Data Gathering with Tunable Compression in Sensor Networks

Yang Yu, *Member, IEEE,* Bhaskar Krishnamachari, *Member, IEEE*
and Viktor K. Prasanna, *Fellow, IEEE,*

***Abstract*— We study the problem of constructing a data gathering tree over a wireless sensor network in order to minimize the total energy for compressing and transporting information from a set of source nodes to the sink. This problem is crucial for advanced computation-intensive applications, where traditional "maximum" in-network compression may result in significant computation energy. We investigate a tunable data compression technique that enables effective tradeoffs between the computation and communication costs. We derive the optimal compression strategy for a given data gathering tree and then investigate the performance of different tree structures for networks deployed on a grid topology as well as general graphs. Our analytical results pertaining to the grid topology and simulation results pertaining to the general graphs indicate that the performance of a simple greedy approximation to the Minimal Steiner Tree (MST) provides a constant-factor approximation for the grid topology and good average performance on the general graphs. Although theoretically, a more complicated randomized algorithm offers a poly-logarithmic performance bound, the simple greedy approximation of MST is attractive for practical implementation.**

I. Introduction

For wireless sensor networks, in-network data compression is vital for reducing the communication cost over the routing substrate (e.g., a data gathering tree). State of the art techniques perform a "maximum" in-network compression [1], [2], where the output data is compressed as much as possible from the input data (we consider lossless compression). Given that the computation energy of data compression is negligible for simple applications (e.g., temperature sensing), to perform maximum compression for energy saving is understandable. However, for advanced applications with heavy data flow, including structural health monitoring, video surveillance, and image-based tracking, compression of complex data sets is envisioned to cost energy comparable with wireless communication. Similar situation arises for batch mode data gathering, where a large volume of sensed data is accumulated at source nodes over a long time period before being transmitted to the sink node. It is also shown that with short-range communication, blindly applying maximum compression may lead to extra energy cost compared to transmitting the raw data [3]. This necessitates alternative methods instead of maximum compression for efficient tradeoffs between computation and communication costs [4].

Motivated by the above observation, we investigate the concept of **tunable compression** that is capable of tuning the computation complexity of **lossless** data compression based on the energy availability. Such a concept is not new in itself. For example, the well-known *gzip* program supports up to ten levels of different compression ratio, with larger compression ratio resulting in longer compression time and hence higher energy cost [5], [6]. However, prior works have not considered the application of tunable compression together with routing techniques for data gathering in sensor networks, which is the focus of this paper.

We consider the construction of a data gathering tree spanning a set of source nodes and rooted at a sink node. For this problem, two important data compression schemes have been previously investigated: distributed source coding [7] and compression with explicit communication [8]. While practical distributed source coding schemes for sensor networks are being developed [9], most existing

This work is supported by NSF under grants 0330445, 0325875, and 0347621.

Y. Yu is with the Application Research Center, Motorola Labs, Schaumburg, IL 60196 (e-mail: yang@motorola.com). The presented work was performed when he was at the University of Southern California.

V. K. Prasana and B. Krishnamachari are with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089-2562 (e-mail: {prasanna, bkrishna}@usc.edu).

Digital Object Indentifier 10.1109/TPDS.2007.70709

works for data compression schemes are based on explicit communication [1], [2], [4], [8], [10], [11]. Thus, we also focus on joint data compression with side information via explicit communication.

While most prior works on data gathering focus on minimizing the communication cost only, our goal is to minimize the sum of both computation and communication costs using tunable compression. We refer to our problem as the Tunable Compression-based Data Gathering (TCDG) problem. To tune data compression over the gathering tree, we propose a flow based model where data from each source is compressed and transmitted as a data flow over the corresponding path from the source to the sink. Since a special case of TCDG reduces to the Minimal Steiner Tree (MST) problem, TCDG is NP-Hard in general.

We solve TCDG by decoupling two subproblems: tree construction and flow determination. We first show how the optimal flow can be determined for a given tree. By assuming a network deployed on a grid topology, we then model and analyze and performance of two existing tree structures, MST and the Shortest Path Tree (SPT). The results indicate that while SPT performs well when the relative computation cost compared to communication cost is high, MST is preferred when the relative computation cost is low and data correlation is high. Moreover, MST provides a constant-factor approximation for the grid deployment.

We also study the performance of an approximated MST (A-MST) and SPT for general graphs through simulation. Our results further reveal the tradeoffs between A-MST and SPT with respect to variations in data correlation and relative computation cost. Moreover, A-MST demonstrates acceptable average performance in the studied scenarios, which leads to the conclusion that A-MST is suitable as a practical solution due to its simplicity. For theoretical completeness, we also present a randomized tree construction methodology that achieves poly-logarithmic approximation for general graphs.

Paper Organization: The related work is briefly discussed in Section II. In Section III, we give assumptions and models for the TCDG problem, which is formally defined in Section IV. We then show how to determine the optimal flow for a given tree in Section V, which enables the performance analysis of SPT and MST on a grid deployment in Section VI. In Section VII, a randomized approximation algorithm is described. Simulation results are presented in Section VIII. Finally, concluding remarks are given in Section IX.

II. Related Work

The problem of constructing an energy-efficient data gathering tree in sensor networks with data compression has received increasing research attention. The work by Pattem *et. al.* [2] investigates several practical schemes for tree construction, including routing-driven compression (RDC), compression-driven routing (CDR), and cluster based routing. Essentially, RDC involves opportunistic data compression over an SPT; CDR performs maximum possible compression using a MST-like routing scheme; and cluster based routing is a hybrid scheme of RDC and CDR. Empirical results show that cluster based routing achieves near-optimal performance for a wide range of spatial correlations. By assuming the entropy conditioning at nodes to be only depending on the availability of side information, Cristescu *et. al.* also show that a hybrid scheme of SPT and MST provides constant performance approximation for minimizing the overall communication cost [8]. For a very similar problem, when the joint entropy being modeled as a concave, but unknown function of the number of sources, a randomized logarithmic approximation algorithm is given in [10]. Also, Goel *et. al.* [10] noticed that the data gathering problem is essentially a single-source buy-at-bulk problem [12], where the cost spent on each edge is a concave function of the number of sources that use this edge to communicate to the sink. Other related efforts can be found in [11], [13]–[18].

The adaptation of compression has been widely studied in video transmission to mobile, wireless devices [19], [20]. With lossy video compression, these papers focus on vary the fidelity of the transmitted video in response to changing network condition [19] or wireless channel quality [20]. Our paper, on the contrary, consider lossless compression and emphasize the tradeoffs between computation and communication costs. Also, the underlying data gathering problem has the inherent feature of joint compression for data from multiple sources, which is not presented in case of video compression in general mobile networks.

Very few prior papers have exploited the tradeoffs between computation and communication for

TABLE I
TABLE OF NOTATIONS

$G = < V, E, w >$	the graph representing the sensor network with a set of sensor nodes V, a set of links, E, and a set of weights on edges in E
R	set of source nodes, $R \subseteq V$
$sink$	the sink node in V
w_e	the weight of edge $e \in E$
δ_v	number of source nodes in a subtree rooted at $v \in V$
$p(v)$	the path from $v \in R$ to $sink$ in a given tree
$z(v)$	the last edge on $p(v)$, i.e., the edge to $sink$
γ	relative computation cost for compressing
f_e^v	flow from node v on edge e
H_i	joint entropy of $i \geq 1$ unit data
ρ	data entropy rate, i.e., $\rho = H_1$
L_i	the lower bound of compression for one unit data when jointly compressed with $i-1$ pieces of unit data, $L_i = \frac{H_i}{i}$
ε	energy costs

data gathering [4]. For data gathering over a one-dimensional random Gaussian field, such tradeoffs are enabled by adjusting the group size within which data fusion is performed — large group increases computation cost but decreases communication cost [4]. Simulation results indicate that the optimal group size increases with its distance to the sink. Our paper formally models and studies the tradeoffs between computation and communication costs for data gathering in sensor networks with a general problem setting.

Although we model joint entropy to be a concave function of number of sources, the results in [10] and [12] cannot be directly applied to our problem. This is because when the computation energy is considered, the overall cost on each edge may not be a concave function of the number of sources using this edge to communicate to the sink. Our work shows that by using the notion of probabilistic metric approximation [21], a randomized algorithm gives an expected $O(\log^2 n)$ approximation solution. It is worth noting that the approximation bound can be further improved to $O(\log n)$ [22]. However here we illustrate the tradeoffs between SPT and MST, hence the results in [21] suffice.

III. Models and Assumptions

A. Network Model

A list of notations is given in Table I.

