

Balanced Aggregation Trees for Routing Correlated Data in Wireless Sensor Networks

Huiyu Luo and Gregory Pottie
Electrical Engineering Department
University of California at Los Angeles
Los Angeles, Ca, 90095, USA
Email: {huiyu, pottie}@ee.ucla.edu

Abstract—In a sensor network, the data collected by different sensors are often correlated because they are observations of related phenomena. This property has prompted many researchers to propose data centric routing [2]–[4] to reduce the communication cost [1]. In this paper, we design heuristic algorithms for combined routing and source coding with explicit side information. We build a data rate model upon the observation that in many physical situations the side information that provides the most coding gain comes from a small number of nearby sensors. Based on this model, we formulate a problem to determine the optimal routes for transmitting data to the fusion center. The overall optimization is NP hard because it has minimum Steiner tree as a sub-problem. We then propose a heuristic algorithm that is inspired by balanced trees that have small total weights and reasonable distance from each sensor to the fusion center [5]. The average performance of the algorithm is analyzed and compared to other routing methods through simulations.

I. INTRODUCTION

To reduce communication cost and preserve node energy, many researchers have proposed data centric routing [2]–[4] that makes use of data aggregation in wireless sensor networks. One type of aggregation is to remove redundancy among sensor measurements. Consider the general problem of sampling a distributed field using a wireless sensor network. The sensor measurements are coded and transmitted back to the fusion center, and used to reconstruct the field under some distortion constraint. There is likely to be a great deal of redundancy in the data collected by different sensors, since they are observations of related physical phenomena. Suppose data stream X_i is generated at sensor $i = 1, 2, \dots, n$. We assume X_i satisfies the ergodic condition, so the results of statistical probability theory can be applied here. If the objective is to minimize the total communication power while recovering these data streams subject to some distortion constraint $d(X_1, \dots, X_n; \hat{X}_1, \dots, \hat{X}_n) \leq D$, the optimization problem is:

$$\begin{aligned} & \min \sum_{i=1}^n P_i \\ \text{subject to: } & d(X_1, \dots, X_n; \hat{X}_1, \dots, \hat{X}_n) \leq D \text{ is} \\ & \text{achievable under power budget } (P_1, \dots, P_n). \end{aligned}$$

The bounds on the set of admissible power allocations are prescribed by the complete network information theory [6], which remains an open problem. As a result, the sub-optimal

approach that separately consider channel coding, source coding, and routing is often used in practice.

Source coding in sensor networks is generally lossy. Although high resolution lossy coding resembles Slepian-Wolf coding [7], general network distortion coding remains an open problem. Also, distributed source coding schemes with performance near information theoretic bounds often employ long blocks of data, which results in high complexity and long delays. In this paper, we consider source coding with explicit side information. In other words, only when the side information is available at both the encoder and decoder, can it be used to reduce the data rate. In practice, a lossy encoder (such as the DPCM encoder in [8]) can be employed at each sensor to compress its data using incoming flows as explicit side information. Due to the lossy nature of the scheme, it is desirable to postulate that the data can be compressed only at the sensor that generates them or at one local fusion center so that the final distortion is well controlled. An alternative approach is to quantize the analog signal locally then conduct joint entropy coding with incoming data flows using for example a Lempel-Ziv encoder. This appears sub-optimal compared to the first scheme. However, it allows data to be coded anywhere without violating the distortion constraint, and can be potentially usable in other communication networks. In this paper, the more flexible approach of entropy coding is adopted. We can also conceive a method that combines the previous two schemes. Yet the added complexity has to be warranted by the additional coding gain, which may depend on practical situations.

We use edge weights to represent the cost (e.g. power) of achieving unit data rate under some given channel transition matrix, modulation, and channel coding scheme. These weights are assumed to be given a priori. The problem now becomes how to construct the transmission routes based on the network topology and source correlation such that the given cost function is optimized.

