Designing Routes for Source Coding with Explicit Side Information in Sensor Networks

Huiyu Luo and Gregory Pottie, *Fellow, IEEE,*

***Abstract*—In this paper, we study the problem of designing routes for source coding with explicit side information (i.e. with side information at both the encoder and the decoder) in sensor networks. Two difficulties in constructing such data-centric routes [1]–[5] are the lack of reasonably practical data aggregation models and the high computational complexity resulting from the coupling of routing and in-network data fusion. Our data aggregation model is built upon the observation that in many physical situations the side information providing the most coding gain comes from a small number of nearby sensors. Based on this model, we formulate an optimization problem to minimize the communication cost, and show that finding the exact solution of this problem is NP-hard. Subsequently, two suboptimal algorithms are proposed. One is inspired by the balanced trees that have small total weights and reasonable distance from each sensor to the fusion center [6]. The other separately routes the explicit side information to achieve cost minimization. Bounds on the worst-case performance ratios of two methods to the optimal solution are derived for a special class of rate models, and simulations are conducted to shed light on their average behaviors.**

***Index Terms*—Data-centric routing, source coding, NP-hardness, shortest path tree, Steiner tree, maximum weight branching.**

I. Introduction

A. The problem and its motivation

The need to lower the communication cost in wireless sensor networks [7] has prompted many researchers to propose data-centric routing schemes that utilize in-network data fusion to reduce the transmission rate. There are two major difficulties in designing such routes. First, the lack of reasonably practical data aggregation models has led researchers to use overly simplified ones [1]–[5]. For example, these models generally assume that sensors perform the same aggregation function regardless of the origin of the fused data. As a remedy, [4] suggests looking into models in which data aggregation is not only a function of the number of sources but also the identity of the sources. Second, the resulting optimization problem is often NP-hard due to the coupling of routing and in-network data fusion [1], [2]. Hence, algorithms that find exact solutions in polynomial time are unlikely to exist. In this paper, we attempt to build network models that are computationally useful yet reasonably approximate reality and devise heuristic algorithms for the combined routing and source coding problem.

This work was supported by the NSF Grant CCR-0121778.

The authors are with the Department of Electrical Engineering, University of California at Los Angeles, 90095, USA (email: huiyu@ee.ucla.edu; pottie@ee.ucla.edu.)

Source coding in sensor networks is generally lossy. Although high resolution lossy coding resembles Slepian-Wolf coding [8], general network distortion coding remains an open problem. Also, distributed source coding schemes with performance near information theoretic bounds require knowledge of the source probability distribution and often employ long blocks of data, which results in high complexity and long delays. While distributed source coding remains an active research topic [9], [10], 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 the decoder, can it be used to help compress the data. In practice, a lossy encoder (such as the DPCM encoder in [11]) can be employed at each sensor to compress its data using incoming flows as explicit side information. Alternatively, we can quantize the analog signal locally. Then joint entropy coding is conducted on merged data flows using, for example, a Lempel-Ziv encoder. The latter scheme is more flexible in that data can be compressed at not only the node that generates the data but relays.

Another reason why we consider the joint compression-routing problem is that in many situations, data aggregation is possible because the fusion center (end user) is interested only in some fused function. For example, in [12], only the direction of arrival estimation needs to be transmitted from each sensor sub-array to the fusion center to locate an acoustic source. Although in these cases, the way that data fusion and communication is carried out is highly dependent on the specific application, the algorithms proposed in this paper can be useful if proper data aggregation models are available.

In this paper, two strategies are proposed. One is called the balanced aggregation scheme (BAS), and the other designated side information transmission (DSIT). To motivate the idea and give a preview of the paper, consider the example depicted in Fig. 1. The communication links between adjacent sensors

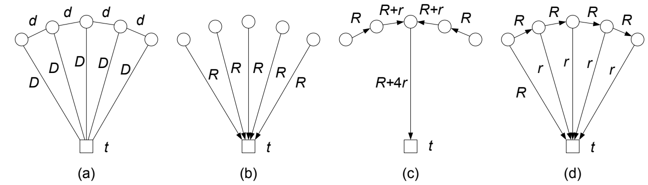


Fig. 1. Three routing strategies (t denotes the fusion center): (a) Network topology; (b) SPT; (c) BAS; (d) DSIT.

(circles) have the weight $c_e = d$, and the links connecting sensors to the fusion center (square) have the weight $c_e = D$. These weights indicate the cost (e.g. power) of transmitting data at unit rate via corresponding communication links. The rate at which each sensor needs to transmit to the fusion center is R without explicit side information and r if explicit side information from an adjacent sensor is available. Assume $r \ll R$ and $d \ll D$. The objective is to minimize the cost $C = \sum_e c_e f_e$ of routing all the data to the fusion center, where f_e is the rate at which data are transmitted across edge e. Consider the three strategies described in Fig. 1. In (b), the shortest path tree (SPT) with distance metric c_e is used. In (c), an example of BAS, before being routed to the fusion center, data are aggregated at relays to reduce the communication cost. In (d), an example of DSIT, each sensor except for the rightmost one transmits its data to the sensor at its right. This transmission provides explicit side information for data compression at the recipient and needs not to be relayed to the fusion center. Note that at least one sensor has to transmit at rate R to the fusion center for all the data to be correctly recovered. The costs of three strategies are: $C_{\text{SPT}} = 5RD$; $C_{\text{BAS}} = RD+4rD+4Rd+2rd$; $C_{\text{DSIT}} = RD+4rD+4Rd$. In this case, the performances of BAS and DSIT are both superior to that of SPT.

B. Related work

There has been much recent research activity on data-centric routing. In [13], the interdependence of routing and data compression is addressed from the viewpoint of information theory. Clustering methods have been used by some researchers to aggregate data at the cluster head before transmitting them to the fusion center [14], [15]. Since the cluster head is responsible for data aggregation and relaying, it consumes the most energy. Hence, dynamically electing nodes with more residual power to be cluster heads and evenly distributing energy consumption in the network is a major issue in these schemes. In [5], a diffusion type routing paradigm that attaches attribute-value pairs to data packets is proposed to facilitate the in-network data fusion. [4] studies an optimization problem formed for a special class of data models.

More closely related to our work are the correlated data routing problems studied in [1]–[3]. In [1], the authors gave a thorough comparison of data-centric and address-centric methods and a overview of recent effort in the field. [2] cast the data-centric routing as an optimization problem and sought solutions to it when different source coding schemes were used. However, with solution routes restricted to trees, their heuristics were limited, and no performance analysis was attempted. [3] discussed two routing algorithms, and we will see that our DSIT scheme resembles their MEGA method. However, both their schemes were analyzed under the restricted contexts of compressing data at only the node that generates the data or at only relaying nodes, which were called self and foreign coding in [3]. Moreover, it was assumed that all nodes produce data, so that the concept of a shallow light tree [16] was directly applicable.

Preliminary results of this paper appeared in [17], [18].

C. Main contributions and paper organization

In this paper, we build a data rate model, which can be extended to more complicated and practical situations, for source coding with explicit side information. Based on this model, an optimization problem for joint routing and source coding is constructed, and shown to be NP-hard. This problem generalizes the formulations in [2], [3] in the following aspects. First, not all sensors are required to generate data. Nodes can act as pure relays or simply do not participate in the routing or data processing activities. Second, in our data rate model, data compression can take place at any node as long as both the data to be compressed and the corresponding side information are present at the node. Lastly, the routing structure is not restricted to trees or any specific topologies. The only requirement is that all sensing data are delivered to the fusion center and no cyclic data dependency occurs during conditional compression. Due to these relaxations, the optimization problems in [2], [3] appear as subproblems of our formulation, and consequently, their solutions are bounded by the optimal solutions here.

Since finding the optimal solution in polynomial time is unlikely, we propose two simple routing schemes, BAS and DSIT. The worst-case performance ratios of both methods to the optimal solution of our general optimization problem are bounded for a special class of rate models. Simulation results also show that these methods have appealing average performances.

The rest of the paper is organized as follows. In Section II, we present our network flow and data rate models, based on which an optimization problem is formulated, and shown to be NP-hard in Section III. The BAS and DSIT methods are presented in Section IV. Their worst-case performance bounds are derived for a special class of rate models in Section V. The average performances of the two algorithms are studied and compared to other schemes through simulations in Section VI. Section VII concludes the paper.

II. Network Models

A. Network flows

The topology of a sensor network is abstracted as a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. $\mathcal{N}$ consists of a set $\mathcal{N}_s$ of n sensors and a special node t representing the fusion center. Denote by $\mathcal{N}_a$ the set of active sensors that generate data. $\mathcal{N}_a \subseteq \mathcal{N}_s$. Both active and non-active sensors can relay and compress data. $\mathcal{E}$ consists of m communication links with infinite capacity. We assume that all links are bi-directional and symmetric. If this is not the case, the network can be modeled as a directed graph, and most derivations in this paper will apply similarly. The network is connected, so messages from any sensor can reach t through direct transmission or relaying.

The network flow f_e is defined as the rate at which data is transmitted across $e \in \mathcal{E}$. A data flow generated by node i and terminating at node j is denoted by f^{ij}. In particular, we define $f^i = f^{it}$. Clearly, $f_e = \sum_{i,j \in \mathcal{N}} f_e^{ij}$, in which the subscript and superscript of f_e^{ij} take their meanings from above definitions respectively. A non-negative weight c_e is defined on each edge $e \in \mathcal{E}$ to indicate the cost (e.g. power) of transmitting data at unit rate across e. For $i, j \in \mathcal{N}$, the distance from i to j along path $\mathcal{P}$, denoted by $d_{ij}^{\mathcal{P}}$, is defined as the sum of the weights of edges from i to j on $\mathcal{P}$. If multiple routes exist between i and j (e.g. $\mathcal{P}$ contains a loop), $d_{ij}^{\mathcal{P}}$ assumes the smallest sum. The minimum distance from i to j, denoted by d_{ij}, is the distance along the shortest path (with metric c_e) from i to j. As a shortcut, we write $d_i^{\mathcal{P}} = d_{it}^{\mathcal{P}}$ and $d_i = d_{it}$. The objective is to minimize the communication cost of routing all the data to the fusion center.

$$C = \sum_{e \in \mathcal{E}} c_e f_e. \tag{1}$$

In practice, the cost can be a more complicated function of f_e, and (1) serves as a first order approximation.