We assume a simplified communication mechanism with a medium access control (MAC) protocol that ensures no packet collisions or interference in the network [23]. This assumption has also been made in several prior papers on data gathering [2], [8], [10]. Thus, the underlying wireless network is modeled as a connected weighted graph $G =< V, E, w >$, where the vertex set V represents the set of n sensor nodes; the edge (link) set E represents the wireless connection between nodes; and associated with each edge $e_i \in E$, its weight w_{e_i} (or simply w_i) abstracts the energy cost of sending a data packet of unit size over e_i. The edge weight is determined by the distance between the two adjacent nodes, the radio device, and the communication environment. We also use (u, v) to denote an edge connecting u and v.

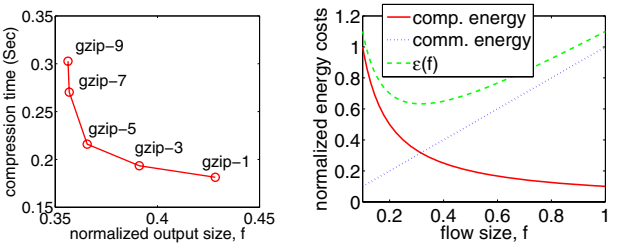
Let $sink \in V$ denote the sink node and $R \subseteq V$ denote the set of source nodes. We consider an epoch-based data gathering paradigm [24]. In each epoch, each source node generates a raw data of one unit size that needs to be transported to the sink, possibly via multi-hop communication.

A data gathering tree is a subtree of G rooted at $sink$ and containing R, denoted as $T =< V', E' >$, where $R \subseteq V' \subseteq V$ and $E' \subseteq E$. Let δ_v denote the number of source nodes in the subtree rooted at v. Given a data gathering tree, let $p(v)$ denote the path in the tree that connects v to $sink$, with $u \in p(v)$ ($e \in p(v)$) signifying that node u (edge e) is along the path $p(v)$. Also, for two edges e_1 , e_2 on the same path, let $e_1 \prec e_2$ denote the fact that e_1 is a predecessor of e_2.

B. Energy Model for Tunable Compression

Since it is difficult to define a general form to characterize the energy costs of various compressing schemes, we use a simple model that captures the principle rationale: the computation time complexity of compressing one unit of data is inversely proportional to the output size. Further, the energy cost is proportional to the time complexity [5]. We illustrate the above rationale using the example of *gzip* to compress the benchmark file "alice29.txt" from the Canterbury Corpus [6] at 5 different levels of compression ratio (by properly parameterizing *gzip*). The curve of running time *vs.* the normalized output size is shown in Figure 1(a). The results are averaged over 20 runs on a SUN SPARC II machine.

In fact, similar compression time *vs.* normalized output size tradeoffs are observed for a collection of various compression techniques [5].



(a) Experimental results of compression time *vs.* normalized output size for *gzip* (the curve is modeled as function $g(f) = \frac{\gamma}{f}$ in this paper)

(b) Computation energy *vs.* communication energy tradeoffs on a single link ($w_e = 1$, $\gamma = 0.1$)

Fig. 1. Energy tradeoffs with tunable compression

We define a pre-specified system parameter, $\gamma \geq 0$, to abstract the energy cost of compressing one unit of data normalized by the cost of communicating one unit of data. Following the above rationale and for the purpose of illustration, the energy cost of compressing a source information of size s to an output of size f is modeled as function

$$g(f) = \gamma s \frac{s}{f} . \tag{1}$$

The intuition behind Eq. (1) is that the energy cost is (1) proportional to the input size s since it has to scan the whole input at least once, and (2) proportional to the compression ratio given by $\frac{s}{f}$.

We now illustrate the fundamental tradeoffs between computation and communication energy using the example of a one-hop link. Let $e = (u, v)$ denote the link, where u generates a data packet of one unit size that needs to be transmitted to v after appropriate compression by u. Let f denote the output size of the compression by u, which is also the size of the data flow over e. f is lower bounded by the entropy of one unit of data, denoted as ρ. Since $s = 1$, we simplify the energy function in Eq. (1) to $g(f) = \frac{\gamma}{f}$, which is also used throughout this paper. Let w_e denote the cost of transmitting one unit of data over e. The overall energy costs, denoted as $\varepsilon(f)$ can then be modeled as follows:

$$\varepsilon(f) = \frac{\gamma}{f} + f \cdot w_e . \tag{2}$$

We plot $\varepsilon(f)$ in Figure 1(b) with $w_e = 1$, $\gamma = 0.1$ and $f \in [0.1, 1]$ (we omit the boundary effect of ρ as for now). Intuitively, $w_e = 1$ means that to transmit one unit of data costs one unit energy. Since the energy of transmitting one bit is typically around 500 - 1000 times greater than a single 32-bit computation [25], the practical meaning behind $\gamma = 0.1$ is that around 50 - 100 instructions need to be executed for generating each bit in the output.

Clearly, $\varepsilon(f)$ is convex and the minimum is achieved when $\varepsilon'(f) = 0$, where $\varepsilon'(f)$ is the first derivative of $\varepsilon(f)$. Let f_0 denote the desired flow with $\varepsilon'(f_0) = 0$. We derive $f_0 = \sqrt{\frac{\gamma}{w_e}}$. Considering the boundary effects of ρ, the optimal value of f equals f_0 if $\varepsilon'(\rho) \leq 0$ and $\varepsilon'(1) \geq 0$, or ρ if $\varepsilon'(\rho) \geq 0$, or 1 if $\varepsilon'(1) \leq 0$.

C. Flow-Based Data Gathering

Given a data gathering tree over a sensor network, we model data transmission over the tree as a composition of different data flows from each source node to $sink$. That is, each path from a source node to $sink$ in the tree corresponds to a data flow over the path. The flow size may change along its corresponding path due to data compression performed by intermediate nodes. Also, the energy cost of the system is the sum of the computation and communication costs of all paths in the tree.

Consider an arbitrary path $p(v)$ in the tree from a source node v to $sink$. Let f_e^v denote the flow over $e \in p(v)$ and $z(v)$ denote the last edge in $p(v)$, i.e, the edge incident to $sink$ in $p(v)$. We assume that the total energy spent on data compression over the path $p(v)$ is determined by the flow on $z(v)$, i.e., the total energy cost for data compression over $p(v)$ is calculated as $\frac{\gamma}{f_{z(v)}^v}$. We will discuss this assumption in Section III-D.

Given a node in the tree, the number of incoming flows equals the number of source nodes in its subtree. The output size for compressing each incoming packet is lower bounded by the joint entropy of these source nodes. Following the entropy model in [10] (which effectively abstracts the entropy models in [2], [8]), we assume that the joint entropy of any i source nodes, H_i is a non-decreasing and concave function of i with $H_1 = \rho$, where $\rho \in (0, 1]$ is the entropy of one unit of data. We assume that the compression of i incoming data flows at node v can be performed in such a way that the lower bound for compressing each data flow equals $L_i = \frac{H_i}{i}$, with $L_1 = H_1 = \rho$. In other words, we assume that

when maximal compression is performed on i pieces of source information, the fraction of compressible data of each piece is the same.

Thus, for any $e = (a, b) \in p(v)$, we impose the constraint on f_e^v such that $f_e^v \geq L_{\delta_a} = \frac{H_{\delta_a}}{\delta_a}$ (recall that δ_a is the number of incoming data flows to a). This constraint is further explained using an example in Section III-E. We also assume that L_i decreases with i, i.e., $L_i \geq L_{i+1}$ for $i \geq 1$ (see footnote 1 in Section VI-B for a practical example). Hence, when a data flow is compressed and transmitted along a path, the lower bound on the flow decreases as the packet approaches $sink$.

D. Discussion

First, our analysis is not restricted to the specific $g(f)$ in Eq. (1). In fact, while the energy characteristics of various compression algorithms have been studied [5], accurate models for abstracting the energy cost of tunable compression are still open problems. However, the tradeoffs between computation and communication costs essentially depend on the convexity of the total energy cost function, e.g., Eq. (2). The requirement that energy cost is inversely proportional to compression ratio is one nice example leading to such a convexity. We expect other models to be investigated in this context.

Second, the above flow model naturally models the data (information) streaming from sources to the sink and facilitates the computation of energy cost of compression. This paper considers only energy cost under this flow model. Other performance metrics such as delivery latency can be defined by virtually combining different outgoing flows from a node as a whole and assessing the resulting time cost accordingly.

Also, this paper is based on the simplified assumption that the joint entropy of any set of i sources is H_i and the flow from any of the i sources is lower bounded by L_i after joint compression. To incorporate other more sophisticated joint entropy models is part of our future work.