In many situations, data aggregation is possible because the fusion center (end user) is interested only in some fused function. For example, in [9], only the direction of arrival estimation needs to be transmitted from each sensor sub-array to the fusion center to locate an acoustic source. In these cases, the way that data aggregation and communication is

carried out is highly dependent on the specific application. This problem under the broad title of distributed data fusion is by itself an area under active research.

There are numerous papers devoted to distributed source coding and network routing in sensor networks. We sample a few. The information theoretic bounds on distributed source coding in sensor networks is analyzed as the quadratic Gaussian CEO problem in [10]. [11] discusses a framework for performing distributed compression in a dense sensor network. The interaction of source coding and routing is discussed from the viewpoint of information theory in [12] and [13]. Clustering methods have been used by some researchers [14] to perform data aggregation at the cluster head before transmitting them to the fusion center. In [3], a diffusion type routing paradigm that attaches attribute-value pairs to data packets is proposed to facilitate the in-network data fusion. [2] gives a comparison of data-centric and address-centric routing. The correlated data routing problem is studied in [4]. A similar optimization problem is also the subject of [15], where a grossly simplified data model is assumed.

The rest of the paper is organized as follows. In section II, we discuss network flow and source coding models. Based on these models, the optimization problem is formulated and shown to be NP hard in section III. A sub-optimal algorithm is presented in section IV, where simulation results are also given. Section V concludes the paper.

II. NETWORK MODELS

A. Network flows

The sensor network is modelled as a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. The node set \mathcal{N} consists of a set \mathcal{N}_s of n sensors and a special node t acting as the fusion center. Denote by \mathcal{N}_a the set of active sensors that produce data. Both active and non-active sensors can be relays. The edge set \mathcal{E} includes m communication links. We assume that all the communication links are bi-directional and symmetric. If this is not the case, the network can be modelled as a directed graph, and we believe the ideas in this paper will apply similarly. The network is assumed to be connected such that at least one path exists from each sensor to t . A weight c_e is associated with each $e \in \mathcal{E}$, and it represents the cost (e.g. power) of transmitting data at unit rate across the edge e . These weights are assumed to be given a priori. The flow f_e is defined as the rate at which data is transmitted across the edge $e \in \mathcal{E}$. If there is the need to identify the origin of the flow, we use f^k to indicate that the data is generated by sensor k . Clearly, $f_e = \sum_{k \in \mathcal{N}_a} f_e^k$.

The objective of our optimization is to minimize the cost C of routing all the data from active sensors to t .

$$C = \sum_{e \in \mathcal{E}} c_e f_e \quad (1)$$

B. Source coding with explicit side information

Denote by X_i the data stream produced by sensor i . (It is assumed that X_i has been quantized and has a discrete alphabet.) In general, data flow f_e across edge e is the joint transmission of flows originated at different sensors, some of

which may be jointly coded and some may not. Supposing data streams X_1, X_2, \dots, X_K are jointly coded and transmitted across e , we can decompose the rate using the chain rule:

$$H(X_1, X_2, \dots, X_K) = \sum_{k=1}^K f_e^k$$

$$f_e^k = H(X_k | X_1, X_2, \dots, X_{k-1})$$

The rate f_e^k is a function of available side information. Fully characterizing such a data rate function requires an exponential amount of information. For example, the availability of $X_i, i = 1, \dots, k-1$ produces 2^{k-1} possibilities. To simplify the problem, we make the following assumptions: (1) data stream X_k is highly correlated with the data of a set of sensors \mathcal{H}_k , and only the data from the sensor in \mathcal{H}_k can be used to effectively compress X_k ; (2) each sensor uses side information from at most one other sensor, and the coding gain of compressing X_k is the same for any $i \in \mathcal{H}_k$. Therefore,

$$f_e^k = \begin{cases} b_0^k & \text{no side information} \\ b_1^k & \text{jointly coded with } i \in \mathcal{H}_k \end{cases} \quad (2)$$

where $b_0^k = H(X_k)$ is the data rate without side information, and $b_1^k = H(X_k | X_i)$ is the rate when X_k is jointly coded with any data stream $X_i, i \in \mathcal{H}_k$.