B. Source coding with explicit side information

Denote by X_i the data stream produced by $i \in \mathcal{N}_s$. Assume X_i satisfies the ergodic condition so that the results of statistical probability theory can be applied. 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 the decoder can it be used to compress the data. We decompose transmissions to t into individual data flows f^i and consider the minimum rate required such that X_i can be recovered at the fusion center. Under lossy coding, suppose the side information for compressing X_i is $\hat{X}_{k_1}, \cdots, \hat{X}_{k_j}$, where $k_1, \cdots, k_j \in \mathcal{H}_i$. $\mathcal{H}_i$ is defined as the set of sensors whose data are correlated with X_i, and $\hat{X}_k$ denotes the coded version of X_k. Subject to the distortion constraint $d(X_i, \hat{X}_i) \leq D$, the minimum rate is given by the conditional mutual information

$$f^i = \min_{d(X_i, \hat{X}_i) \leq D} I(X_i, \hat{X}_i | \hat{X}_{k_1}, \cdots, \hat{X}_{k_j}). \tag{2}$$

When $X_j, \forall j \in \mathcal{N}_s$ takes values from a discrete alphabet, entropy coding can be used, so that

$$f^i = H(X_i | X_{k_1}, \cdots, X_{k_j}). \tag{3}$$

In either case, the data rate depends on what type of side information is available, and hence is a function of $M_i = |\mathcal{H}_i|$ binary variables, where $|\cdot|$ denotes the cardinality of the set. For a network of n sensors, M_i can be as large as $(n-1)$. Thus, this description alone has exponential complexity. To simplify the problem, in this paper, we assume that M_i is relatively small and side information from at most one sensor is used. Hence,

$$f^i = \begin{cases} b_i & \text{no side information;} \\ b_{i|j} & X_j, j \in \mathcal{H}_i, \text{ is used as side information.} \end{cases} \tag{4}$$

Note that $b_i \geq b_{i|j} \geq 0$ and $b_j \geq b_i - b_{i|j}$. Also, we have $b_i = 0$ if $i \in \mathcal{N}_s \setminus \mathcal{N}_a$. When X_i and X_j are correlated, $i \in \mathcal{H}_j$ and $j \in \mathcal{H}_i$.

The data rate model in (4) appears rather simple. However, there are practical reasons to assume such small values for M_i and the number of helpers. First, in many physical situations, high correlation occurs only in a small neighborhood. In others, reconstruction fidelity constraints may permit thinning the number of active sensors, so again only a small number of sensors has high correlation. Second, due to the correlation among side information streams, coding gain often saturates quickly as the number of helpers increases. In addition, determining coding gain information and processing side information incurs cost, and the gain of using additional helpers is often not enough to be worth it. Third, using more helpers increases the complexity of the model and optimization. On the other hand, our study on this simpler case can serve as a first step in dealing with more complicated problems where multiple helpers are allowed.

In this paper, we assume that (4) is given a priori. However, in practice, to obtain such rate functions, a training process must take place before the route design. In most situations, sensors in $\mathcal{H}_i$ are in the neighborhood of sensor i. Thus, each active sensor needs to transmit a small set of data to nearby sensors. Alternatively, such information can be fed back from sensors or the fusion center that performs the data aggregation. If neither is available, then simple indicators such as the attribute-value pairs [5] may be used to provide information on the correlation level of data collected at different sensors. Although the subject is not pursued here, we recognize that the cost and effectiveness of these various means of obtaining data correlation information remains an essential topic in data-centric routing.

C. Discussion

Data stream X_i can be compressed at i or any relaying nodes en route to the fusion center as long as the side information is present. Moreover, flows need not be bundled in transmission. For instance, in Fig. 2, flows f_{21}^2 and f_{31}^3 join one another at node 1, then split to take different paths f_{15}^2 and f_{14}^3 in ensuing transmissions. As a result, the overall routing structure is not necessarily a tree, and we point out that in data-centric routing, trees are not necessarily optimal. (Refer to Fig. 1 for one such example.)

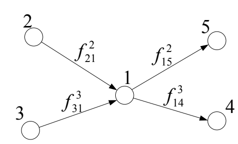


Fig. 2. Data flows split in the network. Subscript ij indicates the edge (i, j) from node i to node j.

When applying conditional compression, we must avoid cyclic data dependency. In other words, if X_j's recovery relies on X_i, then X_j cannot be used as side information for coding X_i. To formalize the idea, define a directed network $\mathcal{G}_h$ that consists of all active sensors. If X_i is used as side information for coding X_j, a directed edge (i, j) is formed from sensor i to sensor j. Then we have the following proposition:

Proposition 1: No cyclic data dependency occurs during conditional compression iff $\mathcal{G}_h$ contains no directed cycles.

The proof is straightforward, and hence omitted. We state the proposition here because it will become handy in Section IV.

We adopt an example from [11] to solidify our data rate model. In Fig. 3, a near-field sensor array records the sound of a tank as it moves by. This array is part of a wireless sensor network, and their data are to be transmitted to a fusion center (not shown in the figure). Since we are only concerned about

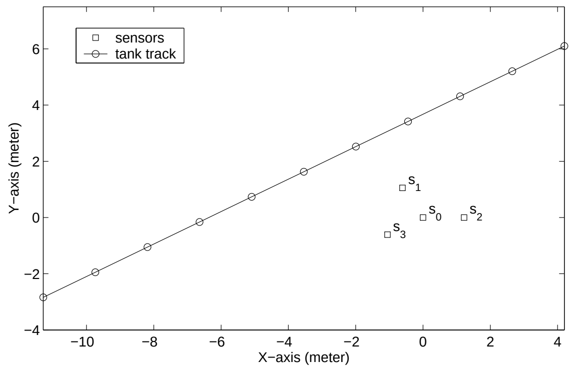


Fig. 3. Near field sensor array configuration

TABLE I
DATA RATE WITH DIFFERENT SIDE INFORMATION.

Helped by	none	s_0	s_1	s_2	s_0, s_1	s_0, s_2	s_1, s_2
σ_3^2	5.33	.443	.973	3.43	.424	.431	.943
f^3(bits)	6.19	4.40	4.96	5.87	4.36	4.38	4.94

the data correlation here, only four acoustic sensors that record the sound are shown in Fig. 3. The variance of the observations at s_3 is listed in Table I when they are compressed alone or with side information from other sensors using an adaptive DPCM encoder. The data rate is estimated by $0.5\log_2(\sigma_3^2/D)$, where D is set to 0.001. Notice that the coding gain provided by different sensors varies significantly with sensor locations. In practice, the cost of processing side information may lead to $\mathcal{H}_3 = \{s_0, s_1\}$ and using at most one helper.

III. COMBINED ROUTING AND SOURCE CODING

A. Problem formulation

Given the network model in Section II, the problem becomes how to construct transmission routes such that the total communication cost is minimized. We formulate this as an optimization problem, which is stated as follows.

Combined Routing and Source Coding (CRSC)
GIVEN: A graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ with a weight $c_e \geq 0$ defined on each $e \in \mathcal{E}$, a node $t \in \mathcal{N}$, a set $\mathcal{H}_i \subseteq \mathcal{N} \setminus \{t, i\}$, which satisfies $i \in \mathcal{H}_j$ iff $j \in \mathcal{H}_i$, and the rate f^i, at which data must be transmitted from i to t, defined for each $i \in \mathcal{N} \setminus \{t\}$.

$$f^i = \begin{cases} b_i & \text{without side information;} \\ b_{i|j} & \text{with side information from } j \in \mathcal{H}_i. \end{cases} \tag{5}$$

Here, $b_i \geq b_{i|j} \geq 0$ and $b_j \geq b_i - b_{i|j}$.
FIND: A set of routes for supplying explicit side information and transmitting data from $i \in \mathcal{N} \setminus \{t\}$ to t at required rate such that the total communication cost,

$$C = \sum_{e \in \mathcal{E}} c_e f_e = \sum_{e \in \mathcal{E}} \sum_{i,j \in \mathcal{N}} c_e f_e^{ij}, \tag{6}$$

is minimized and no cyclic data dependency occurs.

In our problem formulation, the total cost C includes not only the cost of transmitting data to t but the cost of any designated communication that supplies side information. Therefore, the destination j in (6) can be any node in $\mathcal{N}$.

B. NP-hardness

We prove the following theorem in this section.

Theorem 1: Solving CRSC is NP-hard.

Proof: To prove CRSC's NP-hardness, it is sufficient to show that one special case of CRSC is NP-hard [19]. We consider the following instance, and demonstrate that it is the minimum Steiner tree problem. Assume $|\mathcal{N}_a| < |\mathcal{N}_s|$ and $\mathcal{H}_i = \mathcal{N}_a \setminus i$ for $i \in \mathcal{N}_a$, where $\mathcal{N}_s = \mathcal{N} \setminus \{t\}$ and $\mathcal{N}_a$ is the set of nodes with nonzero data rates. Define

$$f^i = \begin{cases} 1 & i \in \mathcal{N}_a \text{ without side information;} \\ 0 & i \in \mathcal{N}_a \text{ with side information, or } i \in \mathcal{N}_s \setminus \mathcal{N}_a. \end{cases}$$

We first show that the cost of transmitting all the data to the fusion center can be minimized by using a tree route. Since all edges have unlimited capacity, there is no benefit in splitting an individual flow. (A path with the smallest distance among all splitting routes can be selected to carry the complete flow without increasing the cost.) Furthermore, since the rate function is either 1 or 0, the flow on any particular edge f_e must be an integer. Hence, the optimal route consists of a set of edges that carry flows $f_e > 1$, t, all active sensors, and relaying nodes. Suppose this optimal route does not form a tree. Since there is at least one path from each $i \in \mathcal{N}_a$ to t in the optimal route (otherwise, X_i cannot be recovered at t), all active nodes and t are connected, and we can find at least one tree that is embedded in the optimal route and connects all the active sensors to t. If there is more than one such tree, we pick the one with the minimal total of edge weights and use this tree as the transmission route. Given our data rate model, it is evident that the data flow f_e will always be 1 on any edge e of our tree route. Thus, the total cost is simply the sum of the edge weights of the tree, which is less than that of the optimal route. This contradiction shows that the set of optimal routes must form a tree.

