is to carry out distributed source coding [Slepian and Wolf 1973; Wyner and Ziv 1976; Pradhan and Ramchandran 1999]. In Fig. 2, we suppose that all nodes transmit using omni-directional antennas under the free-space path loss model. Let (A) transmit its data to (C) first before (B) does. Furthermore, assuming that the distance between (A) and (B) is not more than that between (A) and (C), then when (A) transmits to (C), its data can be received by (B). Now, if (B) receives the data of (A), it can utilize these data in carrying out data compression. More specifically, node (B) can compress its data based on the explicit knowledge of the data of (A), and therefore, avoid the complexity associated with implementing distributed source coding.

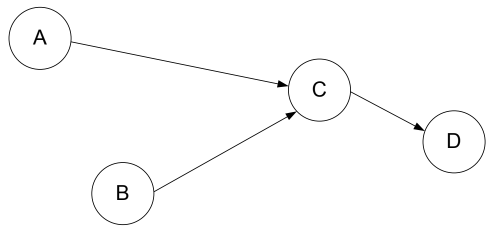

Fig. 1. System of four wireless sensor nodes. Each wireless channel is abstracted as a single point-to-point link. Transmission from (A) to (C) does not reach (B). (A) and (B) can only carry out joint data compression by following the complex distributed source coding approach.

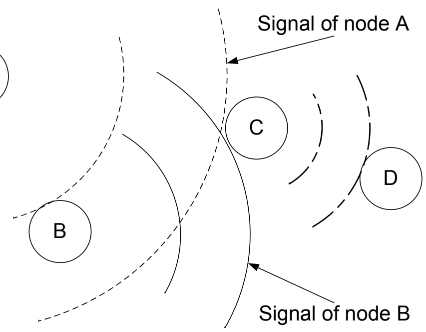

Fig. 2. All nodes transmit using omni-directional antennas. Assuming that the distance between (A) and (C) is not less than the distance between (A) and (B), then (B) is able to capture data sent from (A) to (C) and then uses that data to compress its own data.

The above observations motivate us to exploit the wireless broadcast property for nodes to carry out joint data compression in a spatially-correlated cluster-based wireless sensor network. In our network model, there are two types of nodes, i.e., sensing nodes and cluster head/relaying nodes. Each cluster consists of multiple sensing nodes (also called sensors) and one cluster head. Data collected by each sensor are forwarded to the corresponding cluster head using direct transmission and time division multiple access (TDMA). The cluster head in turn routes data collected in its clusters toward a command center which can be accessed by the end users. This network is depicted in Fig. 3. We also assume that sensing nodes are much more energy constrained compared to cluster head/relaying nodes. The objective is therefore to conserve energy and prolong lifetimes of sensing nodes in each cluster. This is achieved by specifying the manner in which sensors in each cluster collaborate in transmitting (broadcasting) their data and, at the same time, receiving and utilizing the data transmitted by others for data compression.

The main contributions of our paper are:

— We propose the collaborative broadcasting and compression (CBC) approach which allows sensors to cooperate in transmitting, receiving, and compressing data in order to conserve energy (Section 4).

— We formulate an optimization problem of which the objective is to find a CBC scheme that jointly optimizes the lifetimes of all sensors in each cluster, with respect to some optimality criteria (Section 5). We show that this lifetime optimization problem can be solved by a sequence of linear programming problems (Section 6).

— When the number of sensors in each cluster is large, we propose a heuristic CBC scheme that achieves near optimal performance at a lower complexity (Section 7).

— We discuss how our CBC schemes perform under node startup cost and transmission errors and also argue that the schemes are nearly independent to the operation of the relaying network (Section 8).

— Finally, we obtain numerical results which show that by applying the CBC approach, significant increase in sensor lifetime can be achieved (Section 9).

We note that some of the above results have been presented in [Hoang and Motani 2005a; 2005b].

2. RELATED WORKS

Works that are most closely related to this paper are by Chou et al. [2003], Agnihotri et al. [2005], and by Scaglione and Servetto [2002]. In [Chou et al. 2003], the authors propose an approach that combines adaptive signal processing and distributed source coding for sensor nodes in cluster-based WSNs to conserve energy. The main idea of [Chou et al. 2003] is to let sensors in each cluster blindly compress their data with respect to one another, but without the need of explicit inter-sensor communication. The processing burden in this case is shifted to the cluster heads, who need to perform decoding with side information and adaptive filtering to estimate relevant correlation. In [Agnihotri et al. 2005], the authors follow a similar approach of implementing distributed source coding for sensors to conserve transmission energy. Their objective is also similar to our paper, i.e., to maximize the lifetime of the sensor who dies first. For more details on distributed source coding, please refer to [Slepian and Wolf 1973; Wyner and Ziv 1976; Pradhan and Ramchandran 1999]. In [Scaglione and Servetto 2002], the authors propose an approach which is opposite to that of [Chou et al. 2003] and [Agnihotri et al. 2005]. In particular, they promote the idea of source coding based on explicit data of other nodes in the network, which are made available through routing. They argue that, as a routing scheme is already implemented in a WSN, nodes in each routing path actually have explicit information of some other nodes, and therefore, they can carry out classical source coding and avoid the complexity of distributed source coding.

Our work combines the advantages of both [Chou et al. 2003], [Agnihotri et al. 2005], and [Scaglione and Servetto 2002] while avoiding their disadvantages. On one hand, like [Chou et al. 2003] and [Agnihotri et al. 2005], we allow nodes in each cluster to carry out data compression with respect to one another, and without any extra inter-sensor transmissions. The core idea here is the realization that as one node transmits its data to the cluster head, due to the broadcast nature of the media, its transmission reaches multiple other nodes. In addition, as nodes carry out compression based on the explicit information that they receive when other nodes broadcast, classical source coding can be employed as in [Scaglione and Servetto 2002].

In a very recent work, [Vuran and Akyildiz 2006], an event-based MAC protocol is proposed to exploit spatial correlation to reduce the amount of data transmitted in WSN. The main idea is that, as the measurements of nearby sensors are highly correlated, only a subset of these nodes need to report their measurements in order to meet a certain distortion constraint. This approach is applicable for event-based sensing applications in which all sensors observe the same event (represented by a single random variable). Our model used in this paper is more general than this event-based model. In particular, we cover the case when the measurement at each location is equally important. We also note that the work in [Vuran and Akyildiz 2006] is based on the point-to-point communication model. As a result, CBC can be added to the proposed event-based MAC to further improve the performance of the proposed protocol. In particular, when a subset of nodes has been selected to report their measurements, they can employ CBC to further reduce the amount of data transmitted.

3. MODEL OF A CLUSTER-BASED WIRELESS SENSOR NETWORK

3.1 Network Architecture

We consider a small-to-medium-sized cluster-based wireless sensor network as shown in Fig. 3. Sensor nodes are organized into clusters and each cluster is responsible for monitoring a geographical area. We adopt a heterogeneous model in which there are two types of nodes. Type I nodes are sensors whose responsibility is to sense the surrounding environment and then transmitting collected data directly to cluster heads who are type II nodes. Type II nodes gather/aggregate the data collected in their corresponding clusters and relay them toward a command center. We assume that type II nodes are less energy-constrained than type I nodes. We note that the algorithms presented in this paper will work with any clustering algorithms in which nodes are clustered based on having correlated data. For the numerical analysis in Section 9, we cluster nodes based on their location. In particular, each sensor will be associated with the closest cluster head. Note also that in Fig. 3, broadcast communication always takes place and the transmission of one node can be received by every node in the coverage area. The arrows are used to indicate intended destinations only. As will be explained in Section 3.4, our assumption of direct transmission toward cluster heads is suitable for WSNs with small to medium cluster size.

3.2 Sensing and Communication

We consider a periodic sensing scenario in which time is divided into intervals of equal duration called *data-gathering rounds*. In each data-gathering round, each sensor collects useful information about the surrounding environment and outputs a data packet. The data are then forwarded toward the command center using the following mechanism.

— Within each cluster, sensors send data directly to the cluster head using time division multiple access (TDMA). In particular, the duration of each round is divided into slots and each sensor is assigned one slot to transmit data. We assume

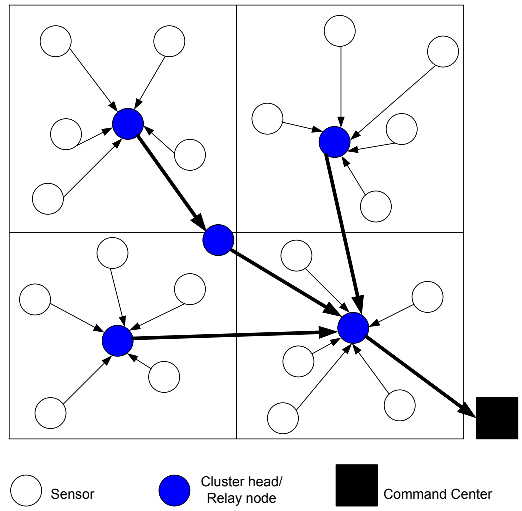

Fig. 3. Model of a cluster-based wireless sensor network. There are two types of nodes, i.e. sensing nodes (type I) and data-gathering/relaying nodes (type II). Sensing nodes transmit collected data directly to the corresponding cluster heads, who then route the data toward a command center.

that inter-cluster interference is negligible. One way to achieve this is by assigning non-overlapping frequency bands to adjacent clusters.

— Upon receiving data collected in their clusters, cluster heads carry out the necessary data fusion/aggregation tasks. After that, the processed data is routed toward the command center over the relay network formed by all type II nodes.

We note that TDMA has been chosen in a number of WSN implementations [Heinzelman et al. 2000; Shih et al. 2001; Arisha et al. 2003] due to its simplicity, low overhead, short communication duty cycle, and no packet collisions. All these factors help conserve sensor nodes' energy. However, it should be noted that TDMA is only effective for scenarios in which the number of transmitting nodes is relatively stable over time. This is, in fact, true in our model of data-gathering WSN. For other sensing applications in which the number of active nodes change frequently, such as those event-based WSN, a contention-based approach would be more scalable than TDMA.