C. Discussion

Only when flows are jointly coded, do they need to be bundled in transmission. Thus, the overall routing structure is not necessarily a tree. (We will explain later why we still call our route a tree.) For example, in Fig. 1, if flows f_{21}^2 and f_{31}^3 are not jointly coded, they can split to take different paths in ensuing transmissions $f_{14} = H(X_2)$ and $f_{15} = H(X_3)$. We point out that in data-centric routing, trees are not necessarily optimal. For more discussion on this and another interesting strategy that uses non-tree routing structure, the readers are referred to [16].

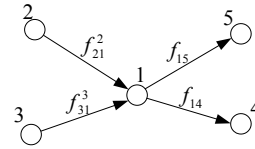


Fig. 1. Data flows in the network.

As we have mentioned, our model allows X_i to be compressed not only at i but also enroute to t . For instance, in Fig. 1, if flows f_{21}^2 and f_{31}^3 are jointly coded at node 1, we can have $f_{14} = H(X_2) + H(X_3 | X_2)$ and $f_{15} = 0$.

In many physical situations, sensor measurements are highly correlated only in a small neighborhood. In others, although a large number of sensors have similar measurements, the reproduction fidelity constraints often permit thinning the number of active sensors so that again only a small number of sensors have high correlation. Moreover, finding out the coding gain of other sensors' data and then performing joint

source coding incurs cost. The gain of distant helpers are often not enough to outweigh these costs. Therefore, it is in many situations reasonable to assume that \mathcal{H}_k includes only a small number of sensors near k .

Using side information from at most one sensor appears restrictive. Nonetheless, the coding gain by additional helpers is often significantly less than that of the first helper due to the correlation among side information. Hence, our model may be considered as the first order approximation of a complete model. Also, using more than one helper increases the complexity of the model and optimization.

In practice, the values of b_0^k, b_1^k and members in \mathcal{H}_k for each k in our rate model need to be determined during a training period. Compression schemes similar to the one in [8] can be used to obtain such information. Since this training process often involves transmissions among sensors in a small neighborhood, the cost should not be excessively high.

Finally, to enforce the chain rule, we label the set of sensors \mathcal{N}_a according to an arbitrary order. This means: (1) there is a unique relation $i < j$ defined for any pair $i, j \in \mathcal{N}_a$; (2) if $i < j$ and $j < k$, then $i < k$ for $i, j, k \in \mathcal{N}_a$. We impose the restriction that i can be in set \mathcal{H}_j only when $i < j$.

III. PROBLEM FORMULATION

We state the problem in a standard two-part format.

Combined Routing and Entropy Coding (CREC)

GIVEN: A graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ with weight c_e defined on each edge $e \in \mathcal{E}$, a special node $t \in \mathcal{N}$, a set \mathcal{H}_k and data rate f_e^k defined as in Eq. (2) for each sensor $k \in \mathcal{N}_s = \mathcal{N} \setminus \{t\}$.

FIND: A set of routes such that the cost of routing all the data to t , $C = \sum_{e \in \mathcal{E}} c_e f_e$, using joint entropy coding is minimized.

Proposition: CREC is NP hard.

Proof: We prove this by showing that one of the subproblems of CREC is the minimum Steiner tree problem.

Assume $i \in \mathcal{H}_j$ whenever $i < j$ and $i, j \in \mathcal{N}_a$. Define the rate function $f_e^i = 1$ without side information information, and $f_e^i = 0$ with side information for any sensor in \mathcal{H}_i . We first show that the optimal route for this problem must be a tree. Suppose that the optimal solution is not a tree. Since in any solution route there is at least one path from each $i \in \mathcal{N}_a$ to t , we can find a tree that is embedded in the optimal solution and connects all the active sensors to t . Using this tree as the transmission route, we can arrange the data flow and joint coding in a way such that the data rate f_e on any edge of the tree is 1. Thus, the total cost is simply the weight sum of the edges on the tree, which is less than that of the optimal solution. This contradiction proves that the set of optimal routes must constitute a tree. As $f_e = 1$ on every edge of the routing tree, finding the optimal tree that routes all the data to the fusion center is equivalent to constructing the minimum Steiner tree that connects t and the sensors in \mathcal{N}_a , which is a well-known NP hard problem. Therefore, our problem is also NP hard. Q.E.D.