As f_e is 1 on all edges of the routing tree, finding the optimal route is equivalent to constructing the minimum Steiner tree that connects all active sensors to t, which is an NP-hard problem. Hence, CRSC is also NP-hard. ■

C. SPT and Clusters

Since finding the exact solution of CRSC in polynomial time is unlikely, we turn to heuristics. Here, we present two routing methods that our proposed schemes, BAS and DSIT, will be compared to in Section VI.

A shortest path tree (SPT) is used to route data to t, and data compression is performed whenever explicit side information is available due to the mergence of flows in the network. We establish a result regarding SPT's worst-case performance.

Proposition 2: When the rate function is given by

$$f^i = \begin{cases} b_0 & \text{without side information;} \\ \beta b_0 & \text{with side information,} \end{cases} \tag{7}$$

where $i \in \mathcal{N}_a$ and $0 \leq \beta \leq 1$. The costs of the SPT (C_{SPT}) and optimal solution (C_{OPT}) satisfy the following relation.

$$C_{\text{OPT}}/C_{\text{SPT}} \geq \beta + (1-\beta)/n_a, \quad \text{where } n_a = |\mathcal{N}_a|. \tag{8}$$

This bound is tight in that, for any β and n_a, there are cases in which the equality in (8) holds.

Proof: Denote by $\mathcal{E}_{\text{ST}}$ the set of edges in the Steiner tree (ST) that connects all active sensors to t.

$$C_{\text{SPT}} \leq b_0 \sum_{i \in \mathcal{N}_a} d_i \leq b_0 n_a \sum_{e \in \mathcal{E}_{\text{ST}}} c_e, \quad (9)$$

where d_i is the minimum distance from i to t as defined in Section II-A. The first inequality is straightforward, and the second is due to the following relation.

$$\max_{i \in \mathcal{N}_a} d_i \leq \sum_{e \in \mathcal{E}_{\text{ST}}} c_e.$$

On the other hand, we have the following for an optimal route.

$$C_{\text{OPT}} \geq \beta b_0 \sum_{i \in \mathcal{N}_a} d_i + (1-\beta) b_0 \sum_{e \in \mathcal{E}_{\text{ST}}} c_e. \quad (10)$$

The first term is the minimum cost of routing data from all active sensors to the fusion center assuming side information is available at every node. The second term represents the extra cost of routing at least one complete set of data to t (to avoid cyclic data dependency) and supplying side information to active sensors, which is minimized by using a Steiner tree. In the most fortunate situations, only the fraction $(1-\beta)b_0$ of data need to be transmitted (such an example is given in Fig. 4 when $\epsilon \to 0$). (9) and (10) lead to (8).

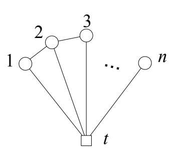


Fig. 4. An instance that achieves the equality in (8): $\mathcal{N}_a = \{1, \cdots, n\}$; $\mathcal{H}_i = \mathcal{N}_a \setminus i$, $c_{it} = 1, i \in \mathcal{N}_a$; $c_{k,k+1} = \epsilon$, $1 \leq k \leq (n-1)$.

To show that the bound is tight, consider the instance depicted in Fig. 4. The equality in (8) is asymptotically attained when $\epsilon \to 0$. ■

It is no surprise that the worst-case performance ratio depends strongly on β, which indicates the level of the coding gain. After all, the bound represents the penalty that a SPT receives for ignoring the data correlation in designing routes.

Various clustering methods have been proposed [14], [15] for sensor networks. In this paper, we use a clustering method based on geographical proximity. The sensing field is divided into rectangular cells, and sensors that fall in the same cell are grouped into a cluster. The sensor with the smallest distance to the fusion center is picked as the cluster head. In each group, sensors transmit to the cluster head to have their data fused, and subsequently sent to the fusion center.

IV. Two Heuristic Algorithms

A. Balanced Aggregation Scheme

In this section, we propose an approximation algorithm that is inspired by the idea of balancing shortest path trees and trees with small weights [6]. In this algorithm, all transmissions terminate at t. Data are aggregated on their way to the fusion center. Hence, the total cost of routing can be expressed as

$$C = \sum_{e \in \mathcal{E}} \sum_{i \in \mathcal{N}_a} c_e f_e^i. \quad (11)$$

1) Motivation: To motivate the algorithm, we assume $b_{i|j} = b_{i|\cdot}$ in (5), i.e. $b_{i|j}$ is the same for all $j \in \mathcal{H}_i$. We decompose f_e^i into two parts.

$$f_e^i = q_e^i + r_e^i.$$

When $f_e^i = 0$, $q_e^i = r_e^i = 0$. Otherwise, we define

$$q_e^i = b_{i|\cdot}.$$

$$r_e^i = \begin{cases} b_i - b_{i|\cdot} & \text{no side information;} \\ 0 & \text{with side information.} \end{cases}$$

Thus, q_e^i is the portion that is independent of side information, and r_e^i represents the part that is compressible by the helper's data. Accordingly, (11) can be decomposed into two parts.

$$C = C_q + C_r \triangleq \sum_{e \in \mathcal{E}} \sum_{i \in \mathcal{N}_a} c_e q_e^i + \sum_{e \in \mathcal{E}} \sum_{i \in \mathcal{N}_a} c_e r_e^i.$$

Consider, for a moment, minimizing C_q and C_r separately. C_q is minimized when the route is a set of shortest paths. On the other hand, to obtain a small C_r, we should try to jointly code X_i and $X_j, j \in \mathcal{H}_i$, and merge flows using routes of small weights. Accordingly, when coding gain is small ($b_{i|\cdot} \gg b_i - b_{i|\cdot}$), the route is expected to be close to a SPT. When there is substantial coding gain ($b_{i|\cdot} \ll b_i - b_{i|\cdot}$), the focus is on achieving aggregation with minimal 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, and the balance is struck based on the relative magnitudes of b_i and $b_{i|\cdot}$.

2) 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. 5, there is a path connecting the active sensor k to t. Define $\mathcal{P}_k = \{k, (k, v_1), v_1, \cdots, v_p, (v_p, t), t\}$ the sequence of nodes and edges on the path, and $d_{uv}^{\mathcal{P}_k}$ $(u, v \in \mathcal{P}_k)$ the distance from u to v along $\mathcal{P}_k$ as in Section II-A. Set $d_u^{\mathcal{P}_k} = d_{ut}^{\mathcal{P}_k}$. We want to find a path to route the data of i to t such that the resulting cost is minimized. This amounts to determining an

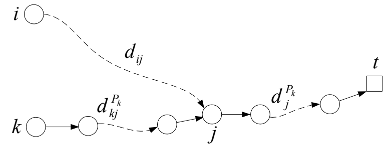


Fig. 5. Construct a balanced path for i.

aggregation node $j \in \mathcal{P}_k$ where f^k and f^i join one another. Note that once aggregation takes place, we require flows to be bundled in subsequent transmission such that compressed data can be recovered properly at all relays to be potential side information for other flows. There are two possibilities.

(1) X_i and X_k are uncorrelated. The optimal route is the shortest path from i to t, and the cost is:

$$C_{i|k} = b_i d_i.$$

(2) X_i and X_k are correlated. Assuming the two flows merge at node $j \in \mathcal{P}_k$, the cost of routing f^i to t is

$$C^j_{i|k} = b_i d_{ij} + b_{i|k} d_j^{\mathcal{P}_k}.$$

We choose $\arg\{\min_{j\in\mathcal{P}_k} C^j_{i|k}\}$ as the aggregation node and set $C_{i|k} = \min_{j\in\mathcal{P}_k} C^j_{i|k}$.

Balanced paths do not necessarily constitute a tree. For example, in Fig. 5, the path from i to j may have used nodes in $\mathcal{P}_k$ as relays. The aggregation does not take place at the first node where f^i encounters $\mathcal{P}_k$ because $\mathcal{P}_k$ is not necessarily a shortest path, and it may be more advantageous to delay the mergence of flows until j. One such scenario is depicted in Fig. 6, where f^k runs through $\{k, a, b, j, t\}$ to aggregate with f^l at b. The route for f^i consists of $\{i, a, j, t\}$, which bypasses b and merges with f^k at j. It is easy to come up with rate functions and edge weights to result in such a scenario.

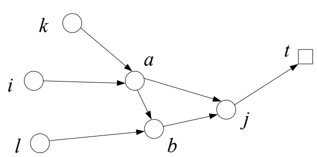


Fig. 6. A balanced path that is not a tree.

3) Balanced aggregation scheme: The complete algorithm involves successive steps of adding balanced paths of active sensors to the solution route. Each time, the newly routed sensor incurs the smallest routing cost among all the remaining sensors. We state our algorithm as follows.

Balanced Aggregation Scheme (BAS)

Given a graph $\mathcal{G}$ with edge weights and data rate models properly defined, set $\mathcal{U} = \mathcal{N}_a$ and $C = 0$. Carry out the following steps.

(1) Find the shortest path from each active sensor to t. Denote by I the sensor that has the minimum routing cost $C_i = b_i d_i$, $i \in \mathcal{N}_a$.

(2) $C = C + C_I$. Remove I from $\mathcal{U}$, and add I's path to the solution route. If $\mathcal{U}$ is empty, stop the algorithm.

(3) For each sensor $i \in \mathcal{U}$, compute $C_{i|k}$ resulting from merging f^i with f^k, $k \in \mathcal{N}_a \setminus \mathcal{U}$. Determine

$$C_i = \min_{k\in\mathcal{N}_a\setminus\mathcal{U}} C_{i|k}.$$

(4) Find $I = \arg\{\min_{i\in\mathcal{U}} C_i\}$. Return to step (2).

Since exactly one active sensor is removed from $\mathcal{U}$ during each iteration, we can number sensors according to the order that they are removed. If we construct the directed graph $\mathcal{G}_h$ in Proposition 1 for the route built from BAS, a directed edge (i, j) is possible only when sensor i has been marked with a smaller number than j. It is easy to show that such a $\mathcal{G}_h$ contains no directed loops. Thus, no cyclic data dependency results from BAS.