3.3 Energy Model for Wireless Sensor Nodes

First of all, we assume that the sensing operation of each sensor consumes a fixed amount of energy during each data-gathering round. In order to achieve energy-efficiency for sensors, we only focus on controlling their communication-related activities. For the communication-related energy consumption, we adopt the first-order energy model used in [Heinzelman et al. 2000; 2002]. In particular:

—The energy consumed to receive r bits is

$$E_{rx}(r) = E_e \times r \tag{1}$$

where E_e (in Joules/bit) is the energy consumed in the electronic circuits of the transceiver when receiving or transmitting one bit of information. Typical values

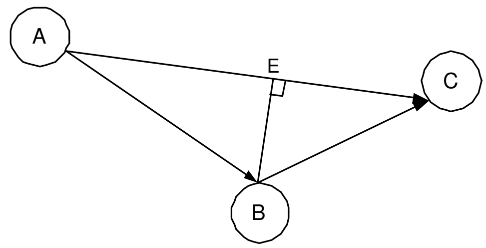

Fig. 4. A simple network with two sensors (A) and (B) communicating to cluster head (C).

for E_e range from 10nJ/bit to 100nJ/bit.

—The energy consumed to transmit r bits over a distance of d meters is

$$E_{tx}(r, d) = E_e \times r + E_a \times d^{\alpha} \times r \quad (2)$$

where α is the channel loss exponent which is typically in the range $2 \leq \alpha \leq 4$. For short communication distances, a free-space path loss model can be assumed and $\alpha = 2$. As the distance increases, a multipath model is more appropriate and $\alpha = 3$ or 4 [Rappaport 1996]. E_a (in Joules/bit/m$^{\alpha}$) is the energy consumed in the power amplifier to transmit one bit of information over a distance of one meter. E_a depends on the receiver sensitivity and its range is from 10pJ/bit/m^2 to 100pJ/bit/m^2 for the free-space path loss model.

—The energy consumed to compress r bits is

$$E_{cp}(r) = E_c \times r. \quad (3)$$

where E_c (in Joules/bit) is the energy used by the processor to compress one bit of information in a data packet based on given side information. In general, E_c is much smaller than the electronic energy E_e. We note that a more complicated model for the compression energy could take into account various factors such as compression ratio and the amount of side information.

3.4 Direct Transmission versus Multihopping

At this point, let us justify our assumption of direct data transmission from sensors toward corresponding cluster heads. Note that the same assumption has been made in some related WSN works, i.e., [Heinzelman et al. 2000; Melo and Liu 2002; Chou et al. 2003; Agnihotri et al. 2005]. In small-to-medium-sized WSNs (which is our assumption), due to short distance between nodes, the energy consumed for receiving is comparable to what is consumed for transmitting a given amount of data. In such scenarios, it has been pointed out in [Heinzelman et al. 2000] that direct transmission is in fact more energy-efficient than multihop routing. Let us demonstrate this fact based on a simple network in Fig. 4.

In Fig. 4, node (A) needs to communicate r bits to cluster head (C). If (A) transmits the data directly to (C), from Section 3.3, the total energy consumption would be:

$$E_{direct} = E_e r + E_a d_{AC}^2 r, \quad (4)$$

where d_{AC} is the distance (in meters) between (A) and (C).

Now, consider using node (B), which lies somewhere in between (A) and (C), to relay data from (A) to (C). In that case, the total energy consumed to transmit r bits from (A) to (B), and then from (B) to (C) would be:

$$\begin{aligned} E_{two\_hop} &= (E_e r + E_a d_{AB}^2 r) + (E_e r) + (E_e r + E_a d_{BC}^2 r) \\ &= 3E_e r + E_a(d_{AB}^2 + d_{BC}^2) r. \end{aligned} \tag{5}$$

Note that when (B) lies in between (A) and (C) as in Fig. 4, we have:

$$d_{AB}^2 + d_{BC}^2 \geq d_{AE}^2 + d_{EC}^2 = 0.5(d_{AC}^2 + (d_{AE} - d_{EC})^2) \geq 0.5 d_{AC}^2. \tag{6}$$

From (4), (5), (6), it can be seen that, for the network in Fig. 4, direct transmission will be more energy-efficient than two-hop routing when:

$$d_{AC} < 2\sqrt{E_e/E_a}. \tag{7}$$

As an example, if we select some typical values as $E_e = 50nJ/bit, E_a = 100pJ/bit/\mathrm{m}^2$, then when $d_{AC} < 45m$, it is more energy-efficient to employ direct communication than two-hop routing. In other words, the assumption of direct transmission from sensors to cluster heads is reasonable in our model of small-to-medium-sized WSNs.

3.5 Spatial Correlation and Data Compression

The objective of this paper is to exploit the spatial correlation among sensor readings for nodes to carry out data compression. In that light, it is appropriate to discuss how spatial correlation and data compression are related.

First we discuss a statistical/information-theoretic approach for specifying spatial correlation and data compression. In this approach, the readings at each sensor are regarded as samples of a random variable and the correlation among readings at different sensors are characterized in an exact way, i.e., by specifying their joint probability distribution [Deshpande et al. 2004], or by establishing the relationship among the random variables [Jindal and Psounis 2004], or by determining their joint entropy [Pattem et al. 2004]. Given a spatial correlation model, the conditional entropy of the quantized data of one sensor, given knowledge of some other sensors's data, can be computed. In general, it is expected that the conditional entropy will decrease when nodes get closer. Using entropy coding, sensors can then compress and transmit at a rate equal to the corresponding conditional entropy.

Now let us consider a more practical approach which is useful when all sensors measure continuous values in the same range and then employ the same quantization scheme. For sensors that are close to one another, the difference in their quantized measures can be small. In that case, simple differential encoding can be employed, i.e., when a node knows the quantized measure of another node, it will only transmit the difference with respect to that measure. This is suboptimal to the approach of characterizing the joint entropy and employing entropy encoding discussed above. However, it has the advantage of not requiring nodes to know the exact spatial correlation structure.

Using either entropy coding or differential encoding as described above, a sensor can compress its data based on the data of another node and therefore, eliminate or reduce the redundancy due to spatial correlation. This will allow the compressing node to transmit less data in a data-gathering round.

4. COLLABORATIVE BROADCASTING AND COMPRESSION: A SIMPLE CASE

4.1 A Simple Cluster-based Sensor Network

Let us introduce our approach by considering a simple cluster-based WSN depicted in Fig. 2. This network consists of only one cluster, which is composed of two sensors (A) and (B) and the cluster head (C), which gathers data collected by (A) and (B) and routes them toward the command center (D). We assume that all nodes transmit using omni-directional antennas and a free-space path loss scenario ($\alpha = 2$). By studying this simple network, we will illustrate the main concepts of our approach. A more general network will be considered in Sections 5, 6, and 7.

If the distance between (A) and (B) is not more than that between (A) and (C), then when (A) transmits to (C), its transmission can also be received by (B). Node (B) therefore has the option of first receiving the data of (A) and then using these data to compress its own data. If (B) does so, for the sake of brevity, we simply say (B) *compresses based on* (A). In addition, we refer to the approach in which sensor nodes coordinate their transmission and reception activities in carrying out joint data compression as *collaborative broadcasting and compression (CBC).*

4.2 Incentives for Collaboration

Let d_{AC}, d_{BC}, and d_{AB} denote the distances (in meters) between (A) - (C), (B) - (C), and (A) - (B) respectively. For this section, we assume that $d_{AB} \leq d_{AC}$. Let r_A and r_B be the amounts of uncompressed data (in bits) that (A) and (B) need to send to (C) during each data-gathering round. Furthermore, let $r_{B|A}$ be the amount of data that B needs to transmit to (C) if it compresses based on (A).

Using (1), (2), and (3), the energy (B) consumes to transmit r_B bits to (C) without compressing based on (A) is

$$E_B = E_e \times r_B + E_a \times d_{BC}^2 \times r_B. \tag{8}$$

On the other hand, the total energy that (B) will spend if it receives from (A), compresses based on (A), and finally transmits $r_{B|A}$ bits to (C) is

$$E_{B|A} = E_e \times r_A + E_c \times r_B + E_e \times r_{B|A} + E_a \times d_{BC}^2 \times r_{B|A}. \tag{9}$$

To make it easier to identify the incentives for (B) to compress based on (A), assume that $r_A = r_B = R$ while $r_{B|A} = r,\ r \leq R$. Then from (8) and (9), node (B) will save energy by compressing based on (A) when

$$\frac{r}{R} \ < \ \frac{E_a \times d_{BC}^2 - E_c}{E_a \times d_{BC}^2 + E_e}. \tag{10}$$

We call $\frac{r}{R}$ the *compression ratio* as it is the ratio of the compressed and uncompressed amounts of data that (B) sends to (C). Node (B) can choose its compression ratio based on a variety of factors, including requirements on acceptable distortion at the receiver. Based on (8), (9) and (10), we note that there is more incentive for (B) to compress based on (A) when

—d_{BC} increases, i.e., node (B) moves farther from the cluster head. In fact, it is evident from (10) that there is a value of d_{BC} below which compression is ineffective, i.e., node (B) will spend more energy to compress and transmit than not to compress at all.

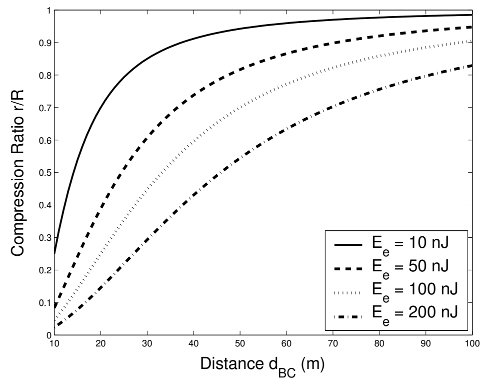

Fig. 5. The incentives for node (B) to compress based on (A) (for the network in Fig. 2). $E_a = 100\text{pJ/bit/m}^2$, $E_c = 5\text{nJ/bit}$ and $E_e = 10,\ 50,\ 100,\ 200\text{nJ/bit}$. The area below each curve corresponds to the region in which (B) can save energy by compressing based on (A).