Third, by determining the compression energy over a path solely based on the output flow of the last compression on the path, we abstract away decompression and compressions at intermediate nodes. Since techniques such as *gzip* consumes very little time for decompression compared to compression [5], to ignore the decompression cost is acceptable. However, this assumption under-estimates the overall compression cost by ignoring the compressions at intermediate nodes. This is tolerable if the last compression along the path dominates the overall compression cost, which prefers the case where few compressions are performed along the path and the data correlation in the last compression is so high that the energy cost for this particular compression dominates previous compressions along the path. Nevertheless, we hope this preliminary study leads to more accurate model of compression cost in the future.

Fourth, since our problem is to minimize the overall energy cost, the energy for packet reception can be easily incorporated into the TCDG problem by adjusting the weight on edges.

E. An Example

We illustrate the flow model using the data gathering tree in Figure 2, where nodes v_1, v_5, v_6, and v_7 are source nodes, v_2 and v_3 are relaying nodes, and v_4 is $sink$. In total, we have 4 paths in this tree.

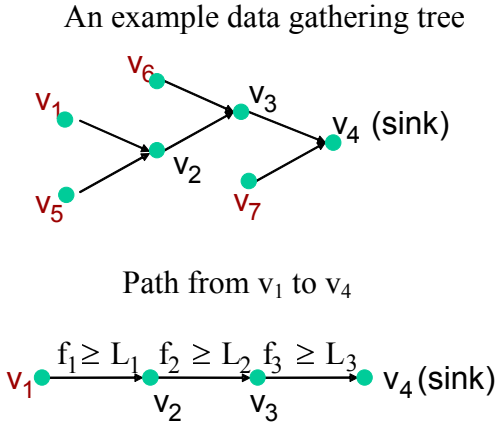


Fig. 2. An example data gathering tree and a path within it

Consider the path from v_1, denoted as $\{v_1, v_2, v_3, v_4\}$. Based on the structure of the tree, there is 1 source node (v_1 itself) in the subtree rooted at v_1, 2 source nodes (v_1 and v_5) in the subtree rooted at v_2 and 3 source nodes (v_1, v_5, and v_6) in the subtree rooted at v_3. Hence, the lower bound of flow on the path can be calculated as $L_{\delta_{v_1}} = L_1 = H_1$ on link (v_1, v_2), $L_{\delta_{v_2}} = L_2 = \frac{H_2}{2}$ on link (v_2, v_3), and $L_{\delta_{v_3}} = L_3 = \frac{H_3}{3}$ on link (v_3, v_4). The path together with the lower bounds of flow on each link are illustrated in Figure 2 (the superscription for f_i^v is omitted in the figure). Based on our model, we also have $L_1 \geq L_2 \geq L_3$.

Similarly, the flow on the path from v_5 to $sink$ is lower bounded by L_1 on link (v_5, v_2), L_2 on link (v_2, v_3), and L_3 on link (v_3, v_4). The flow on the path from v_6 is lower bounded by L_1 on link (v_6, v_3) and L_3 on link (v_3, v_4), respectively. The flow on the link $(v_7, sink)$ is lower bounded by L_1.

IV. Problem Definition

We formally define the **Tunable Compression-based Data Gathering (TCDG)** problem as:
Given:
(i) a weighted graph $G =< V, E, w >$, $sink \in V$, set of source nodes $R \subseteq V$,
(ii) an energy function for data compression given by Eq. (1) with a pre-specified parameter γ,
(iii) the joint entropy of i sources, H_i and $L_i = \frac{H_i}{i}$,
find *a subtree $T =< V', E' >$ that contains R and $sink$, and flow from all $v \in R$ to $sink$, so as to minimize*

$$\sum_{v \in R} \left(\frac{\gamma}{f^v_{z(v)}} + \sum_{e \in p(v)} f^v_e w_i\right) , \tag{3}$$

where $z(v)$ is the last edge on path $p(v)$,
subject to

$$\forall v \in R, \forall e = (a, b) \in p(v) \quad \Rightarrow \quad f^v_e \geq L_{\delta_a} \tag{4}$$

$$\forall v \in R, \forall e_1 \prec e_2 \in p(v) \quad \Rightarrow \quad f^v_{e_1} \geq f^v_{e_2} , \tag{5}$$

where δ_a is the number of source nodes in the subtree rooted at a. Note that in Eq. (3), when $f^v_{z(v)} = 1$, we still count γ as the computation cost. Since γ is close to zero in most application scenarios, this is acceptable.

We consider two interesting special cases of the TCDG problem. In the first case, we assume $\gamma = \infty$ or $H_i = i$, i.e., computation energy is arbitrarily high or data is uncorrelated. Either condition leads to the solution that no compression shall be performed. Thus, an SPT tree that combines the shortest weighted path from every node in R to $sink$ is optimal. In the second case, we assume $\gamma = 0$ and $H_i = 1$, i.e., computation energy is negligible and the joint entropy of any arbitrary i source nodes is always one. Thus, the desired flow on all edges of the tree equals one. In this case, TCDG reduces to the MST problem. Thus, TCDG is NP-Hard in general. To cope with this NP-Hardness, we start our study by decoupling the two subproblems of selecting the tree construction and determining the flow from each source node to the sink.

V. Optimal Flow on a Given Tree

Given a data gathering tree and an arbitrary source node $v \in R$, consider the path from v to $sink$. Without loss of generality, let $p(v) = \{v_1, v_2, \ldots, v_k\}$ denote the path, where $v_1 = v$, $v_k = sink$, and k is the number of nodes along $p(v)$. We need to compress and transmit a packet of unit size from v_1 to $sink$ with the minimal computation and communication energy costs. Let $\vec{f}$ denote a vector of flow along $p(v)$, i.e., $\vec{f} = \{f^v_{e_1}, \ldots, f^v_{e_{k-1}}\}$. Since we are considering the specific path $p(v)$, we omit the superscription of elements in vector $\vec{f}$ as well as e in the subscription. Hence, we use $\vec{f} = \{f_1, \ldots, f_{k-1}\}$ to denote the flow vector.

To simplify the notation, let β_i denote the lower bound of f_i, where $i = 1, \ldots, k-1$. Since the path is extracted from a given tree, we can calculate β_i based on the structure of the tree (as shown by the example in Section III-E). That is, $\beta_i = L_{\delta_{v_i}}$, where δ_{v_i} is the number of source nodes in the subtree rooted at v_i. Also, we have $\beta_i \geq \beta_{i+1}$, for $i = 1, \ldots, k-1$.

Let w_i denote the weight of $e_i = (v_i, v_{i+1})$, where $i = 1, \ldots, k-1$. Let W_i denote the sum of edge weights from e_i to e_{k-1}, i.e., $W_i = \sum_{j=i}^{k-1} w_j$. We slightly abuse the notation by letting $\beta_0 = 1$, and $W_k = 0$.

A. Example Revisited

Let $\vec{f} = \{f_1, f_2, f_3\}$ denote the optimal flow on path v_1 to v_4 shown in Figure 2. For this flow, we have $\beta_1 = L_1$, $\beta_2 = L_2$, and $\beta_3 = L_3$. Intuitively, when the relative computation cost increases, the optimal solution shall perform less amount of compression. In the trivial case when the computation cost is prohibitively high, i.e., $\gamma \geq W_1$, no compression is performed and we have the optimal flow as $f_1 = f_2 = f_3 = 1$. Otherwise, the optimal flow can be obtained by examining the following three cases, depending on the relative cost of computation, which is abstracted by γ and w_i's.

1) The cost of compressing the input down to β_1 at node v_1 is more expensive than routing data of size β_1 along the path. Thus, we may reduce the compression energy, and hence the overall energy cost, by decreasing the compression ratio. That is, the optimal solution is to let v_1 compress the data to some $x \in (\beta_1, 1]$ and set $f_1 = f_2 = f_3 = x$.

TABLE II
OPTIMAL FLOW FOR THE EXAMPLE PATH IN FIGURE 2

case	conditions	optimal flow
1	$\gamma \geq W_1$	$f_1 = f_2 = f_3 = 1$
2	$\gamma < W_1$ and $\frac{\gamma}{\beta_1} \geq W_2\beta_1$	$f_1 = f_2 = f_3 \in (\beta_1, 1]$
3	$\frac{\gamma}{\beta_1} < W_2\beta_1$ and $\frac{\gamma}{\beta_2} \geq W_3\beta_2$	$f_1 = \beta_1$, $f_2 = f_3 \in (\beta_2, \beta_1]$
4	$\frac{\gamma}{\beta_2} < W_3\beta_2$	$f_1 = \beta_1$, $f_2 = \beta_2$, $f_3 \in (\beta_3, \beta_2]$

2) Otherwise, further compression at node v_2 is necessary for reducing the total cost. If the cost of compressing the input at v_2 to β_2 is more expensive than the communication cost of routing β_2 over e_2 and e_3, the optimal solution is to set $f_1 = \beta_1$ and $f_2 = f_3 \in (\beta_2, \beta_1]$.
3) Otherwise, the compression is so cheap that it is also beneficial to perform one more compression at node v_3. In this case, the optimal flow is $f_1 = \beta_1$, $f_2 = \beta_2$, and $f_3 \in (\beta_3, \beta_2]$.