B. Designated Side Information Transmission

1) Motivation: In this section, we take up the approach in Fig. 1 (d). Data flows from active sensors to t are routed independently and do not aggregate with one another. Side information transmissions are carried out on designated routes. These transmissions are uncoded and need not to be relayed to the fusion center. The total routing cost can be decomposed into the cost of routing explicit side information C_s and the cost of transmitting data to the fusion center C_t:

$$C = C_s + C_t \triangleq \sum_{i,j\in\mathcal{N}_a}\sum_{e\in\mathcal{E}} c_e f_e^{ij} + \sum_{i\in\mathcal{N}_a}\sum_{e\in\mathcal{E}} c_e f_e^i. \qquad (12)$$

Note that the destination of f_e^i in C_t is t, while f_e^{ij} in C_s always terminates at an active sensor ($j \in \mathcal{N}_a$) since it is for side information transmission only.

In addition to achieving good performance when there is high data correlation in the network, DSIT offers greater flexibilities than strategies that bundle flows terminating at the fusion center and flows carrying side information. Here, f^i can be routed to t through any path. As a result, traditional address-centric routing schemes that are designed to evenly distribute the traffic load in the network and maximize node lifetime [20], [21] can be applied.

2) Maximum weight branching: We first consider constructing routes for transmissions to the fusion center. These routes affect only C_t. In addition, as $f^i, i \in \mathcal{N}_a$ does not provide any side information, its routing is decoupled from the data aggregation process. Hence, shortest paths should be used to achieve the minimum C_t.

$$C_t = \sum_{i\in\mathcal{N}_a} d_i f^i, \qquad (13)$$

where f^i is a function of the side information transmission.

Designing routes for side information transmission is more complicated. First, it is NP-hard. Consider an instance where a node needs to transmit side information to multiple active sensors. This is a multi-cast problem with Steiner trees as its optimal solution. Second, we need to ensure that no cyclic data dependency results while routing the side information. This amounts to avoiding directed cycles in G_h according to Proposition 1.

For the moment, we ignore the Steiner tree part, and use shortest paths to route all the side information. This leads to constructing a network $\mathcal{G}_a$ as follows. $\mathcal{G}_a$ consists of the set of active sensors $\mathcal{N}_a$. In addition, for each ordered pair of nodes $i, j \in \mathcal{N}_a$, create a directed edge (i, j) from sensor i to j and assign the weight w_{ij} to represent the net gain of routing side information from i to j. We have

$$w_{ij} = \begin{cases} (b_j - b_{j|i})d_j - d_{ij}b_i & \text{if } i \in \mathcal{H}_j; \\ -d_{ij}b_i & \text{otherwise.} \end{cases} \qquad (14)$$

Denote by $\mathcal{A}_a$ the set of directed edges with $w_{ij} > 0$, and define $\mathcal{G}_a = (\mathcal{N}_a, \mathcal{A}_a)$. A branching on the directed graph $\mathcal{G}_a$ is a set of directed edges $\mathcal{B} \subseteq \mathcal{A}_a$ satisfying the conditions that no two edges in $\mathcal{B}$ enter the same node, and $\mathcal{B}$ has no directed cycle. It is evident that a branching on $\mathcal{G}_a$ represents a feasible set of routes for side information transmission. No two directed edges in $\mathcal{B}$ pointing to the same node ensures that a sensor uses side information from at most one helper, and no directed cycle avoids the cyclic data dependency. The problem of minimizing the total cost is equivalent to maximizing the weight sum of the branching $\mathcal{B}$, which is stated as follows.

Maximum Weight Branching (MWB)
GIVEN: A directed graph $\mathcal{G}_a = (\mathcal{N}_a, \mathcal{A}_a)$ with a weight w_e defined on each directed edge $e \in \mathcal{A}_a$.
FIND: A branching $\mathcal{B} \subseteq \mathcal{A}_a$ that maximizes $\sum_{e\in\mathcal{B}} w_e$.

A polynomial time algorithm for solving this problem has been discovered independently by [22] and [23].

3) Designated side information transmission: Once the optimal branching $\mathcal{B}$ is determined, we revert to using Steiner trees. Define $\mathcal{S}_i$ as the set of sensors that receive side information from $i \in \mathcal{N}_a$ based on the optimal branching $\mathcal{B}$. We use the shortest path heuristic [24] to construct the subtree that connects i and $\mathcal{S}_i$. Our heuristic algorithm is a combination of the maximum weight branching and the Steiner tree approximation. We state it as follows:

Designated Side Information Transmission (DSIT Heuristic)
Given a network $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ with edge weights and rate functions properly defined, carry out the following steps.

(1) Find the shortest path from each active sensor to the fusion center. These are the routes for transmitting data to the fusion center.
(2) Construct a directed graph $\mathcal{G}_a = (\mathcal{N}_a, \mathcal{A}_a)$, where $\mathcal{N}_a$ are the set of active sensors and $\mathcal{A}_a$ is the set of directed edges from i to j ($i, j \in \mathcal{N}_a$ and $i \neq j$) whose edge weights w_{ij} defined as in (14) are greater than zero.
(3) Find the maximum weight branching $\mathcal{B}$ on $\mathcal{G}_a$. Based on $\mathcal{B}$, determine the set of sensors $\mathcal{S}_i$ that each active sensor $i \in \mathcal{N}_a$ transmits side information to.
(4) Run a shortest path heuristic for the Steiner tree problem to find the subtree for transmitting side information from $i \in \mathcal{N}_a$ to the sensors in $\mathcal{S}_i$.

It is easy to see the similarity between DSIT and the MEGA method in [3]. Instead of transmitting side information, MEGA transmits data to other nodes to seek aggregation. The benefit of this approach is that data can be relayed toward the fusion center while being compressed. On the other hand, DSIT offers the opportunity of multicasting side information to multiple receivers, and thus reducing the cost. In [3], the MEGA method is evaluated in the context of foreign coding and shown to be optimal. In this paper, the performance of DSIT will be bounded against the optimal solution of CRSC.

V. Performance Analysis

The performance analysis for arbitrary data correlation is very difficult. Instead, most of our derivation in this section will be based on a special class of data rate model: $\mathcal{H}_i = \mathcal{N}_a \backslash i$, $i \in \mathcal{N}_a$, and the rate function is given by (7).

A. Balanced Aggregation Scheme

BAS completes in n_a iterations. During each round, there is no need to compute C_i, $i \in \mathcal{U}$ in step (3) all over. It suffices to update C_i such that the newly added path in the previous iteration is accounted for. The bottleneck of BAS is on constructing shortest paths to each active sensor. Using Dijkstra's algorithm, it runs in $O(n_a m \log n)$ time. The distributed implementation of BAS relies on efficient parallel algorithms for constructing shortest paths between pairs of sensors, which is a well-researched topic. We refer readers to, for example, [25] for discussions of the subject.

For a set of sensors with uncorrelated data, BAS builds the shortest path from each active sensor to the fusion center. For the special instance that results in the minimum Steiner tree problem in Section III-B, BAS collapses to the shortest path heuristic [24], which has a worst-case performance ratio of $C_{\text{BAS}}/C_{\text{OPT}} = 2n_a/(n_a + 1)$. For more general cases, we begin our analysis by proving the following lemma.

Lemma 1: Denote by d_i^{BAS} the distance from i to t along the BAS path constructed for $i \in \mathcal{N}_a$. Given $\mathcal{H}_i = \mathcal{N}_a \setminus i$, $i \in \mathcal{N}_a$ and the rate function in (7), we have

$$\sum_{i\in\mathcal{N}_a} d_i^{\text{BAS}} \leq \kappa_1 \sum_{i\in\mathcal{N}_a} d_i, \tag{15}$$

where

$$\kappa_1 = \frac{1}{\beta} - \frac{(1-\beta)[1-(1-\beta)^{n_a}]}{n_a \beta^2}.$$

This bound is tight in that, for any β and n_a, there are network instances that attain the equality in (15).

Proof: We label all the nodes in $\mathcal{N}_a$ according to the order that they are added to the BAS route. In other words, node i is the ith active node connected to t. Denote by J_i the aggregation node for flow f^i, and K_i the corresponding helper. When no aggregation takes place, we set $J_i = K_i = t$. To obtain an upper bound for $\sum_{i\in\mathcal{N}_a} d_i^{\text{BAS}}$, we try to maximize d_i^{BAS} for $i = 1, 2, \cdots, n_a$ in successive steps.

For $i = 1$,

$$d_1^{\text{BAS}} = d_1 = \min_{i\in\mathcal{N}_a} d_i. \tag{16}$$

For $i = 2$, since the BAS path constructed for node 2 is the one that incurs minimum cost among all $\{j \mid 2 \leq j \leq n_a\}$,

$$b_0 d_{2J_2} + \beta b_0 d_{J_2}^{\mathcal{P}_{K_2}} \leq b_0 \min_{2\leq j\leq n_a} d_j.$$

Otherwise, the node with the smallest distance to t will have lower routing cost than node 2. On the other hand,

$$d_{J_2}^{\mathcal{P}_{K_2}} \leq d_1^{\text{BAS}} = \min_{i\in\mathcal{N}_a} d_i.$$

Therefore,

$$d_2^{\text{BAS}} = d_{2J_2} + d_{J_2}^{\mathcal{P}_{K_2}} \leq \theta_2 + (1-\beta)\theta_1, \tag{17}$$

where we have defined $\theta_i = \min_{i\leq j\leq n_a} d_j$. Continuing in the same fashion, we can show that, at iteration i,

$$d_{J_i}^{\mathcal{P}_{K_i}} \leq \max_{1\leq j\leq i-1} d_j^{\text{BAS}} \leq \sum_{j=1}^{i-1} (1-\beta)^{i-1-j}\theta_j, \tag{18}$$

and

$$\begin{aligned} d_i^{\text{BAS}} &= d_{iJ_i} + d_{J_i}^{\mathcal{P}_{K_i}} \leq \theta_i + (1-\beta) d_{J_i}^{\mathcal{P}_{K_i}} \\ &\leq \sum_{j=1}^{i} (1-\beta)^{i-j}\theta_j. \end{aligned} \tag{19}$$

Define

$$a_j = \sum_{k=0}^{n_a-j} (1-\beta)^k = [1-(1-\beta)^{n_a+1-j}]/\beta.$$

Sum (19) over all $1 \le i \le n_a$.

$$\begin{aligned}\sum_{i=1}^{n_a} d_i^{\text{BAS}} &\le \sum_{i=1}^{n_a}\sum_{j=1}^{i}(1-\beta)^{i-j}\theta_j \overset{(a)}{=} \sum_{j=1}^{n_a} a_j\theta_j \\ &\overset{(b)}{\le} \left(\sum_{j=1}^{n_a}\theta_j\right)\left(\sum_{j=1}^{n_a} a_j\right)\Big/ n_a \\ &\overset{(c)}{\le} \left(\sum_{j=1}^{n_a} d_j\right)\left(\sum_{j=1}^{n_a} a_j\right)\Big/ n_a.\end{aligned}$$

Step (a) is obtained by defining $k = i - j$ to replace i. Step (b) uses Proposition 3 proved in the Appendix. Step (c) comes from the fact that $\theta_j \le d_j$. From this, (15) is easily derived.