Since finding the exact solution in polynomial time is unlikely, we turn to heuristics in the next section.

IV. BALANCED AGGREGATION TREE

A. Motivation

In this section, we propose an approximation algorithm that is inspired by the idea of balancing shortest path trees and trees with small total weights [5]. To motivate the algorithm, we decompose the flow f_e^k into two parts.

$$f_e^k = u_e^k + v_e^k$$

When $f_e^k = 0$, $u_e^k = v_e^k = 0$. If $f_e^k > 0$, we define

$$u_e^k = \begin{cases} b_0^k - b_1^k & \text{no side information} \\ 0 & \text{jointly coded with } i \in \mathcal{H}_k \end{cases}$$

Thus, u_e^k represents the portion of f_e^k that is independent of the side information, and v_e^k is the part that is compressible by helper's data. Accordingly, the total cost of routing data to t can be decomposed into two parts $C = C_1 + C_2$, where

$$C_1 = \sum_{e \in \mathcal{E}} \sum_{k \in \mathcal{N}_a} c_e u_e^k, \quad C_2 = \sum_{e \in \mathcal{E}} \sum_{k \in \mathcal{N}_a} c_e v_e^k$$

Consider for a moment minimizing the costs C_1 and C_2 separately. C_1 is minimized when the route is a set of shortest paths. On the other hand, to obtain a small C_2 , we should try to jointly code X_k and $X_i, i \in \mathcal{H}_k$, and merges flows using routes that have small weights. This resembles a Steiner tree problem, but each aggregation involves only a subset of active sensors. We apply this to the two extreme cases of minimizing C . When coding gain is small ($b_1^k \gg b_0^k - b_1^k$ for $k \in \mathcal{N}_a$), the aggregation tree is expected to be close to a sub-tree of SPT. Whereas, when there is substantial coding gain ($b_1^k \ll b_0^k - b_1^k$), the focus is on achieving aggregation with small routing cost. For the general case of varying coding gains, we speculate that an approximation to the optimal solution can be obtained by constructing balanced aggregation routes that have small total weights and reasonable distance from each sensor to the fusion center, and the appropriate balance is struck based on the relative values of b_0^k and $b_1^k, k \in \mathcal{N}_a$.

B. Constructing balanced paths

We first examine how to route a sensor's data to t using an existing path while taking into account the data compression. In Fig. 2, there is a path \mathcal{P}_k connecting the active sensor k to t . Specifically, $\mathcal{P}_k = \{k, (k, v_1), v_1, \dots, v_p, (v_p, t), t\}$ consists of a sequence of nodes and edges on the path. Denote by d_{uv} ($u, v \in \mathcal{N}$) the shortest distance from u to v (i.e. the weight sum of edges on the shortest path from u to v), and $d_{uv}^{\mathcal{P}_k}$ ($u, v \in \mathcal{P}_k$) the distance from u to v along path \mathcal{P}_k . Define $d_u = d_{ut}$ and $d_u^{\mathcal{P}_k} = d_{ut}^{\mathcal{P}_k}$. We want to find a path to route the data of active sensor i to t such that the resulting cost is minimized. This is equivalent to determining an aggregation node $j \in \mathcal{P}_k$ where the two flows f^k and f^i joins one another. There are three possible situations.

- (1) The data streams at k and i are uncorrelated. Without jointly coding f^i and f^k , the optimal path for routing f^i is the shortest path from i to t .

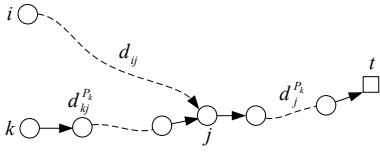


Fig. 2. Construct a balanced path for i .