—$\frac{r}{R}$ is small, i.e., a significant reduction in the size of the data of (B) can be achieved by compression.
—node (B) consumes less energy due to the transceiver electronics and the processor, i.e., when E_e and E_c decrease.

We illustrate the above observations by using the following numerical values: $E_a = 100$ pJ/bit/m^2, $E_c = 5$ nJ/bit, and $E_e = 10,\ 50,\ 100,\ 200$ nJ/bit. Fig. 5 shows the boundary of the region when it is beneficial for (B) to compress based on (A). Specifically, the area below each curve corresponds to the values of compression ratio $\frac{r}{R}$ and transmission distance d_{BC} at which (B) should compress based on (A).

4.3 Maximizing the Lifetime of the Node Who Dies First

In this section, we consider the problem of finding the control scheme that maximizes the time until one of the sensors in a cluster dies. For the network in Fig. 2, we have two possible CBC policies:

—*Policy* μ_1: Let (A) transmit to (C) first, (B) chooses either to transmit uncompressed data to (C) or, if it is beneficial, to compress based on (A) and then transmits to (C).
—*Policy* μ_2: Let (B) transmit to (C) first, (A) chooses either to transmit uncompressed data to (C) or, if it is beneficial, to compress based on (B) and then transmits to (C).

For policy μ_1, the energy consumed by (A) will be

$$E_A^{\mu_1} = E_A = E_e \times r_A + E_a \times d_{AC}^2 \times r_A, \tag{11}$$

while the energy consumed by (B) will be

$$E_B^{\mu_1} = \min\left\{E_B, \frac{E_{B|A}}{\mathbb{1}(d_{AB} \le d_{AC})}\right\}. \tag{12}$$

In (12), $\mathbb{1}(.)$ denotes the indicator function, which returns 1 if the expression inside the brackets is true and returns 0 otherwise. Note that (B) can compress based on

(A) only when $d_{AB} \leq d_{AC}$, if $d_{AB} > d_{AC}$ then $\frac{E_{B|A}}{\mathbb{1}(d_{AB} \leq d_{AC})} = +\infty$ and (12) gives $E_B^{\mu_1} = E_B$.

Similarly, when policy μ_2 is applied, we can write the energy consumption of (A) and (B) as:

$$E_A^{\mu_2} = \min \left\{ E_A, \frac{E_{A|B}}{\mathbb{1}(d_{AB} \leq d_{BC})} \right\} \tag{13}$$

$$E_B^{\mu_2} = E_B. \tag{14}$$

Note that in (13)

$$E_{A|B} = E_e \times r_B + E_c \times r_A + E_e \times r_{A|B} + E_a \times d_{AC}^2 \times r_{A|B} \tag{15}$$

where $r_{A|B}$ is the amount of data that (A) needs to transmit if it compresses based on (B).

Let e_A and e_B be the initial energies of (A) and (B) respectively. The problem of maximizing the time until at least one of the nodes (A) and (B) dies can be formulated as:

$$\arg\max_{t_1, t_2} \{t_1 + t_2\} \tag{16}$$

subject to:

$$t_1 \geq 0, \quad t_2 \geq 0, \tag{17}$$

$$E_A^{\mu_1} \times t_1 \; + \; E_A^{\mu_2} \times t_2 \; \leq \; e_A, \tag{18}$$

$$E_B^{\mu_1} \times t_1 \; + \; E_B^{\mu_2} \times t_2 \; \leq \; e_B. \tag{19}$$

Here t_1 and t_2 are the numbers of data-gathering rounds that policies μ_1 and μ_2 are employed respectively. In addition, t_1, t_2 must take integer values and the above optimization problem is an integer linear program. However, for applications in which sensors' lifetimes are much longer than each data-gathering round, t_1, t_2 can be treated as real variables. Then the above optimization is a linear programming problem and can be solved efficiently with standard methods [Hiller and Lieberman 1995].

5. COLLABORATIVE BROADCASTING AND COMPRESSION: A GENERAL NETWORK

We now apply the CBC approach to control a general cluster-based sensor network as depicted in Fig. 3. Note that our control will still be carried out within each cluster.

5.1 General Notation

We consider a cluster composed of K sensors and a cluster head. The sensor nodes are numbered from 1 to K and the cluster head is denoted by H. Let us introduce the following notation:

—$\mathbf{N} = \{1, \ldots K\}$ is the set of all sensors in the cluster.

—$d_{ik},\ i,k \in \mathbf{N}$, is the distance (in meters) between sensor i and sensor k. d_{kH} is the distance between sensor k and the cluster head.

—$\mathbf{N}_k = \{i \in \mathbf{N},\ i \neq k \mid d_{ik} \leq d_{iH}\},\ k \in \mathbf{N}$, is the set of all nodes whose transmission to the cluster head can be received by k.

—$e_k,\ k \in \mathbf{N}$, is the initial energy of sensor k.

—$E_k(\mathbf{u}),\ k \in \mathbf{N},\ \mathbf{u} \subset \mathbf{N},\ k \notin \mathbf{u}$, is the total energy consumed by node k in each round when it compresses based on all nodes in $\mathbf{u}$. In this way, $E_k(\emptyset)$ denotes the energy consumed by k when it does not compress based on any other node. Note that $E_k(\mathbf{u})$ can be determined using (1), (2), (3).

5.2 Control During Each Data-gathering Round

During each data-gathering round, in order to specify how nodes collaborate their data transmission and compression, two control decisions must be made. Firstly, a transmission order needs to be specified, i.e., each sensor should be assigned a time slot for data transmission. Secondly, given the transmission order, each node needs to know which other nodes it should compress based on.

When there are more than two sensors in the cluster, each of them may be able to compress based on more than one node. We allow this with the following restriction:

Constraint 5.1. During each data-gathering round, each sensor is allowed to compress based on only the data of those sensors who themselves do not compress.

With the above constraint, we give the following definition for a CBC policy that controls the sensors during each data-gathering round.

Definition 5.2. Let $\mathbf{v} \subseteq \mathbf{N}$ be a subset of the set of K sensors, and let $\mathcal{V}$ be the set of all subsets of $\mathbf{v}$, a *CBC policy* is a function $\mu : \mathbf{v} \to \mathcal{V}$ such that for $k \in \mathbf{v}$, sensor k will compress based on the data of all sensors in $\mu(k)$. Furthermore, we require that $\mu(i) = \emptyset$ if $i \in \mu(k),\ \forall i,k$.

Note that a particular CBC policy μ only controls the operation of those sensors belonging to $\mathbf{v}$, a subset of $\mathbf{N}$. This makes Definition 5.2 applicable even if not all K sensors in the cluster are active. We also note that, given a CBC policy μ, a transmission order can always be determined so that each node $k \in \mathbf{v}$ can carry out compression and transmission as specified by μ. This can be achieved by simply scheduling all nodes who do not compress their data to transmit first. Then, all other nodes can be scheduled later so that they can receive data from uncompressing nodes and use these data to compress their own data.

5.3 Control over Multiple Data-gathering Rounds

By definition, a particular CBC policy μ specifies how the sensors in the set $\mathbf{v} \subseteq \mathbf{N}$ operate during a particular data-gathering round. To control the sensors over multiple data-gathering rounds, we define a CBC scheme as:

Definition 5.3. Let $\mathbf{v} \subseteq \mathbf{N}$ be a subset of the set of all K sensors, a *CBC scheme* is a policy-time set

$$\mathbf{\Psi} = \Big\{(\mu_1, t_1), \ldots (\mu_m, t_m)\Big\}$$

where m is some positive integer and the pair (μ_i, t_i), $1 \le i \le m$, indicates that CBC policy μ_i is employed on $\mathbf{v}$ for t_i data-gathering rounds. Furthermore, let e_k^{res} be the residual energy that node k has prior to the application of $\mathbf{\Psi}$, then $\mathbf{\Psi}$ is said to be *feasible* if and only if:

$$\sum_{i=1}^{m} E_k(\mu_i(k)) \times t_i \le e_k^{res},\ \forall k \in \mathbf{v}. \tag{20}$$

Condition (20) guarantees that when $\mathbf{\Psi}$ is applied, no sensor in $\mathbf{v}$ consumes more than its residual energy.

5.4 Sensor Lifetime and System Performance

Let us suppose that some feasible CBC schemes are employed to control K sensors until all of them use up their energy and die. The operation of the cluster can be divided into K consecutive phases, with phase k starting when $k-1$ out of K sensors die and ending when k out of K sensors die. We then define a lifetime vector of the cluster as follows.

Definition 5.4. The K- element vector L, with $L(k)$ being the time when phase k ends, is called a *lifetime vector* of the cluster. Furthermore, a lifetime vector L is said to be *achievable* if it is the result of the application of some K feasible CBC schemes, each controlling one phase of the cluster operation.

It is straight forward to prove the following lemma, which states that by applying the CBC approach, every node in the cluster will achieve at least the lifetime corresponding to the case when no node carry out joint data compression,

LEMMA 5.5. *Let $\widetilde{L}$ be the lifetime vector achieved when no node carry out joint data compression, then for every achievable lifetime vector L, $L(k) \ge \widetilde{L}(k)$, $\forall k \in \mathbf{N}$.*

Now, let us examine some options for characterizing the cluster data-gathering performance based on the lifetime vector L. For the most stringent performance, the cluster ceases functioning when one of its K sensors dies, i.e., at time $L(1)$. For the least stringent case, we may assume that the cluster keeps on functioning until all of its sensors die, i.e., at time $L(K)$. However, in reality, when sensor nodes die one by one, what will be observed is a gradual decrease in the quality of the data-gathering job. The decrease here is in terms of information-fidelity and/or geographical coverage. This gradual decrease in performance can not be captured by any single element of the lifetime vector L. Therefore, we propose to maximize elements of L in sequence, with the maximization of the k^{th} element being carried out conditioned on the maximization of the 1^{st}, $2^{nd}, \ldots (k-1)^{th}$ elements. In a more concrete form, we adopt the following definition for the optimality of the cluster lifetime vector:

Definition 5.6. An achievable lifetime vector L^* is said to be *optimal* if for every other achievable lifetime vector L, $L \neq L^*$, there exists $k \in \mathbf{N}$ such that

$$L^*(i) \ge L(i), \quad \forall i \in \{1, \ldots k\}, \tag{21}$$

with at least one strict inequality.