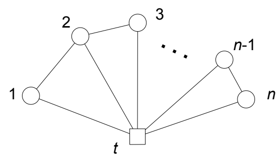


Fig. 7. An instance that attains the equality in (15): $\mathcal{N}_a = \{1, 2, \cdots, n\}$; $c_{1t} = 1$; $c_{it} = 1+\epsilon$, $c_{i-1,i} = (1-\beta)^{i-1}$, $2 \le i \le n$.

The network instance depicted in Fig. 7 achieves the equality in (15) asymptotically when $\epsilon \to 0$. ■

We proceed to prove the following lemma.

Lemma 2: Denote by J_i the aggregation node for flow f^i. ($J_i = t$ when there is no helper.) Given $\mathcal{H}_i = \mathcal{N}_a \setminus i$, $i \in \mathcal{N}_a$ and the rate function in (7), we have

$$\sum_{i\in\mathcal{N}_a} d_{iJ_i} \le \kappa_2 \sum_{e\in\mathcal{E}_{\text{ST}}} c_e + \beta\kappa_3 \sum_{i\in\mathcal{N}_a} d_i, \tag{20}$$

where

$$\kappa_2 = \frac{2n_a}{n_a+1}, \quad \kappa_3 = \frac{1}{\beta} - \frac{1-(1-\beta)^{n_a}}{n_a\beta^2}.$$

Proof: We label all the active nodes the same way as we did in the proof of Lemma 1, and label t as node 0. Assume that we are at iteration i of the BAS algorithm, and node i is picked to be connected to t via aggregation node J_i, helped by node K_i. The cost of this step is

$$C_i = b_0 d_{iJ_i} + \beta b_0 d_{J_i}^{\mathcal{P}_{K_i}}, \tag{21}$$

which is the minimum routing cost among all un-routed active nodes, $\{j \mid i \le j \le n_a\}$. Suppose instead of node i, we choose another node among un-routed active nodes to be connected to t according to the following procedure. We temporarily set $\beta = 0$, and find the node $\hat{i}$ that yield the minimum routing cost $b_0 d_{\hat{i}\hat{J}_i}$ via aggregation node $\hat{J}_i$ helped by $\hat{K}_i$. Then we revert β to its original value, and observe that

$$b_0 d_{iJ_i} + \beta b_0 d_{J_i}^{\mathcal{P}_{K_i}} \le b_0 d_{\hat{i}\hat{J}_i} + \beta b_0 d_{\hat{J}_i}^{\mathcal{P}_{\hat{K}_i}}. \tag{22}$$

This comes from the fact that when β is at its original value, $\hat{J}_i$ is a suboptimal aggregation node for $\hat{i}$, and hence the resulting cost is no less than the minimum cost in (21). Moreover,

$$d_{\hat{i}\hat{J}_i} \le d_{pq}, \tag{23}$$

for any $0 \le p, q \le n_a$ that satisfies $\min(p, q) \le i-1$ and $\max(p, q) \ge i$. (Recall that t is labelled as node 0). Otherwise, we could have set $\hat{i} = \max(p, q)$ and $\hat{J}_i = \hat{K}_i = \min(p, q)$ to yield a cost lower than $b_0 d_{\hat{i}\hat{J}_i}$ when β was temporarily set to 0. Sum (22) over all $i \in \mathcal{N}_a$.

$$\begin{aligned}\sum_{i=1}^{n_a}\left[d_{iJ_i} + \beta b_0 d_{J_i}^{\mathcal{P}_{K_i}}\right] &\le \sum_{i=1}^{n_a} d_{\hat{i}\hat{J}_i} + \beta\sum_{i=1}^{n_a} d_{\hat{J}_i}^{\mathcal{P}_{\hat{K}_i}} \\ &\overset{(a)}{\le} \frac{2n_a}{n_a+1}\sum_{e\in\mathcal{E}_{\text{ST}}} c_e + \beta\sum_{i=1}^{n_a} d_{\hat{J}_i}^{\mathcal{P}_{\hat{K}_i}} \\ &\overset{(b)}{\le} \kappa_2\sum_{e\in\mathcal{E}_{\text{ST}}} c_e + \beta\kappa_3\sum_{i\in\mathcal{N}_a} d_i. \end{aligned} \tag{24}$$

Step (a) is justified because (23) is the same inequality satisfied by the shortest path heuristic of Steiner tree problem. (The full proof can be found in [24].) Step (b) comes from (18) and the same derivation in the proof of Lemma 1. ■

The equality in (20) can be achieved when $\beta = 0$, but the bound is not tight for the general case of $\beta > 0$. With Lemma 1 and 2 at hand, we are now ready to provide a bound on the performance of BAS.

Theorem 2: Given $\mathcal{H}_i = \mathcal{N}_a \setminus i$, $i \in \mathcal{N}_a$ and the rate function in (7), the routing costs resulting from CRSC's optimal solution (C_{OPT}) and BAS scheme (C_{BAS}) satisfy

$$\frac{C_{\text{OPT}}}{C_{\text{BAS}}} \ge \max\left[\beta + \frac{(1-\beta)}{\xi},\ \frac{\beta\xi + (1-\beta)}{\kappa_3\beta\xi + \kappa_2},\ \frac{\beta\xi + (1-\beta)}{(2\kappa_1 - 1)\beta\xi + \kappa_2(1-\beta)}\right]. \tag{25}$$

Here, we have defined

$$\xi = \sum_{i\in\mathcal{N}_a} d_i \Big/ \sum_{e\in\mathcal{E}_{\text{ST}}} c_e,$$

and κ_1, κ_2, and κ_3 are parameters defined in Lemma 1 and 2.

Proof: We observe that the performance bound given in Proposition 2 is attained by a SPT without in-network data aggregation. On the other hand, BAS is no worse than a SPT without in-network data aggregation. Therefore, the first term on the right hand side of (25) is a result of (9) and (10). The second term comes from (24) and (10).

In addition, from Lemma 1 and 2, we have

$$\begin{aligned} C_{\text{BAS}} &= \beta b_0 \sum_{i\in\mathcal{N}_a} d_i^{\text{BAS}} + (1-\beta) b_0 \sum_{i\in\mathcal{N}_a} d_{iJ_i} \\ &\le \left[\beta(2\kappa_1 - 1)\sum_{i\in\mathcal{N}_a} d_i + (1-\beta)\kappa_2\sum_{e\in\mathcal{E}_{\text{ST}}} c_e\right] b_0. \end{aligned}$$

The third term on the right hand side of (25) can be obtained from above inequality and (10). ■

Note that ξ is a parameter that depends on the specific network topology, and it satisfies $1 \le \xi \le n_a$. Similar to Lemma 2, (25) is generally not tight. An example of the bound is depicted in Fig. 8, where $C_{\text{OPT}}/C_{\text{BAS}}$ is confined in the shaded region. We can see that BAS prevents the worst-case performance ratio from continuously dropping as β decreases.

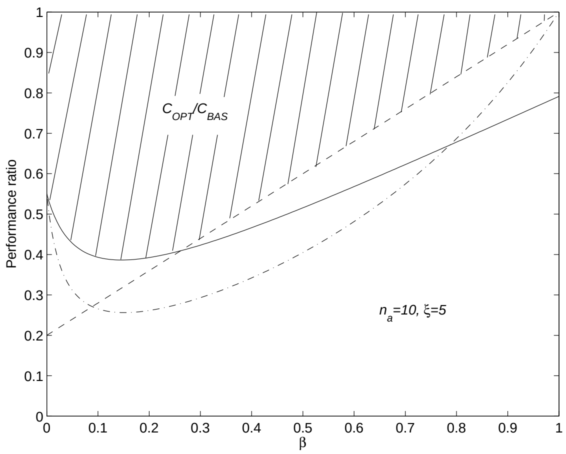


Fig. 8. The performance bound in Theorem 2 given $n_a = 10, \xi = 5$.

Although the rate model in Theorem 2 appears oversimplified, in practice, we may be able to approximately divide a sensor network into groups of active sensors, and within each group, sensors produce highly correlated data and have similar data rates. Hence, the bound in (25) can be applied to each group and a coarse bound on the overall network can be obtained by summing the results from all groups. However, we do not pursue this idea in this paper.

B. Designated side information transmission

Finding the maximum weight branching takes $O(m_a \log n_a)$ time, where $m_a = |\mathcal{A}_a|$. The shortest path heuristic for a Steiner tree requires $O(n_a m \log n)$ time for a sparse network. (The actual running time of the shortest path heuristic is in general much less because the number of nodes involved in constructing the shortest path is $|\mathcal{S}_i|$, which is often much smaller than n_a.) Hence, the computational cost of DSIT and BAS are of the same order. The distributed implementation of DSIT requires efficient parallel algorithms for constructing not only shortest paths but also optimal branchings. For the latter, we refer readers to [26], which discusses a distributed version of the Edmonds' method [23].

Regarding the performance of our heuristic algorithm comparing to that of the optimal solution attainable using a DSIT strategy, we prove the following theorem.

Theorem 3: Given the rate model in CRSC problem, the cost resulting from our DSIT heuristic (C_{DSIT}) and the minimum cost using the DSIT strategy (C_{MIN}) satisfy

$$C_{\text{DSIT}}/C_{\text{MIN}} \leq N, \tag{26}$$

where $N = \max\{1, \max_{i\in\mathcal{N}_a} |\mathcal{S}_i^{\text{opt}}|\}$, the greater of one and the maximum number of sensors that one sensor needs to transmit side information to in the optimal solution. The bound is tight in that there are instances that attain the equality in (26) for any value of N.