The above description is summarized in Table II.

B. Determining the Optimal Flow

Based on the intuition of the previous example, we derive the optimal $\vec{f}$ as follows.

Lemma 1: For any optimal flow $\vec{f}$ over a path $p(v)$ as previously described, if $f_{i+1} < f_i$, we have $f_i = \beta_i$.

Theorem 1: Given a path $p(v)$ as previously described, if $\gamma \geq W_1$, the optimal flow is of unit size on all links. Otherwise, suppose that $\gamma \in (W_{i+1}\beta_i^2, W_i\beta_{i-1}^2]$ for some $1 \leq i \leq k-1$. Then, the optimal flow $\vec{f}$ is:

$$\vec{f} = \{\beta_1, \beta_2, \ldots, \beta_{i-2}, \beta_{i-1}, \underbrace{f^*, \ldots, f^*}_{k-i}\}, \quad (6)$$

where $f^* = \max\{\beta_i, \sqrt{\frac{\gamma}{W_i}}\}$.

The proofs of Lemma 1 and Theorem 1 are detailed in Appendix I. From Theorem 1, the optimal flow is trivial when $\gamma \geq W_1$. Thus, we focus on the case $\gamma < W_1$ in the following discussion.

Theorem 1 reveals the fact that for an optimal flow from v to $sink$, if $\gamma \in (W_{i+1}\beta_i^2, W_i\beta_{i-1}^2]$ for some $1 \leq i \leq k-1$, the flow on the last $k-i$ edges, f^*, remains the same. For a closer understanding of f^*, in Figure 3, we plot f^* as a function of γ for the example path from v_1 to v_4 in Figure 2 by setting $w_1 = w_2 = w_3 = 1$, $\beta_1 = 0.7$, $\beta_2 = 0.6$, and $\beta_3 = 0.5$. The labels on the x-axis are $a_1 = W_3\beta_3^2$, $a_2 = W_3\beta_2^2$, $a_3 = W_2\beta_2^2$, $a_4 = W_2\beta_1^2$, and $a_5 = W_1\beta_1^2$. It can be observed, for example, that when $\gamma \in (a_5, W_1]$, f^* equals $\sqrt{\frac{\gamma}{W_1}}$. When γ is decreased to be within $(a_4, a_5]$, f^* is however lower bounded by β_1, as indicated by Theorem 1.

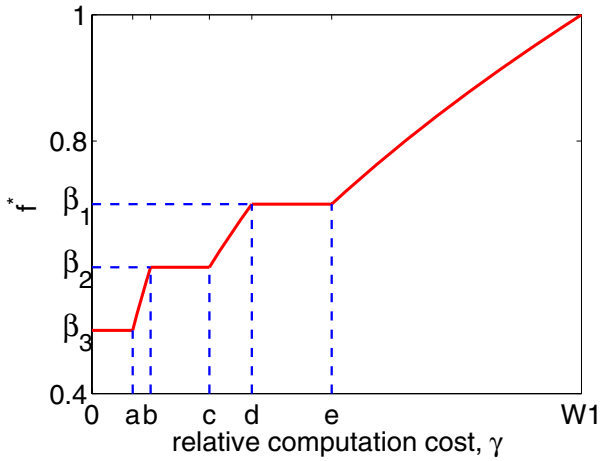


Fig. 3. f^* for the example path in Figure 2 as a function of γ

Moreover, let $Diam(sink, R)$ denote the weighted diameter of G with respect to R and $sink$, i.e., the maximum among the shortest weighted path from any node in R to $sink$. We define $\gamma^* = Diam(sink, R) \times L_1^2$ as the *critical point* of the system. From Theorem 1, we have:

Corollary 1: Given G, if $\gamma \geq \gamma^*$, SPT is the optimal tree for the TCDG problem, with the flow specified by Theorem 1.

VI. ANALYTICAL STUDY OF SPT AND MST

A. Analysis for Deployment on a Grid Topology

For analytical tractability, we assume a deployment of sensor nodes on a grid topology of size $r \times 2r$ (referred to as grid deployment hereafter), where r source nodes at the leftmost column need to send information to the $sink$ located at the bottom right corner of the grid. Each sensor node can communicate to its one hop neighbors, i.e., 8 neighbors when ignoring boundary effects. We also assume $w_i = 1$ for all $e_i \in E$.

The routing constructed by SPT and MST are illustrated in Figure 4. For illustrative purpose, we choose the positions of the sink and source nodes so as to simplify our analysis, while still effectively demonstrating the tradeoffs between SPT and

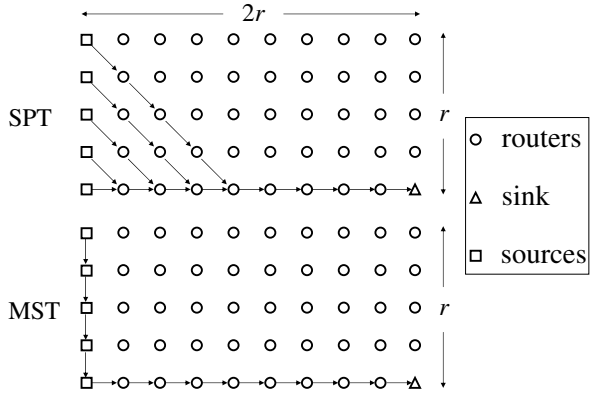


Fig. 4. SPT and MST routing schemes for a grid deployment

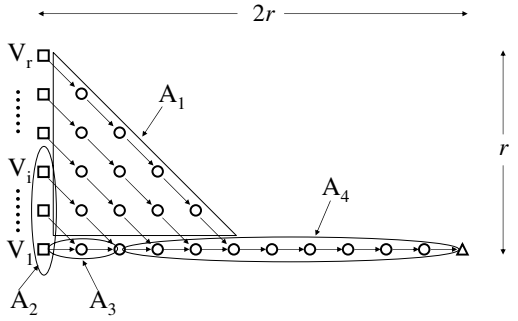


Fig. 5. Decomposition of ε_{SPT}

MST. Similar analysis could be performed for cases, e.g., where $sink$ is at the center of the network. Moreover, our analytical results match well with simulation results on general graphs.

Note that although to find an MST for a general graph is NP-Hard, the MST for the specific grid deployment in Figure 4 is trivial. From Corollary 1, the SPT is optimal when $\gamma \geq \gamma^* = (2r-1)L_1^2$. Hence, we are only interested in the performance of the SPT and MST for $\gamma \in [0, \gamma^*]$.

Let ε_{SPT} denote the energy cost for SPT and ε_{MST} for MST. Using Theorem 1, ε_{SPT} can be calculated as:

$$\varepsilon_{SPT} = \begin{cases} \frac{r(r-1)L_1}{2} + i(\frac{\gamma}{f^*} + f^*(2r-i)) + \sum_{j=1}^{i-1} jL_j \\ \quad + \sum_{j=r}^{2r-i-1} (\frac{\gamma}{f'} + jf'), \\ \quad \text{when } \gamma \in [(2r-i-1)L_i^2, (2r-i)L_{i-1}^2] \\ \quad \text{for some } 1 \leq i \leq r & \text{(7a)} \\ \frac{r(r-1)L_1}{2} + \frac{r\gamma}{L_r} + r^2 L_r + \sum_{j=1}^{r-1} jL_j, \\ \quad \text{when } \gamma \in [0, (r-1)L_r^2] & \text{(7b)} \end{cases}$$

where $f^* = \max\{L_i, \sqrt{\frac{\gamma}{2r-i}}\}$, and $f' = \min\{L_1, \frac{\gamma}{j}\}$. Note that for Eq. (7a), the upper bound of γ equals $(2r-1)$ when $i=1$, which is slightly larger than γ^*. However, this does not affect our further analysis.