- (2) $k \in \mathcal{H}_i$. Assume the two flows merge at node $j \in \mathcal{P}_k$. The additional cost of routing i through j is

$$C_{ij} = b_0^i d_{ij} + b_1^j d_j^{P_k}$$

- (3) $i \in \mathcal{H}_k$. The additional cost of routing i through j is

$$C_{ij} = (b_1^k - b_0^k) d_j^{P_k} + b_0^i (d_{ij} + d_j^{P_k})$$

For (2) and (3), we choose $\arg\{\min_{j \in \mathcal{P}_k} C_{ij}\}$ as the aggregation node and define $C_{ik} = \min_{j \in \mathcal{P}_k} C_{ij}$.

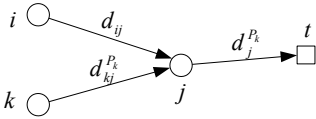


Fig. 3. An aggregation tree constructed from Fig. 2.

As we discussed in section II, the resulting route may not be a tree. For example, the path from i to j may have used some nodes in \mathcal{P}_k as relays. However, the routes can be transformed into a tree as follows. Remove all the nodes except for t , aggregation nodes, and active sensors which has a path to the fusion center. Trace the path from each active sensor, say k , to the fusion center. Each time f^k is being entropy coded at k or some aggregation node j with another flow, a directed edge is created from k or j to f^k 's next aggregation node. If there is no aggregation node where f^k is coded again, the edge terminates at t . Remove redundant edges, and assign the corresponding distances on the path as weights to the edges. For Fig. 2, the resulting tree is shown in Fig. 3. Note that each edge carries a constant flow rate, and the aggregation tree captures the essential picture of how the data aggregation takes place in the network.

C. Balanced aggregation tree

Our algorithm involves successive steps of adding the routes of active sensors to the aggregation tree. Each time, the newly added sensor has the smallest additional cost among all the remaining sensors. We state our algorithm as follows.

Balanced Aggregation Tree (BAT)

Given a graph \mathcal{G} with the edge weights and data rate function determined, define sets $\mathcal{U} = \mathcal{N}_a$ and $\mathcal{V} = \emptyset$. Set $C = 0$. Carry out the following steps.

- (1) Find the shortest path from each active sensor $i \in \mathcal{N}_a$ to t . Denote by I the sensor with minimum $C_i = b_0^i d_i$. Add I to set \mathcal{V} , and its path to the solution route.
- (2) $C = C + C_I$. Remove I from \mathcal{U} . If \mathcal{U} is empty, stop the algorithm.

- (3) For each sensor $i \in \mathcal{U}$, find the minimum C_{ik} resulted from merging f^i with each flow f^k , $k \in \mathcal{V}$. Compute

$$C_i = \min\{\min_{k \in \mathcal{V}} C_{ik}, b_0^i d_i\}$$

- (4) Find $I = \arg\{\min_{i \in \mathcal{U}} C_i\}$. Add the path of I to the solution route.
- (5) Add I to \mathcal{V} if f^I has not been compressed. If I provides side information for coding $k \in \mathcal{V}$, remove k from \mathcal{V} . Return to step (2).

For the special instance that results in the minimum Steiner tree problem in section III, our algorithm collapses to the shortest path heuristic given in [17]. For a set of sensors with uncorrelated data, it builds the shortest path from each active sensor to the fusion center. The bottleneck of the algorithm is on constructing shortest path trees. Using a Dijkstra's algorithm, it runs in $O(n_a m \log n)$ time for a sparse network. (n_a is the number of active sensors.)