Note that our optimality criteria gives priority to improving the lifetimes of nodes who die early. This will keep as many nodes to stay alive as possible, and therefore, assure a high-level data-gathering performance for a long period of time. This also leads to reduction in the variance among nodes' lifetimes, i.e., nodes die closer together.

6. LIFETIME VECTOR OPTIMIZATION PROBLEM

Based on Definition 5.6, we introduce the following lifetime vector optimization (LVO) problem:

Lifetime Vector Optimization (LVO) Problem: *Given a cluster of K sensors, find K feasible CBC schemes that respectively control K phases of the cluster operation so that the resultant lifetime vector is optimal.*

6.1 A General Approach to Solve the LVO Problem

The LVO problem can be solved by the following K-step procedure.

— *Step* 1: Given all K sensors with their initial energies, we find a feasible CBC scheme that controls phase 1 of the cluster operation so that the time when one of the K sensors dies is maximized. Step 1 gives us L_1^* which is the maximum lifetime of the node who dies first.

— *Step* k, $2 \leq k \leq K$: The first $k-1$ steps give us $L_1^*, \ldots L_{k-1}^*$. Now the task is to find k feasible CBC schemes that control the first k phases of the cluster operation so that the time when i out of K sensors die is L_i^*, $\forall i < k$, and the time when k out of K sensors die is maximized. This conditional maximum time when k out of K sensors die is denoted by L_k^*.

THEOREM 6.1. *The K feasible CBC schemes obtained in Step K solve the LVO problem.*

PROOF. After Step K, we obtain K feasible CBC schemes that achieve the lifetime vector $(L_1^*, L_2^*, \ldots L_K^*)$. We will prove that this lifetime vector is optimal with respect to Definition 5.6.

Let L be any achievable lifetime vector and $L \neq (L_1^*, L_2^*, \ldots L_K^*)$. There must be k, $1 \leq k \leq K$, such that $L(i) = L_i^*$, $\forall i < k$ and $L(k) \neq L_k^*$. Note that L_k^* is the maximum time when k out of K sensors die, subject to the constraint that the time when i out of K sensors die is $L(i)$, $\forall i < k$. Therefore, we must have $L(k) < L_k^*$. In other words, k satisfies the optimality condition in Definition 5.6 and $(L_1^*, L_2^*, \ldots L_K^*)$ is the optimal lifetime vector. □

6.2 Linear Programming Formulation

Now let us show how each step in the K-step procedure described in Section 6.1 can be formulated as a linear programming (LP) problem. We do so for Step 1 and 2. For Steps k, $k > 2$, the formulation is similar.

6.2.1 *Formulating Step* 1 *as an LP.* As the number of CBC policies in phase 1 can be very large, what we will do first is to narrow down the policies that should be time-shared. Given a CBC policy μ, let us denote by $\mathbf{u}$ the set of all nodes that transmit without compressing based on another node. We must have:

$$\forall k \in \mathbf{N} \backslash \mathbf{u}, \ \mathbf{u} \cap \mathbf{N}_k \neq \emptyset. \tag{22}$$

In other words, each node in $\mathbf{N}\backslash\mathbf{u}$ must be able to receive the transmission of at least one node in $\mathbf{u}$. Furthermore, we only need to consider those policies μ that satisfy:

$$\mu(k) = \gamma_1^k(\mathbf{u}) = \begin{cases} \emptyset, & \forall k \in \mathbf{u}, \\ \underset{\mathbf{w} \subset (\mathbf{u} \cap \mathbf{N}_k)}{\arg\min} \ \{E_k(\mathbf{w})\}, & \forall k \in \mathbf{N}\backslash\mathbf{u}. \end{cases} \tag{23}$$

This is because given a set $\mathbf{u}$ of nodes that transmit without compressing, all other nodes should choose to compress based those nodes that result in the most energy saving. As a result, each policy μ that we will time-share is completely specified if the set of nodes that transmit without compressing is given.

Let $\mathcal{U}_1$ be the set of all subsets of $\mathbf{N}$ that satisfies condition (22), i.e.,

$$\mathcal{U}_1 = \{\mathbf{u} \subseteq \mathbf{N} \mid \forall k \in \mathbf{N}\backslash\mathbf{u}, \ \mathbf{u} \cap \mathbf{N}_k \neq \emptyset\}. \tag{24}$$

Also, let $t_1^{\mathbf{u}}, \mathbf{u} \in \mathcal{U}_1$, be the number of data-gathering rounds that all nodes belonging to $\mathbf{u}$ transmit without compressing while all nodes not belonging to $\mathbf{u}$ carry out data compression. Note that the subscript '$_1$' of γ_1^k, $\mathcal{U}_1$, and $t_1^{\mathbf{u}}$ is used to indicate that these are function or parameters of phase 1. As we have mentioned, when the lifetimes of sensors are much longer than each data-gathering rounds, $t_1^{\mathbf{u}}$ can be treated as real variables. Then, the problem of maximizing the lifetime of the node who dies first can be written as the following linear program:

$$\underset{t_1^{\mathbf{u}}, \ \forall \mathbf{u} \in \mathcal{U}_1}{\arg\max} \sum_{\forall \mathbf{u} \in \mathcal{U}_1} t_1^{\mathbf{u}} \tag{25}$$

subject to:

$$t_1^{\mathbf{u}} \geq 0, \quad \forall \mathbf{u} \in \mathcal{U}_1, \tag{26}$$

$$\sum_{\forall \mathbf{u} \in \mathcal{U}_1} \left(t_1^{\mathbf{u}} \times E_k(\gamma_1^k(\mathbf{u})) \right) \ \leq \ e_k, \quad \forall k \in \mathbf{N}. \tag{27}$$

Solving the above LP gives us a CBC scheme that maximizes the time until at least one of the K sensors die. This maximum lifetime is denoted by L_1^*. For this particular CBC scheme, let us denote by $\mathcal{D}_1$ the set of nodes that actually die at time L_1^*. $\mathcal{D}_1$ can be determined just by checking the residual energies of all K sensors after phase 1.

6.2.2 *Formulating Step* 2 *as an LP.* Let us first consider the case when the set $\mathcal{D}_1$, obtained by solving the LP for Step 1, only has one element, denoted by k^*. In other words, only sensor k^* dies at time L_1^*. In phase 2 we are left with $K-1$ nodes in the set $\mathbf{N}\backslash\{k^*\}$. Now, similar to γ_1^k, $\mathcal{U}_1$, $t_1^{\mathbf{u}}$ ($\forall \mathbf{u} \in \mathcal{U}_1$) of phase 1, we can define γ_2^k, $\mathcal{U}_1$, $t_2^{\mathbf{v}}$ ($\forall \mathbf{v} \in \mathcal{U}_2$) for phase 2. Then the task of Step 2, i.e., to find two CBC schemes that control phases 1 and 2 so that the duration of phase 1 is L_1^* and the duration of phase 2 is maximized, can be written as the following LP.

$$\arg\max_{t_1^{\mathbf{u}},\, t_2^{\mathbf{v}},\, \forall \mathbf{u} \in \mathcal{U}_1,\, \forall \mathbf{v} \in \mathcal{U}_2} \sum_{\forall \mathbf{v} \in \mathcal{U}_2} t_2^{\mathbf{v}} \tag{28}$$

subject to:

$$t_1^{\mathbf{u}} \geq 0 \ \forall \mathbf{u} \in \mathcal{U}_1, \quad t_2^{\mathbf{v}} \geq 0 \ \forall \mathbf{v} \in \mathcal{U}_2, \tag{29}$$

$$\sum_{\forall \mathbf{u} \in \mathcal{U}_1} t_1^{\mathbf{u}} \ = \ L_1^*, \tag{30}$$

$$\sum_{\forall \mathbf{u} \in \mathcal{U}_1} \left(t_1^{\mathbf{u}} \times E_{k^*}(\gamma_1^{k^*}(\mathbf{u})) \right) \ \leq \ e_{k^*}, \tag{31}$$

$$\sum_{\forall \mathbf{u} \in \mathcal{U}_1} \left(t_1^{\mathbf{u}} \times E_k(\gamma_1^k(\mathbf{u})) \right) + \sum_{\forall \mathbf{v} \in \mathcal{U}_2} \left(t_2^{\mathbf{v}} \times E_k(\gamma_2^k(\mathbf{v})) \right) \ \leq \ e_k, \quad \forall k \in \mathbf{N} \backslash \{k^*\}. \tag{32}$$

In the cases when $\mathcal{D}_1$ contains more than one node, we will need to formulate the above LP for each possible value of k^*, and determine which one leads to the maximum lifetime of phase 2.

6.3 Complexity of Solving Each Step by LP

Note that of all K steps, Step 1 involves solving the smallest LP. The size of the LP for Step 1 depends on the cardinality of the set $\mathcal{U}_1$, which in turn depends on the cluster topology and sensor's energy model. In the worst case, $\mathcal{U}_1$ contains all non-empty subsets of $\mathbf{N}$, and therefore, has the cardinality of $2^K - 1$. This means it is only practical to solve the above LPs when the number of nodes in the cluster is small.

7. HEURISTIC ALGORITHM

In this section, we propose heuristic CBC schemes which can be obtained at a much lower complexity compared to solving the linear programming problems in Section 6.2. In Section 9, we will present numerical results which show that the heuristic schemes achieve a near optimal lifetime vector. We will focus on the heuristic scheme that controls phase 1 of the cluster operation. The schemes for other phases can be constructed in a similar manner.

In the heuristic CBC scheme that controls phase 1, i.e., $\{(\mu_1, t_1), \ldots (\mu_m, t_m)\}$, each policy μ_i is employed for an interval of T data-gathering rounds, i.e., $t_i = T,\ i = 1, \ldots m$, where T is a fixed integer. For each interval, a CBC policy is selected in a greedy way, with the objective of maximizing the minimum residual energy of K nodes after the interval. It is expected that, the smaller the value of T, the better the performance of our heuristic, as it can respond better to the energy state of sensors. However, as will be shown in Section 9, even with a relatively large value of T, the heuristic still performs quite close to optimal.