Proof: It is first noted that $\mathcal{S}_i^{\text{opt}}$ is in general not the same as the $\mathcal{S}_i$ in our heuristic algorithm, and C_{MIN} is the optimal result achievable by the DSIT approach, which may be greater than C_{OPT}, the minimum cost of the CRSC problem.

When $\max_{i\in\mathcal{N}_a} |\mathcal{S}_i^{\text{opt}}| = 0$, a SPT without in-network data aggregation is used. Hence, $C_{\text{DSIT}}/C_{\text{MIN}} = 1$.

Suppose $\max_{i\in\mathcal{N}_a} |\mathcal{S}_i^{\text{opt}}| \geq 1$. In the optimal DSIT solution, side information is routed from i to $\mathcal{S}_i^{\text{opt}}$ using the minimum Steiner tree, and data is transmitted from active sensors to t through shortest paths. Denote by $\mathcal{E}_{\text{ST}(i)}$ the set of edges of the Steiner tree for circulating side information supplied by sensor $i \in \mathcal{N}_a$. $\mathcal{E}_{\text{ST}(i)} = \emptyset$, if $\mathcal{S}_i^{\text{opt}} = \emptyset$. Hence,

$$C_{\text{MIN}} = \sum_{i\in\mathcal{N}_a} f^i d_i + \sum_{i\in\mathcal{N}_a} b_i \sum_{e\in\mathcal{E}_{\text{ST}(i)}} c_e.$$

Instead of the Steiner tree, consider relying on a shortest path tree to route the side information from i to the sensors in $\mathcal{S}_i^{\text{opt}}$. The corresponding cost C' will be

$$C' = \sum_{i\in\mathcal{N}_a} f^i d_i + \sum_{i\in\mathcal{N}_a} b_i \sum_{j\in\mathcal{S}_i^{\text{opt}}} d_{ij}.$$

Since

$$N_i \sum_{e\in\mathcal{E}_{\text{ST}(i)}} c_e \geq \sum_{j\in\mathcal{S}_i^{\text{opt}}} d_{ij}, \text{ where } N_i = |\mathcal{S}_i^{\text{opt}}|, \tag{27}$$

we have

$$C'/C_{\text{MIN}} \leq \max_{i\in\mathcal{N}_a} N_i.$$

On the other hand, C_{DSIT} is at least as good as the optimal result of using the shortest path tree to route the side information. Therefore, $C_{\text{DSIT}} \leq C'$. Together with above inequality, this gives rise to the bound in (26).

To show that the bound is tight, we consider the instance in Fig. 9. The edge weights between sensors v_k and u_k

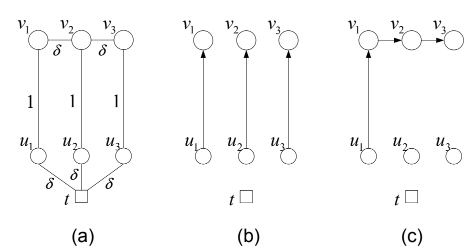


Fig. 9. An instance that attains the equality in (26): (a) network setup; (b) routes of side information transmission using DSIT heuristic; (c) routes of side information transmission in the optimal DSIT solution.

($k = 1, 2, 3$) is 1. Other edges have weight δ. All the sensors are active with data rate R without side information and 0 when side information is available. Denote $\mathcal{U} = \{u_1, u_2, u_3\}$, and $\mathcal{V} = \{v_1, v_2, v_3\}$. We assume $\mathcal{H}_i$ is $\mathcal{U}$ when $i \in \mathcal{V}$, and $\mathcal{V}$ when $i \in \mathcal{U}$. In Fig. 9, (b) and (c) illustrate how side information is transmitted in the DSIT heuristic and optimal DSIT solution. Accordingly, $C_{\text{DSIT}} = 3R + 3R\delta$ and $C_{\text{MIN}} = R + 5R\delta$. When $\delta \to 0$, the ratio $C_{\text{DSIT}}/C_{\text{MIN}}$ approaches $N = 3$ asymptotically. In a similar fashion, problem instances with arbitrary values of N can be devised. ■

It is easy to show that the worst-case scenario in Fig. 9 applies to the MEGA scheme in [3] as well. Hence, its performance will be at least bounded by (26). We now proceed to prove the following theorem, which bounds the ratio of C_{OPT} to C_{DSIT} for a special class of rate models.

Theorem 4: Given $\mathcal{H}_i = \mathcal{N}_a \backslash i, i \in \mathcal{N}_a$ and the rate function in (7), the routing costs of CRSC's optimal solution (C_{OPT}) and DSIT heuristic (C_{DSIT}) satisfy

$$\frac{C_{\text{OPT}}}{C_{\text{DSIT}}} \geq \max\left[\beta + \frac{(1-\beta)}{\xi},\ \frac{1+\beta(\xi-1)}{1+\beta(\xi-1)+2N/(N+1)}\right], \quad (28)$$

where ξ and N are defined as in Theorem 2 and 3 respectively.

Proof: As in Theorem 2, the first term on the right hand side of (28) comes from Proposition 2 since DSIT is no worse than a SPT without in-network data aggregation.

Define $\mathcal{O}$ to be the set of active sensors that receive no side information in the optimal DSIT solution. To avoid cyclic data dependence, we must have $|\mathcal{O}| \geq 1$. We construct a viable DSIT solution as follows. First, build the Steiner tree that connects all active sensors to t, and denote by $\mathcal{E}_{\text{ST}}$ the set of edges on this tree. Then, pick a node $I \in \mathcal{O}$. Through the Steiner tree constructed above, the data of I are distributed to all other active sensors to be used as side information. Shortest paths are used for transmissions from active sensors to t. The total cost of this DSIT solution (C'') satisfies

$$C'' \leq b_0 d_I + \beta b_0 \sum_{i\in\mathcal{N}_a\backslash\{I\}} d_i + b_0 \sum_{e\in\mathcal{E}_{\text{ST}}} c_e.$$

The inequality comes from the fact that side information only needs to be transmitted to active sensors while $\mathcal{E}_{\text{ST}}$ reaches t as well. Then, we have

$$\begin{aligned} C_{\text{MIN}} &= b_0 \sum_{i\in\mathcal{O}} d_i + \beta b_0 \sum_{i\in\mathcal{N}_a\backslash\mathcal{O}} d_i + b_0 \sum_{i\in\mathcal{N}_a}\sum_{e\in\mathcal{E}_{\text{ST}(i)}} c_e \\ &\overset{(a)}{\leq} b_0 d_I + \beta b_0 \sum_{i\in\mathcal{N}_a\backslash\{I\}} d_i + b_0 \sum_{e\in\mathcal{E}_{\text{ST}}} c_e \qquad (29) \\ &\overset{(b)}{\leq} b_0 \sum_{i\in\mathcal{O}} d_i + \beta b_0 \sum_{i\in\mathcal{N}_a\backslash\mathcal{O}} d_i + b_0 \sum_{e\in\mathcal{E}_{\text{ST}}} c_e, \end{aligned}$$

where $\mathcal{E}_{\text{ST}(i)}$ has the same meaning as in the proof of Theorem 3. Step (a) is because $C_{\text{MIN}} \leq C''$, and step (b) comes from $I \subseteq \mathcal{O}$ and $\beta \leq 1$. Therefore,

$$\sum_{i\in\mathcal{N}_a}\sum_{e\in\mathcal{E}_{\text{ST}(i)}} c_e \leq \sum_{e\in\mathcal{E}_{\text{ST}}} c_e \qquad (30)$$

As in the proof of Theorem 3, consider circulating side information within the group of nodes $\mathcal{V}_i = \{i\} \cup \mathcal{S}_i^{\text{opt}}$ using a shortest path tree. Since $\mathcal{H}_j = \mathcal{N}_a \setminus j$, $j \in \mathcal{N}_a$, a bound better than (27) can be obtained by allowing sensors other than i to transmit side information within the group. Let $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i)$ be a complete graph, and the cost c_{uv} of edge $(u,v) \in \mathcal{E}_i$ be the length of a shortest path between u and v, $(u, v \in \mathcal{V}_i)$. Denote by $\mathcal{E}_{\text{MST}(i)}$ the set of edges of the minimum spanning tree on $\mathcal{G}_i$. Then, it is known [27]

$$\sum_{e\in\mathcal{E}_{\text{MST}(i)}} c_e \leq \frac{2N_i}{N_i+1} \sum_{e\in\mathcal{E}_{\text{ST}(i)}} c_e. \qquad (31)$$

Now, we make $(\mathcal{V}_i, \mathcal{E}_{\text{MST}(i)})$ a directed tree by setting all its edges pointing away from i, and for any $(u,v) \in \mathcal{E}_{\text{MST}(i)}$, which points from u to v, node u transmits its data to v as side information via (u,v). It is easy to show that this results in no cyclic data dependency.

Since the DSIT heuristic is no worse than any solution that uses shortest paths to transmit side information, we have the following sequence of inequalities.

$$\begin{aligned} C_{\text{DSIT}} &\leq b_0 \sum_{i\in\mathcal{O}} d_i + \beta b_0 \sum_{i\in\mathcal{N}_a\backslash\mathcal{O}} d_i + b_0 \sum_{i\in\mathcal{N}_a}\sum_{e\in\mathcal{E}_{\text{MST}(i)}} c_e \\ &\overset{(a)}{\leq} b_0 d_I + \beta b_0 \sum_{i\in\mathcal{N}_a\backslash\{I\}} d_i + b_0 \sum_{e\in\mathcal{E}_{\text{ST}}} c_e \\ &\quad - b_0 \sum_{i\in\mathcal{N}_a}\sum_{e\in\mathcal{E}_{\text{ST}(i)}} c_e + b_0 \sum_{i\in\mathcal{N}_a}\sum_{e\in\mathcal{E}_{\text{MST}(i)}} c_e \\ &\overset{(b)}{\leq} \beta b_0 \sum_{i\in\mathcal{N}_a} d_i + (2-\beta) b_0 \sum_{e\in\mathcal{E}_{\text{ST}}} c_e \\ &\quad + b_0 \sum_{i\in\mathcal{N}_a}\left(\frac{N_i-1}{N_i+1}\right)\sum_{e\in\mathcal{E}_{\text{ST}(i)}} c_e \\ &\overset{(c)}{\leq} \beta b_0 \sum_{i\in\mathcal{N}_a} d_i + \left(\frac{2N}{N+1} + 1 - \beta\right) b_0 \sum_{e\in\mathcal{E}_{\text{ST}}} c_e \end{aligned}$$

Step (a) uses (29). Step (b) comes from $d_I \leq \sum_{e\in\mathcal{E}_{\text{ST}}} c_e$ and (31). Step (c) is a result of (30) and note that $\mathcal{E}_{\text{ST}(i)} = \emptyset$ when $\max_{i\in\mathcal{N}_a} N_i = 0$. The second term on the right hand side of (28) is easily obtained from the above and (10). ∎

In general, (28) is not tight. An example of this bound is depicted in Fig. 10. Again, the worst-case routing performance is improved under high gain situations comparing to SPT.