We explain Eq. (7a) using Figure 5. The cost $\frac{r(r-1)L_1}{2}$ is for packet transmissions over edges in A_1. The term $i\frac{\gamma}{f^*}$ corresponds to the computation cost of the i source nodes circled in A_2, which have an optimal flow on their paths to the sink. The cost $i(2r-i)f^*$ is for transmitting flow f^* from the i source nodes in A_2 over their last $2r-i$ hops in A_4. The cost $\sum_{j=1}^{i-1} jL_j$ is for packet transmission from the i nodes in A_2 over edges in A_3. The term $\sum_{j=r}^{2r-i-1}(\frac{\gamma}{f'} + jf')$ is the compression cost for the $r-i$ source nodes not in A_2 plus the cost of packet transmission from these source nodes over edges in A_4. The $\min$ function in f' is due to constraint (5).

For Eq. (7b), maximum compression subject to constraint (4) is performed on all paths. The cost $\frac{r(r-1)L_1}{2}$ is again for packet transmissions over edges in triangle A_1. The cost $\frac{r\gamma}{L_r}$ accounts for the compression energy cost of r sources. The cost $r^2L_r + \sum_{j=1}^{r-1} jL_j$ is for packet transmissions over the edges in A_3 and A_4.

Let $q = 3r-1$. Let i^* be the smallest integer such that $(q-i^*)L_{i^*-1}^2 \geq \gamma^*$. We also calculate ε_{MST} as:

$$\varepsilon_{MST} = \begin{cases} 2\sqrt{\gamma} \sum_{j=2r-1}^{q-i-1} \sqrt{j} + \sum_{j=1}^{i-1} jL_j \\ \quad + i(\frac{\gamma}{f^*} + f^*(q-i)), \\ \quad \text{when } \gamma \in [(q-i-1)L_i^2, (q-i)L_{i-1}^2] \\ \quad \text{for some } i^* \leq i \leq r & \text{(8a)} \\ \frac{r\gamma}{L_r} + \sum_{j=1}^{r-1} jL_j + r(2r-1)L_r, \\ \quad \text{when } \gamma \in [0, 2rL_r^2] & \text{(8b)} \end{cases}$$

where $f^* = \max\{L_i, \sqrt{\frac{\gamma}{q-i}}\}$.

Also, the minimal cost of the TCDG problem is lower bounded by replacing constraint (4) with $\forall v \in R, \forall e \in p(v), f_e^v \geq L_{|R|}$ with $|R| = r$ in this particular case. In other words, we assume that distributed source coding among all source nodes is available at no extra cost. It can be verified that the

optimal routing for such an lower bound case forms exactly a SPT. Hence, the energy costs for the lower bound, ε_{LB} can be calculated as

$$\varepsilon_{LB} = \begin{cases} 2r\sqrt{\gamma(2r-1)}, \\ \qquad \text{for } (2r-1)L_r^2 \le \gamma \le \gamma^* \quad (9a) \\ \dfrac{r\gamma}{L_r} + r(2r-1)L_r, \\ \qquad \text{for } 0 \le \gamma \le (2r-1)L_r^2 \quad (9b) \end{cases}$$

Due to space limitation, we leave the detailed explanation of ε_{MST} and ε_{LB} in [26]. Based on the above results, we make the following observation.

Observation 1: For the grid deployment in Figure 4, we have the following performance bound regarding SPT and MST (refer to [26] for proof):

$$\lim_{\gamma\to\gamma^*} \frac{\varepsilon_{SPT}}{\varepsilon_{LB}} = O(1) \quad (10)$$

$$\lim_{\gamma\to 0} \frac{\varepsilon_{SPT}}{\varepsilon_{LB}} = O(\frac{r}{H_r}) \quad (11)$$

$$\lim_{\gamma\to\gamma^*} \frac{\varepsilon_{MST}}{\varepsilon_{LB}} = O(1) \quad (12)$$

$$\lim_{\gamma\to 0} \frac{\varepsilon_{MST}}{\varepsilon_{LB}} = O(1)\,. \quad (13)$$

where the critical point γ^* equals $(2r-1)L_1^2$.

The main lesson from Observation 1 is that, for this particular grid deployment, the energy cost of MST is a constant approximation of the optimal cost, regardless of H_i and γ. Although theoretically the performance of MST for general graphs is unbounded in the worst case, it is natural to conjecture that MST might also perform well on the average case. In Section VIII, we show that our simulation results on an approximated MST confirmed this conjecture.

We also notice that when γ approaches 0, the ratio of ε_{SPT} over ε_{LB} is $O(\frac{r}{H_r})$, which indicates that the performance of SPT improves when correlation among sources decreases. In the special case when $H_r = \Theta(r)$, SPT becomes the optimal structure.

We verify the above observation through numerical results in the next section.

B. Tradeoffs Between SPT and MST

Using the grid-based analysis developed in Section VI-A, we instantiate H_i using a set of practical joint entropy models from [27]. Specifically, we consider a stationary Gaussian random process with a scalar quantizer with uniform step size and infinite number of levels. Our entropy models are classified into 3 classes as follows, where d is the distance between source nodes:

E1: When the correlation coefficient is e^{-d^2}, H_i scales as $O(\log i)$ as $i \to \infty$ [27].

E2: When the correlation coefficient is e^{-d}, H_i scales as $O(\sqrt{i}\log i)$ as $i \to \infty$ [27].

E3: When all sources are independent to each other, H_i scales as $O(i)$ as $i \to \infty$.

We set $H_1 = \rho$, where ρ is the data entropy rate. According to the compression ratio for 10 test images using CCITT G4 lossless compression tools [28], we set $\rho = 0.1$ (We examine a wider range of ρ via simulations). For $i > 1$, we set H_i to $\rho \log i$ for E1, $\rho\sqrt{i}\log i$ for E2, and $i\rho$ for E3[1]. Intuitively, the correlation among sources is highest in the case of E1, and lowest in the case of E3. We shall see that this difference does affect the tradeoffs between SPT and MST according to Observation 1.

1) Tradeoffs for Entropy Model E1: In Figure 6, we plot ε_{SPT}, ε_{MST}, and ε_{LB} for E1 with $r = 40$ and γ varied between 0 and 0.4 (note that in this case $\gamma^* = (2r-1)\rho^2 \approx 0.8$).

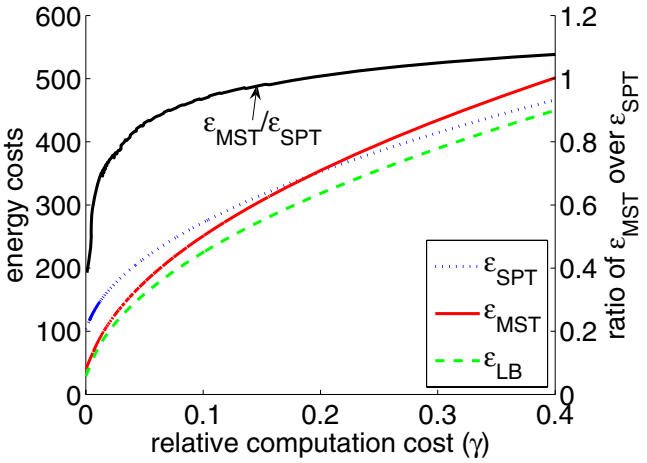


Fig. 6. Performance of SPT and MST for grid deployment with entropy model E1

From Figure 6, we can clearly observe the tradeoffs between SPT and MST with respect to variations in γ. When γ is large, SPT outperforms MST with ε_{SPT} approaching to ε_{LB}. This is because large computation cost discourages data compression, hence shortest paths from source nodes to sink are preferred for saving communication costs.

[1]While H_i is asymptotically bounded by these functions, we set L_i to L_{i+1} if $L_i < L_{i+1}$ for small i's in our numerical calculation to sustain our assumption that $L_i \ge L_{i+1}$.

However, the performance of MST is also quite satisfactory when $\gamma = 0.4$, with no more than 10% increase over SPT. Although not shown in the Figure, ε_{MST} is also within 15% off ε_{SPT} when $\gamma = \gamma^* = 0.8$.

On the other hand, when γ approaches to zero, MST provides up to 60% energy reduction compared to SPT. This is because when the computation costs is low, compressing data from multiple sources before routing to the sink provides higher gains by reducing the flow on the tree. In the special case of $\gamma = 0$, our problem becomes similar to the scenario studied in [2], where tradeoffs between MST and SPT exist due to variations in spatial correlation among source nodes —- MST outperforms SPT when the correlation is high and SPT outperforms MST when the correlation is low. In our case, the spatial correlation captured by $H_i = O(\log i)$ determines that MST outperforms SPT, which is in keeping with the results in [2].