7.1 A CBC Policy for T Data-gathering Rounds

Let interval $n,\ n = 1, 2, \ldots$, denote the time from the beginning of data-gathering round $(n-1) \times T + 1$ until the end of data-gathering round $n \times T$. Let $e_k^n,\ e_k^n \geq 0,\ k \in \mathbf{N}$, be the residual energy of node k at the beginning of interval n. Also, let μ_n be the CBC policy being employed in interval n. If no node uses up its energy during interval n, the residual energy of node k at the beginning of interval $n+1$ is

$$e_k^{n+1} = e_k^n - T \times E_k(\mu_n(k)). \tag{33}$$

The lifetime of each node is directly related to its residual energy. Therefore, during each interval, it is intuitive to employ a greedy CBC policy that maximizes the minimum value of the residual energy of all K nodes after the interval. In other words, for interval n, we will find the CBC policy μ_n^* that satisfies

$$\mu_n^* = \arg\max_{\mu_n} \left\{ \min_{k \in \mathbf{N}} \{e_k^{n+1}\} \right\}. \tag{34}$$

In order to approximate μ_n^*, we start with a CBC policy μ in which no node compresses based on any other node and improve μ in each iteration. Policy μ is improved by first identifying the node i^* that will have the least residual energy at the end of interval n if policy μ is applied and then let i^* compress based on another node. When there are more than one node that i^* can compress based on, i^* will choose the node j^* that satisfies

$$j^* = \arg\max_{j \in \mathbf{Q}} \left\{ \min\{e_{i^*}^n - T \times E_{i^*}(\{j\}), \left(e_j^n - T \times E_j(\emptyset)\right) / \mathbb{1}(j \notin \mathbf{U})\} \right\}. \tag{35}$$

The reason for i^* to compress based on j^* selected by (35) is that if we let i^* compress based on some node j, then j is not allowed to compress based on any node, and j can become the node who has the least residual energy at the end of interval n. In (35), $\mathbf{Q}$ is the set of nodes that i^* can compress based on while $\mathbf{U}$ consists of nodes that are not able to compress and nodes that have already been used by other nodes for their data compression. After improving $e_{i^*}^{n+1}$ by letting i^* compress based on j^*, we move to the next iteration and repeat the process.

We name the above algorithm **Single_CBC** and give its pseudo-code in Fig. 6. Note that the inputs for algorithm Single_CBC are the residual energies of the K sensors at the beginning of interval n, i.e., $(e_1^n, \ldots e_K^n)$. The output of Single_CBC is a CBC policy that controls K sensors during interval n. As has been mentioned, $\mathbf{U}$ denotes the set of nodes that are either used by other nodes for their data compression and/or not able to compress. Besides, $\mathbf{V}$ denotes the set of nodes who compress based on some nodes in $\mathbf{U}$.

7.2 A Heuristic CBC Scheme for Phase 1

By repeatedly applying the Single_CBC algorithm until one of the sensor nodes uses up its energy and dies, we obtain a set of CBC policies, each control the collaboration of K sensors for T data-gathering rounds. We name the algorithm that does so the **Multiple_CBC**. The inputs for Multiple_CBC are the initial energy of K nodes, i.e., $(e_1, \ldots e_K)$. Multiple_CBC outputs a sequence of CBC policies that are employed until one of the sensors dies. The pseudo-code for Multiple_CBC is presented in Fig. 7.

Algorithm: Single_CBC$(e_1^n, \dots e_K^n)$

$\mu(k) \leftarrow \emptyset,\ \forall k \in \mathbf{N}$
$\mathbf{U} \leftarrow \emptyset; \quad \mathbf{V} \leftarrow \emptyset$
loop
$\quad i^* \leftarrow \underset{k\ \in\ \mathbf{N}\backslash(\mathbf{U}\cup\mathbf{V})}{\arg\min} \left\{ e_k^n - T \times E_k(\mu(k)) \right\}$
$\quad \mathbf{Q} \leftarrow \left\{ k \in \mathbf{N}_{i^*} \backslash \mathbf{V} \mid E_{i^*}(k) < E_{i^*}(0) \right\}$
$\quad$ **if** $\mathbf{Q} \neq \emptyset$
$\quad\quad j^* \leftarrow \underset{j \in \mathbf{Q}}{\arg\max} \left\{ \min\{e_{i^*}^n - T \times E_{i^*}(j), (e_j^n - T \times E_j(0))/\mathbb{1}(j \notin \mathbf{U})\} \right\}$
$\quad\quad \mu(i^*) \leftarrow \{j^*\}$
$\quad\quad \mathbf{U} \leftarrow \mathbf{U} \cup \{j^*\}; \quad \mathbf{V} \leftarrow \mathbf{V} \cup \{i^*\}$
$\quad$ **else**
$\quad\quad \mathbf{U} \leftarrow \mathbf{U} \cup \{i^*\}$
$\quad$ **endif**
$\quad$ **if** $\mathbf{U} \cup \mathbf{V} = \mathbf{N}$
$\quad\quad$ **break**
$\quad$ **endif**
endloop
$\mu(k) = \gamma_1^k(\mathbf{U})$
return μ

Fig. 6. Pseudo-code of algorithm **Single_CBC**$(e_1^n, \dots e_K^n)$. Inputs are the residual energies of K sensors at the beginning of interval n, i.e., $(e_1^n, \dots e_K^n)$. Output is CBC policy μ that will be used to control K sensors during interval n.

Algorithm: Multiple_CBC$(e_1, \dots e_K)$

$\mathbf{\Psi} \leftarrow [\ ]$
$e_k^{res} \leftarrow e_k,\ \forall k \in \mathbf{N}$
loop
$\quad \mu \leftarrow$ **Single_CBC**$(e_1^{res}, \dots e_K^{res})$
$\quad \mathbf{\Psi} \leftarrow [\mathbf{\Psi},\ \mu]$
$\quad e_k^{res} \leftarrow e_k^{res} - T \times E_k(\mu(k)),\ \forall k \in \mathbf{N}$
$\quad$ **if** $\min_{k \in \mathbf{N}}\{e_k^{res}\} \ \leq\ 0$
$\quad\quad$ **break**
$\quad$ **endif**
endloop
return $\mathbf{\Psi}$

Fig. 7. Pseudo-code of algorithm **Multiple_CBC**$(e_1, \dots e_K)$. Inputs are the initial energies of K sensors, i.e., $(e_1, \dots e_K)$. Output is a sequence of CBC policies, each policy is employed to control one interval of T rounds.

7.3 Complexity of Heuristic Algorithm

Heuristic CBC schemes for controlling phases $2, 3, \dots K$ can be obtained in a similar manner to that of phase 1. As the complexity for obtaining heuristic CBC schemes is highest for phase 1, let us determine this complexity.

From the pseudo-code of Single_CBC, it can be seen that the main tasks inside the loop are to find i^* and j^*. Both of these involve finding the minimum value from a set of at most K elements and therefore, the complexity is of the order $O(K)$. At the same time, the main loop is repeated for no more than K times. Therefore, we can conclude that the worst-case complexity of Single_CBC is $O(K^2)$.

During each iteration of Multiple_CBC algorithm, Single_CBC algorithm is carried out. The number of iterations being taken in Multiple_CBC depends on the lifetime of the node who dies first. We note that the energy consumed by each node in a data-gathering round is lower-bounded by the energy consumed in the electronic circuits. In particular, if in each data-gathering round, each node is required to communicate a packet of length R bits (without compression) to the cluster head, then no matter whether a node compresses based on other nodes or not, the energy consumed in each round is lower bounded by $E_{lb} = E_e \times R$. Therefore, the lifetime of sensor $k,\ k = 1, \dots K$, is upper-bounded by $L_{ub} = \frac{e_k}{E_{lb}}$. As the upper-bound L_{ub} does not grow with K, the number of iterations of Multiple_CBC algorithm does not grow with K either. As a result, the complexity of Multiple_CBC algorithm is of the same order of that of the Single_CBC algorithm, which is equal $O(K^2)$.

8. REFLECTIONS ON THE CBC APPROACH

8.1 Startup Cost of Sensor Nodes

For wireless sensors, startup cost refers to the energy consumed during the radio startup transient [Shih et al. 2001], [Raghunathan et al. 2002]. Note that no data can be transmitted or received during this transient phase. One way to minimize the negative effect of this (wasted) energy is to operate at a large packet size so that the total energy consumed by the transceiver unit is dominated by transmission and reception energy [Shih et al. 2001].

In our CBC schemes, when a node wants to receive and compress based on the data of another node, its radio needs to be active during at least two time slots in each data-gathering round, i.e., one is for receiving and the other is for transmitting data. If these receiving and transmitting time slots are not adjacent to each other, in order to conserve energy, the node may need to turn off the radio component after the receiving and then turn it on again for transmission. Doing so will not cause any problem as long as the radio startup cost is negligible.

For the case when the startup cost is significant, we can mitigate the problem of non-adjacent receiving and transmitting time slots by constraining that in each CBC policy, at most one node can compress based on any particular node. This will allow a node to transmit right after receiving and compressing its data. Note that the constraint can be easily incorporated into our linear programming and heuristic approaches in Sections 6.2 and 7. In Section 9, we will present numerical result to show that with this extra constraint, our CBC schemes still yield a significant improvement for sensors' lifetimes.

8.2 Packet Transmission Errors

So far, when studying the CBC approach, we have assumed that the packet loss due to transmission errors is negligible. Now let us consider how our CBC schemes perform when packet transmission errors are taken into account.

We suppose that, in a particular CBC scheme, sensor k is assigned to compress based on sensor i during some time interval. This will improve the lifetime of k. However, due to transmission errors, in some data-gathering rounds, k may not be able to receive packets sent by i and therefore, can not compress its data. As a result, our CBC schemes will achieve less lifetime improvement, relative to the case when all transmissions are successful.