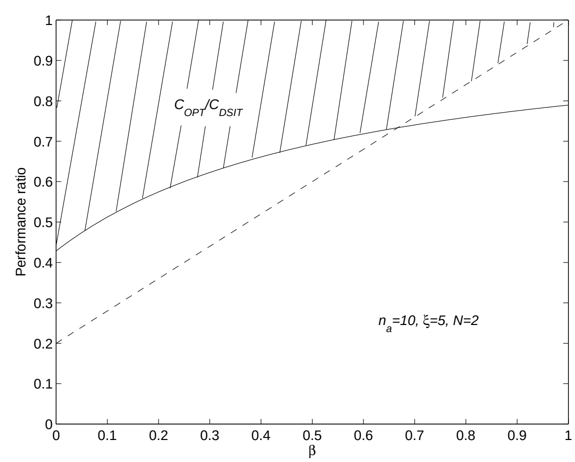


Fig. 10. The performance bound in Theorem 4 given $n_a = 10$, $\xi = 5$, and $N = 2$.

The worst-case scenario in Fig. 9 can be avoided by running multiple maximum weight branching and shortest path heuristic iterations in the DSIT heuristic. At each iteration, only one sensor is added to $\mathcal{S}_i, i \in \mathcal{N}_a$. However, this greatly increases the computational cost. Moreover, the pathological case in Fig. 9, where high correlation exists between sensors that are far away from one another but zero correlation among nearby sensors, is rarely a physical reality. The value of N is expected to be small as one's data help mostly nearby sensors. Also since side information is often circulated within one's neighborhood, using shortest paths to approximate a Steiner tree introduces a moderate amount of error. In the next section, the average behavior of different algorithms is examined through simulations.

VI. SIMULATIONS

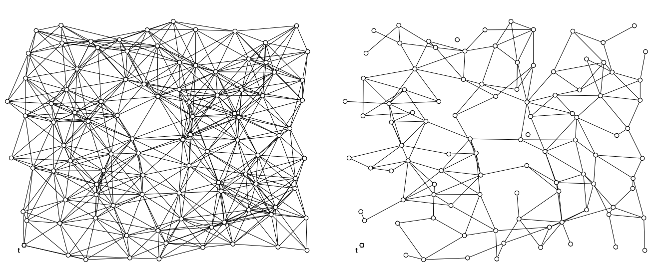
A. Simulation setup

In our simulations, we place $(n+1)$ nodes including the fusion center and n sensors in an $n_d \times n_d$ square, where $n_d = \lceil \sqrt{n+1} \rceil$. Denote by $\lceil z \rceil$ the smallest integer such that $\lceil z \rceil \leq z$, and $\lfloor z \rfloor$ the largest integer such that $\lfloor z \rfloor \geq z$. Supposing $\tilde{x}_i$ and $\tilde{y}_i, i = 1, \cdots, n+1$, are random variables that are uniformly distributed in $[0,1]$, the coordinates of node i is given by

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

We define a transmission radius r_c. If two nodes are no more than r_c away from each other, direct communication between the two nodes is allowed. Otherwise, relays have to be used. Denote by d_e the Euclidean length of edge e. When $d_e \leq r_c$, the edge weight c_e is proportional to d_e^{α}, where $\alpha = 2$ is the path loss factor. When the number of sensors increases, the network covers a larger area while maintaining the communication range and sensor to sensor spacing. A typical 100 node network constructed in this manner is depicted in Fig. 11(a). The node in the lower left corner is the fusion center.

In our simulation, we assume that all sensors are active. The helping set $\mathcal{H}_i$ of sensor i is defined as follows. Any pair of sensors that are no more than r_d away from one another have the probability of p_h to be in the helping sets of one another. Fig. 11(b) shows the resulting data correlation in the network. For simplicity, the data rate function in (7) is used.



(a) (b)

Fig. 11. Simulation setup: $r_c = \sqrt{5}$, $r_d = 1.8$, $p_h = 0.5$. Two nodes are connected if (a) direct transmission is allowed; (b) their data are correlated.

B. Simulation results

Denote by C_{SPT}, C_{CLU}, C_{BAS}, and C_{DSIT} the routing cost of SPT, cluster, BAS and DSIT heuristics. The cluster method is described in Section III-C. Define C_{ASP} to be the cost of routing data when an address centric SPT, in which no data aggregation is performed, is used. The performance ratios of heuristic algorithms to address centric SPT are computed:

$$\mu_s = \frac{C_{\text{SPT}}}{C_{\text{ASP}}},\ \mu_c = \frac{C_{\text{CLU}}}{C_{\text{ASP}}},\ \mu_d = \frac{C_{\text{DSIT}}}{C_{\text{ASP}}},\ \mu_b = \frac{C_{\text{BAS}}}{C_{\text{ASP}}}.$$

We simulate for different network sizes and vary the values of β and p_h. In each case, the performance ratios are computed by averaging over 20 randomized network setups.

The simulation results under high coding gain, $\beta = 0.1$, are plotted in Fig. 12, where $p_h = 0.5$. The wiggling of curves is mainly due to the irregular sensor distribution when $\sqrt{n+1}$ is not a integer. It has a more pronounced effect on the clustering method as it results in an uneven number of sensors in different clusters. It is apparent that all data-centric algorithms provide significant gains over the address-centric SPT, and performance ratios improve as the network size increases. The latter is because the data rate reduction affects the total cost more as the average distance to the fusion center increases. The ratios will eventually saturate before reaching β. Both BAS and DSIT are superior to SPT. However, the clustering method performs worse than SPT. This is not surprising considering that we group sensors based solely on geographical proximity. Possible improvements include taking into account data correlation in forming clusters, varying cluster size, and using data-centric routing for intra-cluster data transmission etc. The performance ratios are plotted under small coding gain, $\beta = 0.8$, in Fig. 13. The DSIT method suffers the most from the decrease of data correlation. It performs worse than the SPT algorithm. However, BAS still produces the best result.

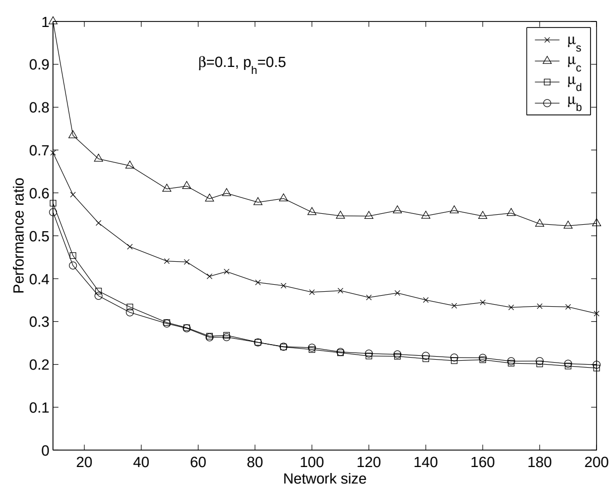


Fig. 12. Performance ratios plotted against network size when coding gain is high, $\beta = 0.1$, $p_h = 0.5$.

We also varied the value of p_h, and observed significant performance improvement for the clustering method and SPT when p_h was raised from 0.5 to 1. In contrast, the improvement on μ_b and μ_d were much smaller. It appears that the benefit of data correlation has been exploited fully in these two methods. Hence, increasing the size of $\mathcal{H}_i$ produces little additional gain.

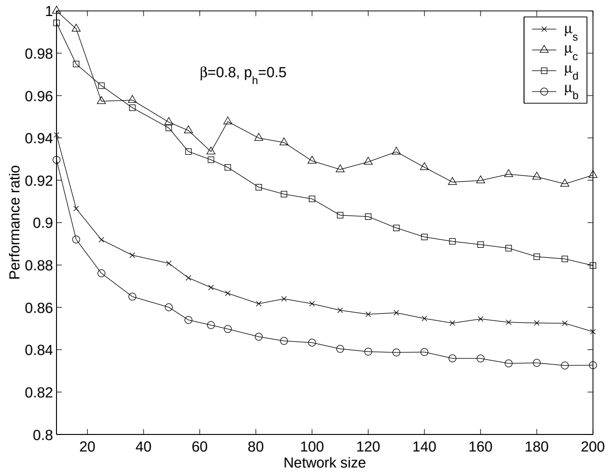


Fig. 13. Performance ratios plotted against network size when coding gain is low, $\beta = 0.8$, $p_h = 0.5$.

To better illustrate the effect of coding gain, we fix the network size at 100, and plot μ as a function of β in Fig. 14. It is observed that all ratios increase monotonically with β, which is expected. BAS has the smallest ratio under almost all coding gain conditions. DSIT has about the same performance as BAS when β is small, but μ_d surpasses μ_s at moderate coding gain and converges to 1 with μ_s and μ_b when β tends to 1. We also observe that $\mu_d, \mu_s, \mu_b \leq 1$ since these methods never perform worse than the address-centric SPT. In contrast, μ_c becomes greater than 1 when β is close to 1.