2) Tradeoffs for Entropy Model E2: The tradeoffs between SPT and MST is more complicated in the case of E2. Based on Observation 1, the asymptotic ratio of $\frac{\varepsilon_{SPT}}{\varepsilon_{LB}}$ when $\gamma \to 0$ approaches $\frac{r}{H_r}$, which is $O(\frac{\sqrt{r}}{\log r})$ in the case of E2. Compared with the $O(\frac{r}{\log r})$ ratio in case of E1, there is an improvement of factor $O(\sqrt{r})$, as shown by the following numerical results. Moreover, $\frac{r}{H_r}$ increases with r, which is also verified by our results.

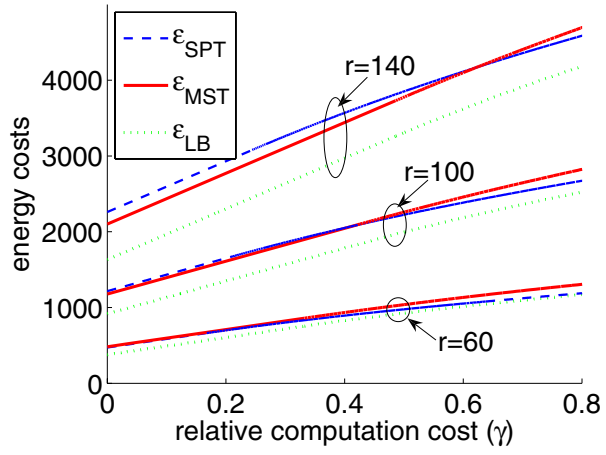


Fig. 7. Performance of SPT and MST for grid deployment with entropy model E2

In Figure 7, we illustrate the performance of SPT and MST with respect to variations in both γ and r. In this figure, we vary r from 60 to 140 in increments of 40, and γ from 0 to 0.8. It can be observed that when $r = 60$, both the performance of SPT and MST is very close to ε_{LB}. However, as r increases, we can see the degradation of the performance of SPT. When $r = 140$, the tradeoffs between SPT and MST is apparent and is similar to that in Figure 6.

3) SPT is optimal for Entropy Model E3: It is easy to understand that in the case of E3, SPT is the optimal solution. This conclusion is true for not only the analyzed grid deployment, but general graphs.

Similar performance tradeoffs are observed in scenarios with r varied from 5 to 150. In short, the above analysis and numerical results for a particular grid deployment clearly demonstrate (1) the tradeoffs between SPT and MST with respect to variations in H_i and γ, and (2) that MST delivers constantly bounded performance compared to the optimal. Another important insight is that when γ varies, the optimal routing scheme shall explore the tradeoffs between SPT and MST. Along this direction, we exploit in next section a hierarchically clustered tree structure to solve TCDG.

VII. A Randomized $\log^2 n$ Approximation

For theoretical completeness, we present a randomized algorithm that achieves poly-logarithmic approximation for the TCDG problem for general graphs. The key idea is to approximate the graph G with a set of k-hierarchically well-separated trees (k-HST's) [21] such that the routing selected according to a randomly chosen k-HST is expected to have a cost at most $O(\log^2 n)$ times the optimal. Due to space limitation, we briefly present our results in this section, details of the algorithm and the proof can be found in [26].

Let $G =< V, E, w >$ denote a weighted connected graph and $d_G(u, v)$ denote the distance between $u, v \in V$. Given any G and a constant k, Bartal defines a set of k-HST's over G, denoted as $\mathcal{S}$, such that:

Theorem 2: [21] There exists a probability distribution over $\mathcal{S}$ such that for every $M \in \mathcal{S}$ and every $u, v \in V$, $E(d_M(u, v)) \leq \alpha \cdot d_G(u, v)$, where $\alpha = O(\log^2 n)$.

Using the above results, we can show:

Theorem 3: [26] Given a TCDG problem on graph G with optimal cost equal to C, there is a feasible solution on the set of k-HST's over G with

the expected cost (over the distribution on the k-HST's) to be at most αC.

We know that the optimal solution to a TCDG problem over a tree M is simply the composition of routes from each node in R to $sink$ on M. Thus, by randomly choosing a k-HST, M from $\mathcal{S}$ and then map the routes from each $v \in R$ to $sink$ in M to a path in G, we get a routing scheme over G with an expected cost within $O(\log^2 n)$ times the optimal.

However, the construction of k-HST requires the knowledge of the entire network topology [21]. It is not clear how to construct a k-HST using a distributed algorithm that is suitable for sensor networks. On the contrary, SPT can be constructed using distributed bellman ford algorithm, while distributed approximated MST can be constructed based on shortest path [29]. Hence, these simple tree construction methods gain advantages if their average performance for general graphs is reasonably good compared with that of k-HST, which will be studied in the next section.

VIII. Simulation Results

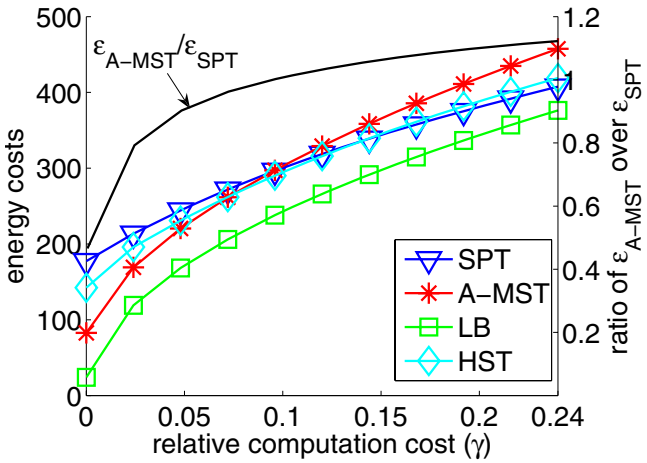
A. Simulation Setup

A sensor network was generated by randomly scattering n sensors in a unit square. The sink node was always fixed at the left-bottom corner of the square, while the source nodes were randomly selected from the rest $n-1$ nodes. The communication range of the radios was set to μ. We set the weight on each edge to be cd, where d was the distance between the two incident nodes, and c was a scaling factor. To correspond to the numerical results in Section VI-B, we set $c = 80$ so that the critical point γ^* in our simulations was around $80\rho^2$. Also, by using d as the basis of edge weight, we explore the case sitting in between the fixed communication cost (regardless to d) model and the d^2 path loss model, and hopefully our results imply the applicability of the proposed techniques to both models.

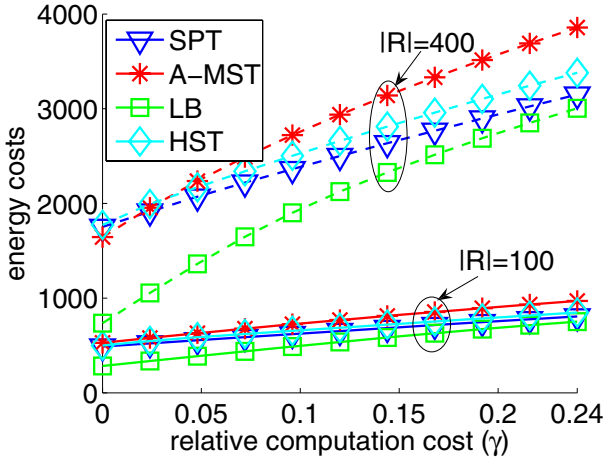
The performance of 3 tree construction methods, SPT, MST, and k-HST (or simply HST hereafter), was studied by simulations. While SPT and HST could be constructed based on polynomial time algorithms, the construction of MST was NP-Hard for general graphs. We used the Greedy Incremental Tree (GIT) algorithm [1] that gave a 2-approximation MST [29], with A-MST denoting the resulting approximated MST. Moreover, the lower bound of the TCDG problem was obtained using the relaxation method described in Section VI.

All the data shown in this section is averaged over at least 200 instances such that they have a 95% confidence interval with a $\leq 2\%$ precision. For each instance, the sensor field was randomly generated using the above procedure.

B. Results



(a) Entropy model E1 with 50 source nodes



(b) Entropy model E2 with 200 and 400 source nodes

Fig. 8. Main simulation results ($n = 600$, $\mu = 0.2$)

Main results: For the results shown in Figure 8, we fixed $n = 600$, $\mu = 0.2$, and $\rho = 0.1$, while varying γ within $[0, 0.24]$ so that we can focus on the tradeoffs between A-MST and SPT. We observed that the simulation results for general graphs confirmed our analytical results in Section VI. In the case of entropy model E1, the performance of SPT approached the lower bound when γ increased,

while A-MST outperformed SPT when γ tended to zero. As expected, HST performed in between SPT and A-MST throughout the variations in γ. More importantly, A-MST demonstrated acceptable performance throughout the variations in γ. The curve of $\varepsilon_{MST}/\varepsilon_{SPT}$ clearly showed that A-MST offers 50% energy savings over SPT when $\gamma = 0$, and $\leq 15\%$ increase over SPT at high γ.