D. Simulations

$(n + 1)$ nodes including t and n sensors are placed in an $n_d \times n_d$ square, where $n_d = \lceil \sqrt{n + 1} \rceil$. Supposing \tilde{x}_i and \tilde{y}_i , $i = 1, \dots, n + 1$, are random variables that are uniformly distributed in $[0, 1]$, the coordinates of node i is given by:

$$\begin{aligned} x_i &= \lfloor (i \bmod n_d) - 1 \rfloor + \tilde{x}_i \\ y_i &= \lfloor (i - 1) / n_d \rfloor + \tilde{y}_i \end{aligned}$$

Define r_c the transmission radius. If two nodes are no more than r_c away from each other, direct communication between the two nodes is allowed. Denote by d_e the length of edge e . When $d_e \leq r_c$, the edge weight c_e is proportional to d_e^α , where α is the path loss factor. We choose α to be 2. A typical

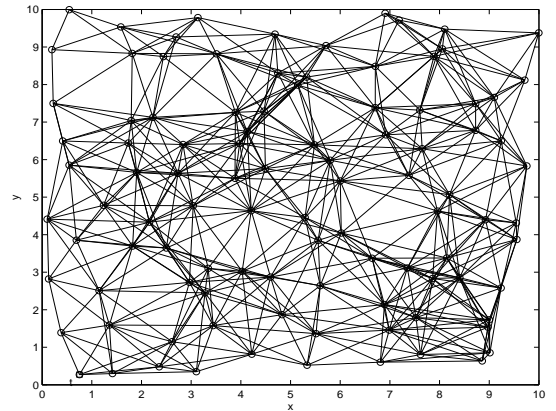


Fig. 4. A network of 100 nodes with $r_c = \sqrt{5}$.

100 node network constructed in this manner is depicted in Fig. 4.

we assume that all the sensors are active. Any sensor j ($j < i$) that is no more than r_d away from i has a probability of 0.5 to be in the \mathcal{H}_i . For simplicity, we assume the data rate function is the same for all the sensors, and $b_0^k = b_0$, $b_1^k = \beta b_0$, where $0 \leq \beta \leq 1$.

Besides the balanced aggregation tree, we consider three other strategies for routing and data aggregation. The first two both use shortest path trees but one with data compression and one without. Their total costs are denoted by C_{sptc} and C_{spt} accordingly. The third one is a clustering method that forms local groups of 4 to 9 sensors depending on the network size, and data are aggregated at the cluster head before transmitting to the fusion center. Denote by C_{bat} and $C_{cluster}$ the total costs using BAT and clustering methods. We define the performance ratios as:

$$\mu_s = \frac{C_{sptc}}{C_{spt}}, \quad \mu_c = \frac{C_{cluster}}{C_{spt}}, \quad \mu_b = \frac{C_{bat}}{C_{spt}}$$

Different network sizes and β values are used in simulation, and the performance ratios are plotted in Fig. 5 and 6.

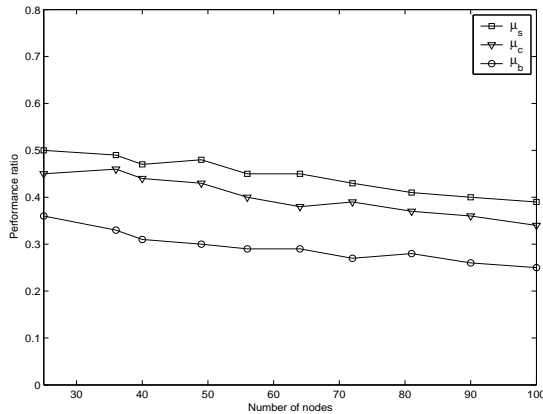


Fig. 5. Performance ratios versus network size for $\beta = 0.1$.

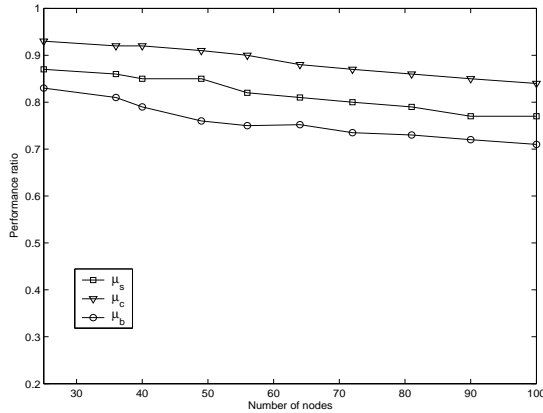


Fig. 6. Performance ratios versus network size for $\beta = 0.6$.