Still referring to the above scenario, we assume that k actually receives a packet sent by i and uses that to compress its own packet. However, let us suppose that the packet of i is not received successfully by the cluster head H. If H keeps on requesting i to resend its packet until a successful reception, then the compressed packet of k will eventually be decoded. On the other hand, if no retransmission is allowed, the loss of the packet of i will leads to the loss of the packet of k as this packet can not be decompressed. As a result, under our CBC schemes, those nodes who compress based on others' data can incur a higher packet loss probability.

For node k, the packet loss probability will be worst when the packet loss processes corresponding to the transmission from i to H and the transmission from k to H are independent. In that case, let P_i^e and P_k^e be the packet loss probabilities for the transmissions from i and k (to H) respectively, the packet loss probability for k can be written as:

$$P_{k|i}^e = P_k^e + P_i^e - P_k^e P_i^e \approx P_k^e + P_i^e. \tag{36}$$

As our CBC schemes may increase the packet loss rate for nodes that compresses based on others, apart from the lifetime improvement, it is useful to look at the performance in terms of the total number of packets successfully transmitted by each node throughout its lifetime. In Section 9 we will present result to show that even with a high packet loss rate (10%), our CBC schemes still result in significant increases in the total number of successful packets transmitted by each node.

8.3 Effects on the Relaying Network

Now, let us discuss the effects that our CBC approach can have on the relaying network formed by type II nodes. First of all, as nodes in each cluster jointly compress their data, the amount of data sent to the cluster heads will be reduced. This can allow the cluster heads to spend less energy receiving. Secondly, as nodes encode their data based on explicit side information, the decoding scheme at each cluster head will not be complex. In fact, the cluster heads may not want to decompress the data, since they will eventually perform data fusion/aggregation. Finally, after data fusion/aggregation, there will be no change on the amount of data flowing out of each cluster. Therefore, other parts of the relaying network are not affected by the data compression carried out within each cluster.

Based on the above discussion, we state that our CBC approach is independent to the operation of the relaying network. Therefore, it can be applied in conjunction with energy-efficient routing schemes that have been proposed for WSNs ([Intanagonwiwat et al. 2000]).

9. NUMERICAL STUDY

In this section, we present numerical results which show the performance gain, i.e., the increase in sensors' lifetimes and number of packets successfully transmitted, when the CBC approach is employed. We will compare the performance of three control schemes, i.e., the optimal scheme obtained by solving the LPs formulated

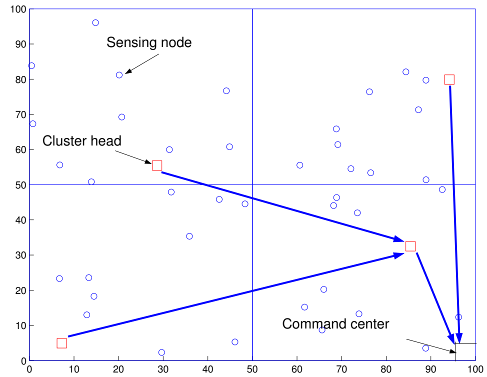

Fig. 8. An example of a network of size $100 \times 100m$. The monitoring area is divided into four clusters. In each cluster, there are $K = 10$ sensing nodes and one cluster head. Sensing nodes and cluster heads are deployed randomly and uniformly within their cluster area.

in Section 6.2, the heuristic scheme proposed in Section 7, and finally the scheme in which all sensors transmit to the cluster head without joint compression.

9.1 Experimental Model

The monitored field is represented by a square area of size D meters. This area is further divided into C^2 disjoint clusters, each is a square of size $\frac{D}{C}$. In each cluster, there are K sensors and one cluster head. We assume that the sensor nodes, together with the cluster head, are deployed randomly within each cluster, with their coordinates uniformly distributed. In Fig. 8, a sample network of size $D = 100$ meters, divided into four clusters and with $K = 10$ sensors per cluster, is shown.

The energy model of each sensor node is as described in Section 3.3 with: $E_a = 100\text{pJ/bit/m}^2$, $E_c = 5\text{nJ/bit}$, and $E_e = 10,\ 50,\ 100\text{nJ/bit}$. Each sensor node has an initial energy storage of 5J. In each round, without compression, each sensor needs to send a packet of length $R = 400$ bits to the cluster head.

Each of the results presented in the following section is obtained by generating 500 instances of the network and averaging the performance of tested schemes. Note that $L(k)$, $k = 1, \ldots K$, denotes the time when k out of the K sensors in a cluster die when some control scheme is employed.

9.2 Results and Discussion

First, let us study the benefit, if any, of allowing a sensor to compress based on the data of *more than one* other sensors. In particular, we compare the case when each sensor compresses based on at most one other sensor to the case when each sensor can compress based on up to two other sensors. Without compression, each sensor needs to transmit $R = 400$ bits during each data gathering round. Let us assume that if a particular sensor compresses based on the data of one other sensor, it will need to transmit r bits per data gathering round. Also, if that sensor compresses based on the data of two other sensors, it will only need to transmit r_2 bits, where $r_2 \leq r$. We simplified the analysis by assuming r and r_2 are fixed for all sensors. In

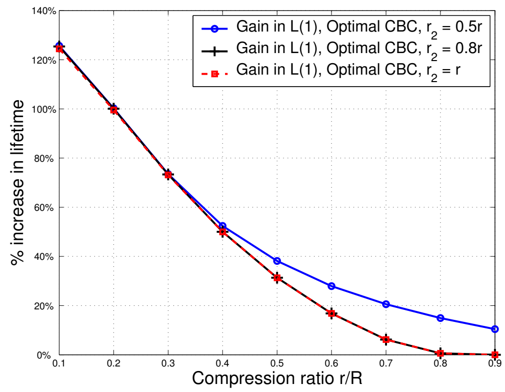

Fig. 9. Percentage increases (relative to no compression) in the lifetime of the sensor who dies first when the optimal CBC is applied. r is the rate when a sensor compresses based on one other sensor. r_2 is the rate when a sensor compresses based on two other sensors. There are $K = 10$ nodes in each cluster and the energy model is: $E_a = 100\text{pJ/bit/m}^2, E_e = 50\text{nJ/bit}$ and $E_c = 5\text{nJ/bit}$.

Fig. 9, we plot the gain in $L(1)$, i.e., the lifetime of the node who dies first, when $r/R = 0.1 \to 0.9$ and $r_2/r = 0.5,\ 0.8$, and 1. As can be seen, when $r_2/r = 0.5$, there is some small gain when letting each sensor to compress based on up to two other sensors' data. However, we note that $r_2/r = 0.5$ is not very realistic. When $r_2/r = 0.8$ or 1, there is absolutely no gain when allowing a sensor to compress based on two other nodes, relative to constraining it to compress based on at most one node. This is because when a sensor compresses based on two nodes, it needs to spend twice as much energy in receiving, compared to when it compresses based on one node only. This extra receiving energy is more than what is gained by compressing based on one extra node. Due to this observation, in the rest of this section, we assume that each sensor will only compress based on the data of at most one node.

In Fig. 10, we show the percentage increases in sensors' lifetimes when the optimal control schemes and the heuristic control schemes are applied, relative to when no node compresses its data. The percentage increases in the lifetimes of nodes who die first, second, fifth, and tenth are plotted versus the compression ratio $\frac{r}{R}$. Here, we assume that there no extra gain when a node compress based on multiple nodes. We also assume that all packets are successfully transmitted. As can be seen, the lifetime improvements strongly depend on the compression ratio $\frac{r}{R}$, i.e., on the spatial correlation among data collected at different sensors. When $\frac{r}{R}$ is low, the performance gain of both optimal and heuristic schemes are very significant. The gain is largest for $L(1)$ while there is negligible gain for $L(10)$. This is exactly what our objective is; we want to improve the lifetimes of those nodes who die earlier than others.

Another important observation from Fig. 10 is that the performance of the heuristic scheme is nearly the same as that of the optimal control scheme. We note that for the heuristic scheme, the value of T, i.e., the interval at which the greedy policy is updated, is set to 1000 while the network lifetime is around 100000.

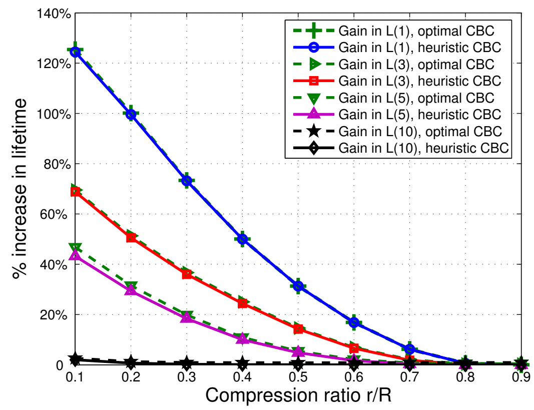

Fig. 10. Percentage increases (relative to no compression) in sensors' lifetimes versus compression ratio when the optimal CBC and heuristic CBC schemes are applied. $L(1)$, $L(3)$, $L(5)$, $L(10)$ are the lifetimes of nodes who die first, third, fifth, and tenth, respectively. There are $K = 10$ nodes in each cluster and the energy model is: $E_a = 100\text{pJ/bit/m}^2, E_e = 50\text{nJ/bit}$ and $E_c = 5\text{nJ/bit}$. Packet loss is assumed to be negligible.

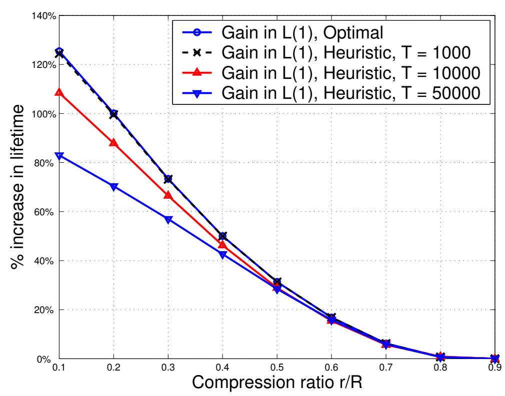

Fig. 11. Percentage increases (relative to no compression) in $L(1)$ versus compression ratio when the optimal CBC and heuristic CBC schemes are applied. There are $K = 10$ nodes in each cluster and the energy model is: $E_a = 100\text{pJ/bit/m}^2, E_e = 50\text{nJ/bit}$ and $E_c = 5\text{nJ/bit}$. Packet loss is assumed to be negligible.