Compared to Figure 6, we observed a lower threshold value of γ (around 0.1) for the crossover of SPT and MST in our simulations. This was mainly because even with the scaling factor $c = 80$, the average distance from the source nodes to $sink$ was around $0.77c \approx 60$ [30], which was smaller than the average distance, 80, in Figure 4.

In the case of entropy model E2, we observed the expected performance improvement of SPT over the case of E1. For the sub-case of 200 source nodes, the performance of SPT and A-MST was very close to the lower bound. When the number of sources was increased to 400, the tradeoffs between SPT and A-MST became observable. This confirmed the analytical results for the grid deployment in Figure 7, which indicated that the choice between A-MST and SPT depended on the exact entropy model as well as the number of sources.

Above simulation results indicate that when the entropy model and relative computation cost γ is known, either SPT or A-MST can be selected accordingly as a practical routing scheme. When γ is unknown or demonstrates high spatio-temporal variation, HST can be used to provide an approximation with theoretically guaranteed performance bound. Nevertheless, in practice, the simple A-MST performs well on the average, with only slight degradation compared to SPT when γ is large or the correlation among sources is low.

We have also conducted simulations using other values of n with similar performance trend observed. Due to space limitation, these results are omitted in this paper. Also, we focus on the results for entropy model E1 in the following presentation.

Impact of the data entropy rate, ρ: Based on the study of CCITT G4 lossless compression over 10 testing images, ρ vary from 0.02 to 0.27. Thus, we set $n = 600$, $\mu = 0.2$, $|R| = 50$, $\gamma = 0.1$ and 0.4, while varying ρ between $[0.02, 0.2]$. We illustrate the results in Figure 9.

We observed that for all cases of γ, the energy costs of SPT, A-MST, and HST increased with ρ.

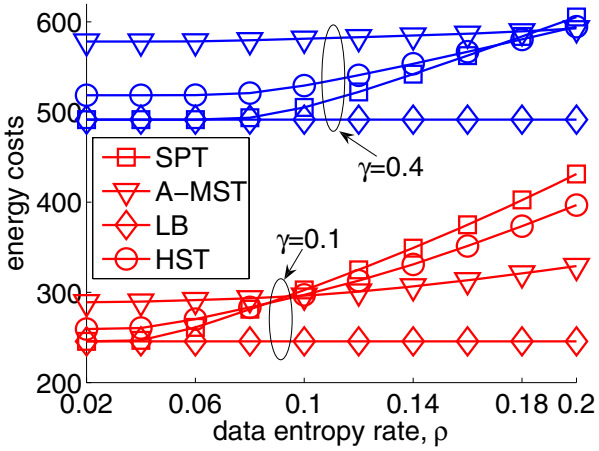


Fig. 9. Impact of the data entropy rate ρ ($n = 600$, $|R| = 50$, $\mu = 0.2$)

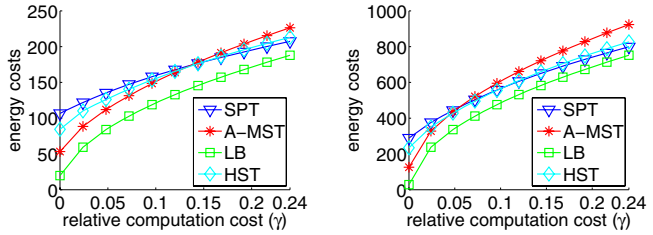
This was because a larger ρ meant larger lower bound of data flow on all edges, thus increasing the data volume for communication. However, the cost of LB remained the same throughout variations in ρ. This was because LB assumed a perfect distributed source coding, which led to a smaller lower bound of data flow and mitigated the impact of ρ.

Moreover, we observed tradeoffs between SPT and A-MST when ρ increased. When ρ is smaller, SPT outperformed A-MST since a sufficient compression ratio was achieved without using join compression, which favored a SPT routing structure. When ρ increased, A-MST exhibited better performance due to the need of joint compression to further reduce the communication cost. Also, joint compression became less necessary when γ increased, since the higher computation cost discouraged high compression ratio. This was reflected by the fact that the crossover point of SPT and A-MST moved right with γ. Moreover, HST behaved in between SPT and A-MST as expected.

Further, the performance of A-MST was again acceptable throughout variations in γ and ρ.

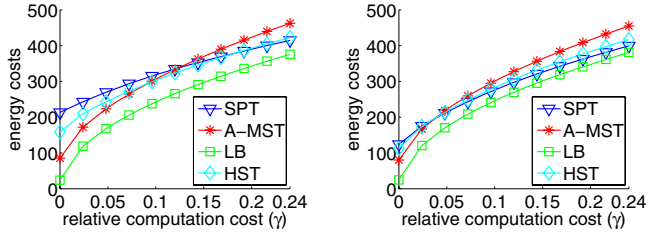
Impact of the number of source nodes $|R|$: For the results shown in Figure 10, we fixed $n = 600$, $\mu = 0.2$, while setting $|R|$ to 25 or 100 and varying γ within $[0, 0.24]$. It was understandable that the energy costs of all tree structures increased with $|R|$. Nevertheless, the tradeoffs between SPT, A-MST, and HST still held for different values of $|R|$.

Impact of the communication range μ: For the results shown in Figure 11, we fixed $n = 600$, $|R| = 50$, while setting μ to 0.1 or 0.3 and varying



(a) $|R| = 25$ (b) $|R| = 100$

Fig. 10. Impact of the number of source nodes $|R|$ ($n = 600$, $\mu = 0.2$)



(a) $\mu = 0.1$ (b) $\mu = 0.3$

Fig. 11. Impact of the communication range μ ($n = 600$, $|R| = 50$)

γ within $[0, 0.24]$. The tradeoffs between SPT, A-MST, and HST were still apparent for different μ. We observed that increasing μ had little impact on the performance of A-MST and LB. However, since SPT did not favor data aggregation, ε_{SPT} increased with μ dramatically when γ was small.

IX. Concluding Remarks

We have presented results regarding data gathering for computation-intensive applications in sensor networks, where computation energy for data compression needs to be carefully traded against the communication energy in scenarios such as streaming applications and video surveillance. Our key contributions include (1) a suitable energy model for tunable compression and a flow-based model to facilitate the tuning of compression over a data gathering tree, (2) techniques for determining the optimal flow for a given tree structure, based on which, the performance of SPT and MST in a grid deployment has been analytically studied, (3) extensive simulation results further revealing the tradeoffs between SPT and A-MST in general graphs, which satisfactorily confirm our analysis for the grid deployment, and (4)a randomized algorithm with poly-logarithmic performance bound.

The lessons from our study are (1) with the knowledge of γ and H_i, either SPT or A-MST can be appropriately chosen, and (2) when such information is unknown or if it shows large spatio-temporal variations, A-MST provides acceptable average performance for general graphs. Due to its simplicity in distributed implementation, A-MST is preferred as a practical routing scheme in this case.

A future research direction is to identify suitable techniques to realize tunable compression for data gathering, e.g., adaptive multimedia processing. It is also crucial to develop accurate energy models for these techniques to apply the algorithms described in this paper.

References

[1] B. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks," in *International Workshop on Distributed Event-Based Systems*, July 2002.

[2] S. Pattem, B. Krishnamachari, and R. Govindan, "The impact of spatial correlation on routing with compression in wireless sensor networks," in *ACM/IEEE IPSN*, Apr. 2004.

[3] C. M. Sadler and M. Martonosi, "Data compression algorithms for energy-constrained devices in delay tolerant networks," in *ACM SenSys*, Nov. 2006.

[4] J. Acimovic, B. Beferull-Lozano, and R. Cristescu, "Adaptive distributed algorithms for power-efficient data gathering in sensor networks," in *IEEE International Symposium on Wireless Sensor Networks*, June 2005.

[5] K. Barr and K. Asanović, "Energy aware lossless data compression," in *ACM MobiSys*, May 2003.

[6] T. Bell, M. Powell, J. Horlor, and R. Arnold, "The Canterbury Corpus." [Online]. Available: http://www.cosc.canterbury.ac.nz

[7] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. on Information Theory*, vol. 19, no. 4, pp. 471–480, 1973.

[8] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, "On network correlated data gathering," in *IEEE InfoCom*, Mar. 2004.

[9] A. Kashyap, L. A. Lastras-Montano, C. Xia, and Z. Liu, "Distributed source coding in dense sensor networks," in *Data Compression Conference*, Mar. 2005, pp. 13–22.