All three methods with data compression achieve appreciable gains over the address centric approach especially when coding gain is high. When $\beta = 0.1$ (high coding gain), clustering method outperforms shortest path tree with compression. When $\beta = 0.6$, the opposite is true. In both situations, BAT achieves the minimum cost. This shows that the performance of BAT is relatively immune to coding gain variations, which is expected.

V. CONCLUSION

Our study continues the recent development of data-centric routing. The data transmissions are decomposed into individual flows originated at different sensors to build a simplified first order rate model. Based on this model, we propose a balanced aggregation tree algorithm for cost minimization. The balanced approach is attractive in that it automatically adjusts based on level of data aggregation. Simulations show the effectiveness of this method in varying coding gain situations.

Our ongoing work continues to explore the potential of this scheme in the following directions. First, we suspect a bound on the worst case performance of BAT in comparison to the optimal solution will be related to the number of elements in \mathcal{H}_k . Second, assuming that high correlation occurs in a small neighborhood, distributed algorithms can be devised with exchange of a few global control messages. Third, our model assumes that side information comes from at most one source. The performance loss caused by discarding additional side information can become severe either because the additional gain is comparable to that of first helper or large network sizes. We surmise that our algorithm here may serve as a preliminary step or sub-algorithm for a more complicated model that allows multiple helpers.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation (NSF) Grant #CCR-0121778.

REFERENCES

- [1] G. Pottie and W. Kaiser, "Wireless sensor networks," *Comm. ACM*, vol. 43, no. 5, pp. 51-58, May 2000.
- [2] B. Krishnamachari, D. Estrin, and S. Wicker, "Modelling data-centric routing in wireless sensor network," *USC Computer Engineering Technical Report CENG, 02-14*, 2002.
- [3] C. Intanagonwiwat, et al. "Directed diffusion for wireless sensor networking," *IEEE/ACM Trans. Networking*, vol. 11, no. 1, pp. 2-16, Feb 2003.
- [4] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, "On network correlated data gathering," *IEEE Infocom*, Hongkong, March 7-11, 2004.
- [5] S. Khuller, B. Raghavachari, and N. Young, "Balancing minimum spanning trees and shortest-path trees," *Algorithmica*, vol. 14, pp 305-321, 1995.
- [6] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, 1991.
- [7] R. Zamir and Toby Berger, "Multiterminal source coding with high resolution," *IEEE Trans. Inform. Theory*, vol. 45, no. 1, Jan 1999.
- [8] H. Luo, Y. Tong, and G. Pottie, "A two-stage DPCM scheme for wireless sensor networks," to appear in *ICASSP 2005*, Philadelphia, USA.
- [9] J. Chen et al. "Coherent acoustic array processing and localization on wireless sensor networks," *Proc. IEEE*, vol. 91, no. 8, Aug 2003.
- [10] H. Viswanathan and T. Berger, "The quadratic Gaussian CEO problem," *IEEE Trans. Inform. Theory*, vol. 43, no. 5, Sep 1997.
- [11] S. Pradhan, J. Kusuma, and K. Ramchandran, "Distributed compression in a dense microsensor network," *IEEE Sig. Proc. Mag.*, March, 2002.
- [12] A. Scaglione and S. Servetto, "On the interdependence of routing and data compression in multi-hop sensor networks," *Mobicom*, 2002.
- [13] S. Tatikonda and K. Ramchandran, "Coding and routing complexity in sensor networks," *ISIT*, 2004, Chicago, USA.
- [14] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," *IEEE Infocom*, 2003.
- [15] A. Goel and D. Estrin, "Simultaneous optimization for concave costs: single sink aggregation or single source buy-at-bulk," *ACM/SIMA Symposium on Discrete Algorithms*, 2003.
- [16] H. Luo and G. Pottie, "Routing explicit side information for data compression in wireless sensor networks," *DCOSS 2005*.
- [17] H. Takahashi and A. Matsuyama, "An approximate solution for the steiner problem in graphs," *Math. Japonica*, 24, No. 6, pp 573-577, 1980.