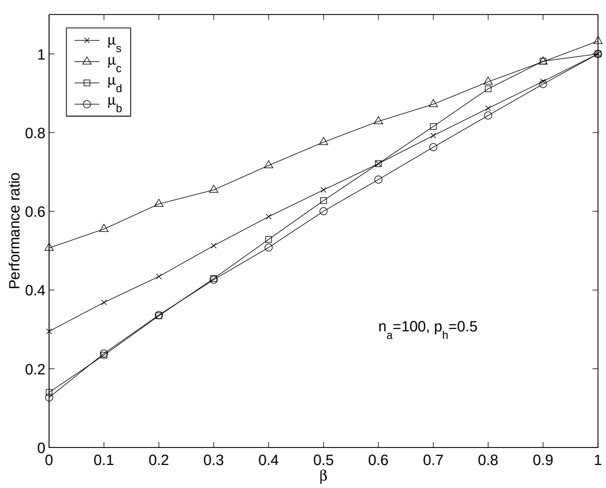


Fig. 14. Performance ratios plotted against β, $p_h = 0.5$.

C. Discussion

Our simulations highlight the importance of coding gain information in designing data-centric routes. The clustering method based on geographical proximity illustrates that serious performance loss may occur when routing is ill-advised.

The SPT fares fairly well when $p_h = 1$. In a SPT, sensors in neighborhoods have good chances of quickly merging their flows, which leads to data compression and cost reduction. However, in practice, sensors in a network may carry out different tasks and collect various types of data, so it is more common that p_h is less than 1. In addition, our simulations are conducted on networks where sensors are evenly distributed, and cases similar to the worst scenario in Fig. 4 rarely occur.

BAS and DSIT methods yield good results under high coding gain and converge to SPT when coding gain diminishes. The two schemes' ability to take advantage of data correlation is evident when p_h drops from 1 to 0.5. While decreasing the number of potential helpers gives rise to significant performance loss for SPT and clustering methods, μ_b and μ_d only dip slightly. Although our simulations show that the performance of DSIT deteriorates when β becomes greater than 0.5, the performance loss is relatively benign due to the low coding gain and the convergence of μ_d to 1. Although we only presented centralized algorithms for BAS and DSIT, distributed implementations are not difficult to devise given the extensive research that have been conducted on SPT and MWB.

VII. Conclusion

Our study continues the recent development of data-centric routing. The data transmissions are decomposed into individual flows originating at different sensors to build a simplified first order rate model. Based on this model, we formulate an optimization problem and show it to be NP-hard. Subsequently, two heuristic algorithms, BAS and DSIT, are proposed. Our analysis and simulations show that both methods are amenable to distributed implementation, have moderate complexity, and provide good performance under both high and low coding gains.

The algorithms proposed in this paper rely heavily on our assumed network model. In particular, we assume that data streams are highly correlated only when they are generated by a small group of sensors close to one another, and the coding gain of side information saturates when the number of helpers exceeds one. Hence, these methods may not be as effective in cases that deviate from these assumptions. Nonetheless, as we discussed in Section II-B, there are practical reasons to consider this simplified case. Furthermore, if more than one helper has to be considered, we speculate that the problem can be approached in a multi-step procedure. At each step, the number of helpers is restricted to at most one, and algorithms similar to our heuristic schemes are used. This is an area that needs further research.

Acknowledgement

We would like to thank reviewers for their valuable comments, and pointing out reference [3] to us.

References

[1] B. Krishnamachari, D. Estrin, and S. Wicker, "Modelling data-centric routing in wireless sensor network," University of Southern California, Tech. Rep. CENG 02-14, 2002.

[2] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, "On network correlated data gathering," in *Proc. IEEE Infocom*, Hongkong, China, March 2004.

[3] P. von Rickenbach and R. Wattenhofer, "Gathering correlated data in sensor networks," in *DIALM-POMC*, Philadelphia, Oct. 2004.

[4] A. Goel and D. Estrin, "Simultaneous optimization for concave costs: single sink aggregation or single source buy-at-bulk," in *ACM/SIMA Symposium on Discrete Algorithms*, 2003.
[5] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed diffusion for wireless sensor networking," *IEEE/ACM Trans. Netw.*, vol. 11, no. 1, pp. 2–16, Feb 2003.
[6] S. Khuller, B. Raghavachari, and N. Young, "Balancing minimum spanning trees and shortest-path trees," *Algorithmica*, vol. 14, pp. 305–321, 1995.
[7] G. Pottie and W. Kaiser, "Wireless sensor networks," *Communications of ACM*, vol. 43, no. 5, pp. 51–58, May 2000.
[8] R. Zamir and T. Berger, "Multiterminal source coding with high resolution," *IEEE Trans. Inf. Theory*, vol. 45, no. 1, pp. 106–117, 1999.
[9] S. S. Pradhan, J. Kusuma, and K. Ramchandran, "Distributed compression in a dense microsensor network," *IEEE Commun. Mag.*, vol. 19, pp. 51–60, March 2002.
[10] Z. Xiong, A. D. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *IEEE Signal Process. Mag.*, vol. 21, pp. 80–94, Sept 2004.
[11] H. Luo, Y. Tong, and G. Pottie, "A two-stage dpcm scheme for wireless sensor networks," in *Proc. IEEE International Conference on Accoustic, Speech, and Signal Processing*, Philadelphia, USA, 2005.
[12] J. Chen, L. Yip, J. Elson, H. Wang, D. Maniezzo, R. E. Hudson, K. Yao, and D. Estrin, "Coherent acoustic array processing and localization on wireless sesnor networks," *Proc. IEEE*, vol. 91, no. 8, pp. 1154–1162, Aug 2003.
[13] A. Scaglione and S. Servetto, "On the interdependence of routing and data compression in multi-hop sensor networks," in *Proc. ACM/IEEE Mobicom*, 2002.
[14] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Proc. IEEE Infocom*, 2003.
[15] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication procotol for wireless microsensor networks," in *Proc. Hawaii International Conference on System Sciences*, Jan 2000.
[16] K. Bharath-Kumar and J. M. Jaffe, "Routing to multiple destinations in computer networks," *IEEE Trans. Commun.*, vol. 31, no. 3, pp. 343–351, 1983.
[17] H. Luo and G. Pottie, "Routing explicit side information for data compression in wireless sensor networks," in *Proc. International Conference on Distributed Computing in Sensor Systems*, Marina del Rey, CA, USA, June 2005.
[18] ——, "Balanced aggregation trees for routing correlated data in wireless sensor networks," in *Proc. IEEE International Symposium on Information Theory*, Adelaide, Australia, Sept. 2005.
[19] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
[20] J. H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," in *Proc. IEEE Infocom*, 2000.
[21] S. Singh, M. Woo, and C. S. Raghavendra, "Power-aware routing in mobile ad-hoc networks," in *Proc. ACM/IEEE Mobicom*, Dallas, Texas, 1998.
[22] Y. J. Chu and T. H. Liu, "On the shortest arborescence of a directed graph," *Science Sinica*, vol. 14, pp. 1396–1400, 1965.
[23] J. Edmonds, "Optimum branchings," *J. Research of the National Bureau of Standards*, vol. 71B, pp. 233–240, 1967.
[24] H. Takahashi and A. Matsuyama, "An approximate solution for the steiner problem in graphs," *Math. Japonica*, vol. 24, no. 6, pp. 573–577, 1980.
[25] S. Haldar, "An 'all pairs shortest paths' distributed algorithm using $2n^2$ messages," *Journal of Algorithms*, vol. 24, pp. 20–36, 1997.
[26] P. A. Humblet, "A distributed algorithm for minimum weight directed spanning trees," *IEEE Trans. Commun.*, vol. 31, pp. 756–762, June 1983.
[27] F. K. Hwang, D. S. Richards, and P. Winter, *The Steiner Tree Problem*. North-Holland, 1992.

APPENDIX

In this appendix, we prove the following proposition.

Proposition 3: Given real numbers a_1, a_2, $\cdots$, a_n and variables $\theta_1, \theta_2, \cdots, \theta_n$ that satisfies the following conditions

$$\begin{aligned} &a_1 \geq a_2 \geq \cdots \geq a_n \geq 0, \\ &\theta_n \geq \cdots \geq \theta_2 \geq \theta_1 > 0, \end{aligned} \tag{32}$$

we have the inequality

$$r(\theta_1, \cdots, \theta_n) = \frac{\sum_{i=1}^n a_i \theta_i}{\sum_{i=1}^n \theta_i} \leq \frac{\sum_{i=1}^n a_i}{n}, \tag{33}$$

and the equality is achieved when $\theta_1 = \theta_2 = \cdots = \theta_n$.

Proof: We first show that the value of $r(\theta_1, \cdots, \theta_n)$ can be increased if $\theta_i \neq \theta_{i+1}$ for any $1 \leq i \leq n-1$. Due to condition (32), we have $\theta_i < \theta_{i+1}$, if $\theta_i \neq \theta_{i+1}$. Construct a new sequence as follows

$$\theta'_j = \begin{cases} (\theta_i + \theta_{i+1})/2 & j = i \text{ or } i+1; \\ \theta_j & \text{otherwise.} \end{cases}$$

It is an easy matter to show that

$$r(\theta_1, \cdots, \theta_n) \leq r(\theta'_1, \cdots, \theta'_n)$$

Keep performing this action as long as there are unequal pairs of (θ_i, θ_{i+1}), eventually θ_i will converge to $\sum_{j=1}^n \theta_j / n$, and we obtain (33). ■

Huiyu Luo was born in China. He received his B.S. in Mechanical Engineering from University of Science and Technology of China in 1999. He then attended University of California, Los Angeles, and earned his M.S. and Ph.D. degrees, both in Electrical Engineering, in 2002 and 2005 respectively.

Gregory J. Pottie was born in Wilmington DE and raised in Ottawa, Canada. He received his B.Sc. in Engineering Physics from Queen's University, Kingston, Ontario in 1984, and his M.Eng. and Ph.D. in Electrical Engineering from McMaster University, Hamilton, Ontario, in 1985 and 1988 respectively. From 1989 to 1991 he worked in the transmission research department of Motorola/Codex in Canton MA, with projects related to voice band modems and digital subscriber lines. Since 1991 he has been a faculty member of the UCLA Electrical Engineering Department, serving in vice-chair roles from 1999-2003. Since 2003 he has served as Associate Dean for Research and Physical Resources of the Henry Samueli School of Engineering and Applied Science. From 1997 to 1999 he was secretary to the board of governors for the IEEE Information Theory Society. In 2005 he became a Fellow of the IEEE for contributions to the modeling and applications of sensor networks.