When we increase the updating interval T, the performance gain of heuristic policy decreases. This is shown in Fig. 11. However, as can be seen, even when this interval is set to a very large value, i.e., $T = 50000$, the gain, relative to the case when no compression is carried out, is still very significant. This indicates that we can use the heuristic scheme, which has low complexity without sacrificing performance.

We then look at how the performance of the heuristic scheme depends on the number of sensors per cluster and the cluster size. In Fig. 12 we show the percentage increase for $L(1)$ when the heuristic scheme is applied, as compared to the case when

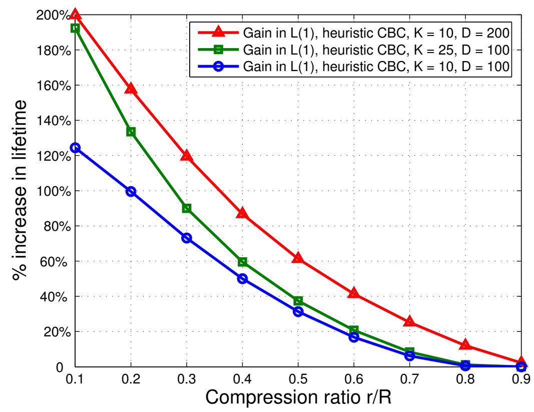

Fig. 12. Percentage increase (relative to no compression) in the lifetime of the node who dies first versus compression ratio when the heuristic CBC scheme is applied. The cluster size is $D = \{100, 200\ m\}$ and the number of sensors/cluster is $K = \{10, 25\}$. The energy model is: $E_a = 100\text{pJ/bit/m}^2, E_e = 50\text{nJ/bit}$ and $E_c = 5\text{nJ/bit}$. Packet loss is assumed to be negligible.

no node carries out compression for different values of K and D, i.e., $K = 10,\ 25$ and $D = 100\text{m},\ 200\text{m}$. The percentage increase is plotted against the compression ratio. As can be seen, the gain in lifetime increases in the number of nodes per cluster. This can be explained by the fact that, when there are more nodes in each cluster, the distance among them gets shorter, each node has more options on which node it can used to compress its data. At the same time, when the cluster size D is increased, the performance gain also increases. This is because with a larger cluster size, the average distance from sensors to the cluster head increases and in Section 4.2, we have shown that this increase in the distance will give node more incentive to jointly compress data.

Next, we look at how the performance gain of the heuristic scheme depends on the energy model of sensor nodes. In particular, we let the value of electronic energy, i.e., E_e vary from 10 to 100nJ/bit while still keeping the amplifier and processing energy unchanged. In Fig. 13, we plot the percentage increase for the lifetime of the node who dies first versus the compression ratio for $E_e = 10,\ 50$ and 100nJ/bit. As expected, the gain decreases in E_e. However, even when $E_e = 100\text{nJ/bit}$, the gain is still about 30% for the compression ratio of 0.5.

In Figs. 9 - 13, we assume that, for each packet, the header size is negligible, relative to the data payload. This assumption is reasonable in our cluster-based sensor network model, as the one-hop communications between sensors and cluster heads is simple and does not require much signaling. Nevertheless, there might be special cases in which a significant portion of packet size is used for header transmission, for example, to signal some sensors' specific information. In Fig. 14, we plot the percentage increase in the lifetime of the sensor who dies first for different header sizes. In particular, the head size is equal to 5%, 10%, and 20% of the data payload (of 400 bits). As can be observed, the performance gain of our heuristic CBC scheme decreases when the header size increases. This is expected

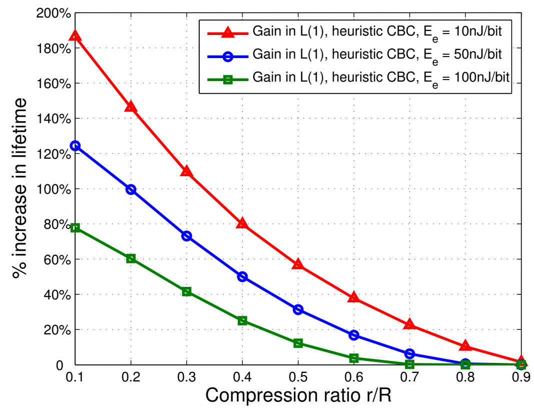

Fig. 13. Percentage increase (relative to no compression) in the lifetime of the node who dies first versus compression ratio when the heuristic CBC scheme is applied. There are $K = 10$ nodes in each cluster and the energy model is: $E_a = 100\text{pJ/bit/m}^2$, $E_c = 5\text{nJ/bit}$ and E_e takes the values $\{10, 50, 100\text{nJ/bit}\}$. Packet loss is assumed to be negligible.

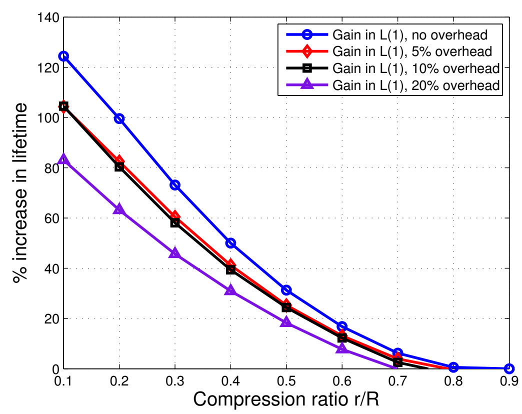

Fig. 14. Percentage increases in lifetime of node who dies first when the heuristic CBC scheme is applied for different overhead costs. There are $K = 10$ nodes in each cluster and the energy model is: $E_a = 100\text{pJ/bit/m}^2, E_e =$ 50nJ/bit and $E_c = 5\text{nJ/bit}$.

given our assumption that packet header cannot be compressed. However, even when the header size is relativly large, i.e., 20% of payload, the gain of heuristic CBC is still significant at a certain range of compression ratio.

In Section 8.1, we suggest that to deal with the scenario when the radio startup cost is significant, an extra constraint, i.e., no more than one node can compress based on any node can be enforced. In Fig. 15, we look at the performance of the heuristic scheme with this extra constraint. Here we plot the percentage increase in different components of the lifetime vector L. It is obvious that by enforcing the extra constraint, the increase in sensor lifetime is less, however, as can be seen in

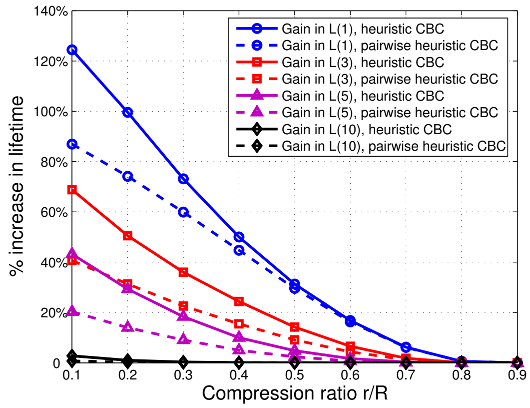

Fig. 15. Percentage increases in $L(1)$, $L(3)$, $L(5)$, $L(10)$ versus compression ratio when the heuristic CBC and pairwise heuristic CBC schemes are applied. Pairwise heuristic CBC schemes alow at most one node to compress based on any particular node. There are $K = 10$ nodes in each cluster and the energy model is: $E_a = 100\text{pJ/bit/m}^2, E_e = \ 50\text{nJ/bit}$ and $E_c = 5\text{nJ/bit}$. Packet loss is assumed to be negligible.

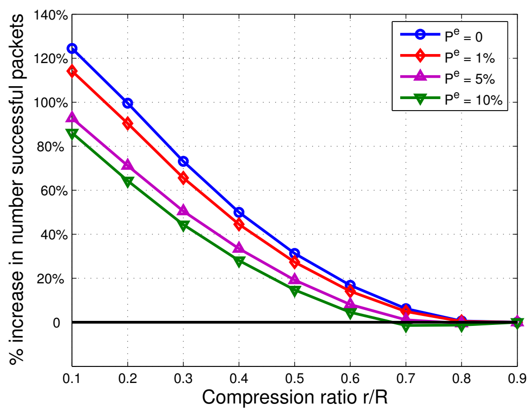

Fig. 16. Percentage increases in the number of packets successfully transmitted for the node who dies first when the heuristic CBC scheme is applied. There are $K = 10$ nodes in each cluster and the energy model is: $E_a = 100\text{pJ/bit/m}^2, E_e = \ 50\text{nJ/bit}$ and $E_c = 5\text{nJ/bit}$. Packet loss processes of different transmissions are independent and with the same packet loss probability P^e. The points at which performance curves cut the zero-level line is where the CBC approach does not give any performance improvement.

Fig. 15, the gain in $L(1)$ is still very significant. In particular, when the compression ratio is 0.5, applying the modified heuristic scheme results in 30% increase in the lifetime of the node who dies first.

Let us look at how our heuristic CBC schemes perform under packet loss due to transmission errors. Here, we assume that the packet loss processes for the transmissions between different pairs of nodes in the cluster are independent (note that

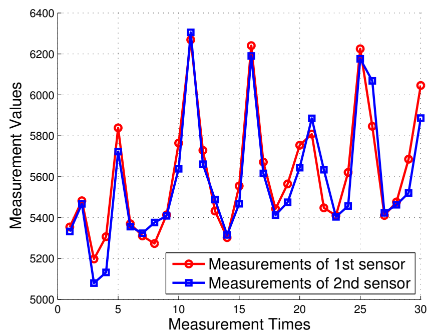

Fig. 17. Temperature readings of two nearby sensors in the project Habitat Monitoring on Great Duck Island.