[10] A. Goel and D. Estrin, "Simultaneous optimization for concave costs: Single sink aggregation or single source buy-at-bulk," in *ACM-SIAM SODA*, Jan. 2003.

[11] R. Kanna and S. S. Iyengar, "Game-theoretic models for reliable path-length and energy-constrained routing with data aggregation in wireless sensor networks," *IEEE JSAC*, vol. 22, no. 6, pp. 1141–1150, Aug. 2004.

[12] B. Awerbuch and Y. Azar, "Buy-at-bulk network design," in *FOCS*, Oct. 1997, pp. 542–547.

[13] W. Choi and S. K. Das, "A framework for energy-saving data gathering using two-phase clustering in wireless sensor networks," in *ACM MobiQuitous*, Aug. 2004, pp. 203–212.

[14] B. Hong and V. K. Prasanna, "Optimizing a class of in-network processing applications in networked sensor systems," in *IEEE MASS*, Oct. 2004.

[15] W. Choi and S. K. Das, "A novel framework for energy-conserving data gathering in wireless sensor networks," in *IEEE InfoCom*, Mar. 2005.

[16] P. Leone, S. E. Nikoletseas, and J. Rolim, "An adaptive blind algorithm for energy balanced data propagation in wireless sensors networks," in *IEEE DCOSS*, June 2005.

[17] A. Jarry, P. Leone, O. Powell, and J. Rolim, "An optimal data propagation algorithm for maximizing the lifespan of sensor networks," in *IEEE DCOSS*, June 2006.

[18] H. Luo, J. Luo, Y. Liu, and S. K. Das, "Adaptive data fusion for energy effi cient routing in wireless sensor networks," *IEEE Trans. on Computers*, vol. 55, no. 10, pp. 1286–1299, Oct. 2006.

[19] B. D. Nobel, M. satyanarayanan, D. Narayanan, J. E. Tilton, J. Flinn, and K. R. Walker, "Agile application-aware adaptation for mobility," in *ACM SOSP*, Oct. 1997.

[20] M. D. Corner, B. D. Nobel, and K. M. Wasserman, "Fugue: Time scales of adaptation in mobile video," in *the SPIE Multimedia Computing and Networking Conference*, Jan. 2001, pp. 75–87.

[21] Y. Bartal, "Probabilistic approximations of metric spaces and its algorithmic applications," in *FOCS*, 1997.

[22] J. Fakcheroenphol, S. Rao, and K. Talwar, "A tight bound on approximating arbitrary metrics by tree metrics," in *STOC*, June 2003.

[23] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves, "Energy-effi cient, collision-free medium access control for wireless sensor networks," in *ACM SenSys*, Nov. 2003.

[24] Y. Yu, B. Krishnamachari, and V. K. Prasanna, "Energy-latency tradeoffs for data gathering in wireless sensor networks," in *IEEE InfoCom*, Mar. 2004.

[25] V. Raghunathan, C. Schurgers, S. Park, and M. B. Srivastava, "Energy-aware wireless microsensor networks," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 40–50, March 2002.

[26] Y. Yu, B. Krishnamachari, and V. K. Prasanna, "Energy-effi cient data gathering with tunable compression in wireless sensor networks," University of Southern California, Tech. Rep. CENG-2004-15, 2004. [Online]. Available: http://halcyon.usc.edu/~yangyu/data/TR_CENG200415.pdf

[27] D. Marco, E. J. Duarte-Melo, M. Liu, and D. L. Neuhoff, "On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of its data," in *ACM/IEEE IPSN*, Apr. 2003, pp. 1–16.

[28] A. Knoll, "Compression of bi-level images: compressor performance report," in *INFORUM 2000 Conference*, May 2000, pp. 23–25. [Online]. Available: http://www.inforum.cz/inforum2000/prednasky/kompresebitona.html

[29] H. Takahashi and A. Matsuyama, "An approximate solution for the steiner problem in graphs," *Mach. Japonica*, vol. 24, no. 6, pp. 573–577, 1980.

[30] E. W. Weisstein, "Square point picking." [Online]. Available: http://mathworld.wolfram.com/SquarePointPicking.html

APPENDIX I
PROOFS

Proof of Lemma 1: Otherwise, decreasing f_i to β_i does not change the cost for compression over $p(v)$, since the compression energy is determined only by the flow on the last link in $p(v)$. However, this reduces the cost of communication over e_i, contradicting the optimality of the flow $\vec{f}$. ■

Proof of Theorem 1: If $\gamma \geq W_1$, it means that any compression is more expensive than transmitting the original data along the path. Hence the optimal solution is to simply transmit the data packet without any compression. Otherwise, the proof is as follows.

First, since both W_i and β_i decreases with i, i.e., $W_{i+1} \leq W_i$ and $\beta_i \leq \beta_{i-1}$, the condition for γ is valid. Also, since $W_k\beta_{k-1}^2 = 0$ and $W_1\beta_0^2 = W_1$, the range of γ is within $[0, W_1]$.

Suppose that $\gamma \in [W_{i+1}\beta_i^2, W_i\beta_{i-1}^2]$ for some $1 \leq i \leq k-1$. Suppose that $\vec{x} = \{x_1, \ldots, x_{k-1}\}$ is the vector of the optimal flow with cost ϵ_x. Let f^* denote $\max\{\beta_i, \sqrt{\frac{\gamma}{W_i}}\}$. Let $\vec{f}$ denote the flow constructed by setting $f_j = x_j$ for $1 \leq j < i$ and $f_j = f^*$ for $i \leq j \leq k-1$. Let ϵ_f denote the cost of $\vec{f}$. We have

$$\begin{aligned}
\epsilon_x - \epsilon_f &= (\frac{\gamma}{x_{k-1}} + \sum_{j=1}^{k-1} x_j w_j) \\
&\quad -(\frac{\gamma}{f^*} + \sum_{j=1}^{i-1} x_j w_j + f^* \sum_{j=i}^{k-1} w_j) \\
&= \frac{\gamma}{x_{k-1}} + \sum_{j=i}^{k-1} x_j w_j - (\frac{\gamma}{f^*} + f^* W_i) \\
&\geq (\frac{\gamma}{x_{k-1}} + x_{k-1} W_i) - (\frac{\gamma}{f^*} + f^* W_i) \quad (14)
\end{aligned}$$

We define an optimization problem, $P(y)$, as to:

$$\begin{aligned}
\min \quad & P(y) = \frac{\gamma}{y} + yW_i \\
\text{subject to} \quad & y \geq \beta_i \ .
\end{aligned}$$

Hence, Eq. (14) is actually $P(x_{k-1}) - P(f^*)$. It is easy to verify that $P(y)$ is a convex function. We consider two cases for $\gamma \in [W_{i+1}\beta_i^2, W_i\beta_{i-1}^2]$.

Case (i): When $\gamma \in [W_i\beta_i^2, W_i\beta_{i-1}^2]$, we have $\sqrt{\frac{\gamma}{W_i}} \geq \beta_i$, hence implying $f^* = \sqrt{\frac{\gamma}{W_i}}$. For the above optimization problem $P(y)$, we have $P'(\beta_i) = -\frac{\gamma}{\beta_i^2} + W_i \leq 0$ and $P'(\beta_{i-1}) = -\frac{\gamma}{\beta_{i-1}^2} + W_i \geq 0$, where $P'(y)$ is the first derivative of $P(y)$. Therefore, the optimal y that leads to $P'(y) = 0$ lies within $[\beta_i, \beta_{i-1}]$. By solving $P'(y) = 0$, we know that the optimal value of y actually equals f^*. Thus, $\epsilon_x - \epsilon_f = P(x_{k-1}) - P(f^*) \geq 0$, implying that $\vec{f}$ is optimal. From Lemma 1 and the fact $f^* \in [\beta_i, \beta_{i-1}]$, we have $f_j = \beta_j$ for $1 \leq j < i$.

Case (ii): When $\gamma \in [W_{i+1}\beta_i^2, W_i\beta_i^2]$, we have $\sqrt{\frac{\gamma}{W_i}} \leq \beta_i$, implying $f^* = \beta_i$. Also, we have $P'(\beta_i) = -\frac{\gamma}{\beta_i^2} + W_i \geq 0$. This means that $P(y)$ is an increasing function when $y \geq \beta_i$. Thus, $P(y)$ is minimized when $y = \beta_i$. Again in this case, we have $\epsilon_x - \epsilon_f \geq 0$, implying the optimality of $\vec{f}$.

Finally, we can combine the above two cases using a $\max$ function for f^*. ■