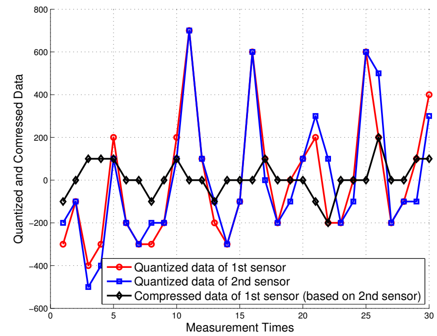

Fig. 18. For each sensor reading, the deviation from the mean is quantized in step of 100. First sensor carry out differential encoding based on the data of second sensor.

this is the worst case assumption) and with the same packet loss probability denoted by P^e. In Fig. 16, we plot the performance of our heuristic CBC schemes, in terms of the percentage increase in the total number of packet successfully transmitted for the node who dies first. Different packet loss probabilities are used, i.e., $P^e = 0,\ 1\%,\ 5\%,\ 10\%$. As can be seen, even when the packet loss probability is relatively high, i.e., at 10%, the performance gain for the node who dies first is still very significant. This suggests that our CBC approach is robust against packet loss due to transmission errors. Note also in Fig. 16 that the points at which performance curves cut the zero-level line is where the CBC approach does not give any performance improvement.

Finally, to examine the implementability of the CBC approach, we consider applying it to a practical sensing scenario. To do so, we use the measured data from the Great Duck Island Habitat Monitoring porject [Szewczyk et al. 2004]. In this project, researchers deployed a wide area sensor network (using Crossbow's Mica2Dot sensor node platform) and collected measurements of temperature, humidity, and barometric pressure over a period of four months. In particular, each sensor repeatedly sampled and relayed their readings to a control center on the island. To examine the spatial correlation, we consider the raw temperature readings of two sensors (specifically nodes 121 and 122) located 23 m apart. We noticed that the sensors did not make measurements at exact periodic intervals but the average was once every ten minutes. For our purpose, we aggregrated this data so that each sensor reported to the cluster head three times per day. First, we plot the aggregrated readings for the two sensors in Fig. 17 for a period of ten days. As can be seen, the data collected by the two sensors are highly correlated. Next, in Fig. 18, for each sensor reading, we calculate the deviation from the mean and then evenly quantize in steps of 100. We assume that the two sensors have to send these quantized readings to a cluster head at some distance away. For such a scenario, a simple way to exploit CBC is to allow one sensor to carry out differential compression based on the data of the other sensor. In Fig. 18, we also

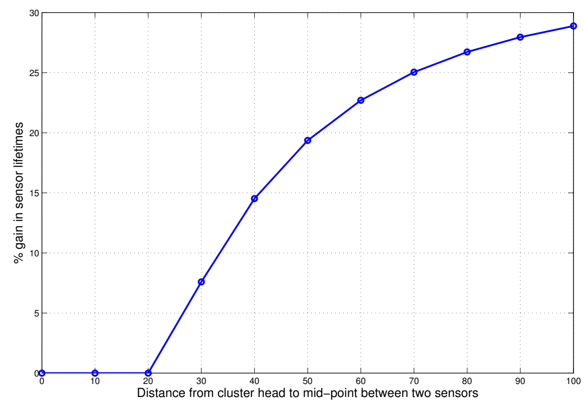

Fig. 19. Lifetime improvement versus distance toward cluster head for two tested sensors. Differential encoding is assumed with compression ratio of 50%. The cluster head is equi-distance from the two sensors and the distance on x-axis is from cluster head to the mid-point between the two sensors.

plot the differential values that the first sensor need to transmit if it can receive and compress based on the data of the second sensor. As can be seen in Fig. 18, the quantized readings of the two sensors are between −800 to 800, which can be represented by 4 bits. On the other hand, the differential values between the two corresponding measurements can be represented using 2 bits. This means that a compression ratio of 50% can be achieved.

With the compression ratio of 50%, we can use the plots in Figs. 10 - 16 to estimate the gain in sensor lifetimes for different system parameters. In Fig. 19, we consider the gain of the CBC approach when the distance between the two sensors is 23m, the cluster head is located equi-distant from the two sensor, and the compression ratio is 50%. Here optimal CBC, obtained by solving a linear programming, is employed. When the distance from cluster head and sensors are varried, the gain in sensor lifetime is plotted in Fig. 19. As can be seen, the gain increases when the distance from sensors toward the cluster head increases.

10. CONCLUSION

In this paper, we propose a novel approach in which the inherent broadcast nature of the wireless medium is exploited by sensor nodes to carry out joint data compression and conserve energy. This is different from the usual abstraction of a communication network by a communication graph, in which nodes interact in a point-to-point fashion.

Our metric of interest is sensor network lifetime. We first present algorithms which optimize the lifetime vector of the network, meaning that any other algorithm will not increase the lifetime of the node which dies first. We then propose a heuristic algorithm which has significantly lower computational complexity with near optimal performance. Important characteristics of wireless sensor networks such as node startup cost and packet loss due to transmission errors are also considered. Extensive numerical results are presented to support our approach.

Taking a broader view, our paper highlights two important issues in designing

WSNs. Firstly, it is important to not over simplify the network model when designing energy-constrained WSNs. One example of over-simplification, we believe, is the popular point-to-point link abstraction of the wireless medium. Secondly, our results show the importance of exploiting the opportunity to collaborate in wireless settings. This is because on one hand, different nodes in a WSN experience different performance/resource constraints while on the other hand, what is important is not a sensor's individual performance, but rather the network's collective performance.

We are currently looking at extending our approach to non-cluster based networks and designing scalable efficient heuristic algorithms.

REFERENCES

Agnihotri, S., Nuggehalli, P., and Jamadagni, H. S. 2005. On maximizing lifetime of a sensor network. In *Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*.

Akyildiz, I., Su, W., Sankarasubramaniam, Y., and Cayirci, E. 2002. A survey on sensor networks. *IEEE Comm. Mag. 40,* 8 (Aug.), 102–114.

Arisha, K. A., Youssef, M., and Younis, M. 2003. Energy-aware tdma-based mac for sensor networks. *Computer Networks Journal (Elsevier) 43,* 5 (Dec.), 539–694.

Chou, J., Petrovic, D., and Ramchandran, K. 2003. A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks. In *Proceedings of IEEE INFOCOM 2003*. 1054–1062.

Cristescu, R., Lozano, B., and Vetterli, M. 2004. On network correlated data gathering. In *Proceedings of IEEE INFOCOM 2004*. 2571–2582.

Deshpande, A., Guestrin, C., Madden, S. R., Hellerstein, J. M., and Hong, W. 2004. Model-driven data acquisition in sensor networks. In *Proceedings of 30th International Conference on Very Large Data Bases Conference (VLDB)*. 588–599.

Goel, A. and Estrin, D. 2003. Simultaneous optimization for concave cost: Single sink aggregation or single source buy-at-bulk. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms*. 499–505.

Heinzelman, W. B., Chandrakasan, A. P., and Balakrishnan, H. 2000. Energy-efficient communication protocol for wireless sensor networks. In *Proceedings of the Hawaii International Conference on System Sciences (HICSS)*. 1–10.

Heinzelman, W. B., Chandrakasan, A. P., and Balakrishnan, H. 2002. An application-specific protocol architecture for wireless microsensor networks. *IEEE Trans. Wireless Comm. 1,* 4 (Oct.), 660–670.

Hiller, F. and Lieberman, G. 1995. *Introduction to Operation Research*. McGraw-Hill Inc.

Hoang, A. T. and Motani, M. 2005a. Collaborative broadcasting and compression in cluster-based wireless sensor networks. In *Proceedings 2nd European Workshop on Wireless Sensor Networks (EWSN) 2005*. 197–206.

Hoang, A. T. and Motani, M. 2005b. Exploiting wireless broadcast in spatially correlated sensor networks. In *In Proceedings of IEEE International Conference on Communications (ICC) 2005*.

Intanagonwiwat, C., Govindan, R., and Estrin, D. 2000. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *Proceedings of the 6th ACM Annual International Conference on Mobile Computing and Networking (MobiCom 2000)*. 56–67.

Jindal, A. and Psounis, K. 2004. Modeling spatially-correlated sensor network data. In *Proceedings of IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON) 2004*. 162–171.

Krishnamachari, B., Estrin, D., and Wicker, S. 2002. Modeling data centric routing in wireless sensor networks. In *USC Computer Engineering Technical Report CENG*. 12–44.

Melo, D. and Liu, M. 2002. Analysis of energy consumption and lifetime of heterogeneous wireless sensor networks. In *Proceedings of IEEE Globecom 2002*.

Pattem, S., Krishnamachari, B., and Govindan, R. 2004. The impact of spatial correlation on routing with compression in wireless sensor networks. In *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*. ACM Press, New York, 28–35.

Pradhan, S. and Ramchandran, K. 1999. Distributed source coding using syndromes: Design and construction. In *Proceedings of the Data Compression Conference (DCC)*. IEEE Computer Society, Washington, DC, USA, 158.

Raghunathan, V., Schurgers, C., Park, S., and Srivastava, M. 2002. Energy aware wireless microsensor networks. *IEEE Signal Processing Magazine 19,* 2 (Mar.), 40–50.

Rappaport, T. 1996. *Ad hoc Mobile Wireless Networks: Protocols and Systems*. Prentice Hall.

Scaglione, A. and Servetto, S. 2002. On the interdependence of routing and data compression in multihop-sensor networks. In *Proceedings of the 8th ACM International Conference on Mobile Computing and Networking (MobiCom 2002)*. 140–147.

Shih, E., Cho, S., Ickes, N., Min, R., Sinha, A., Wang, A., and Chandrakasan, A. 2001. Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. In *Proceedings of the 7th ACM International Conference on Mobile Computing and Networking (MobiCom 2001)*. ACM Press, 272–287.

Slepian, D. and Wolf, K. 1973. Noiseless encoding of correlated information sources. *IEEE Trans. on Inform. Theory IT-19,* 4 (July), 471–480.

Szewczyk, R., Mainwaring, A., Polastre, J., and Culler, D. 2004. An analysis of a large scale habitat monitoring application. In *In Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems (SenSys)*.

Vuran, M. C. and Akyildiz, I. F. 2006. Spatial correlation-based collaborative medium access control in wireless sensor networks. *to appear in IEEE/ACM Transactions on Networking*.

Wyner, A. D. and Ziv, J. 1976. The rate-distortion function for source coding with side information at decoder. *IEEE Trans. Inform. Theory IT-22,* 1 (Jan.), 1–10.