A Distributed Framework for Correlated Data Gathering in Sensor Networks

Kevin Yuen, *Student Member, IEEE*, Ben Liang, *Senior Member, IEEE*, and Baochun Li, *Senior Member, IEEE*

***Abstract*—We consider the problem of correlated data gathering in sensor networks with multiple sink nodes. The problem has two objectives. First, we would like to find a rate allocation on the correlated sensor nodes such that the data gathered by the sink nodes can reproduce the field of observation. Second, we would like to find a transmission structure on the network graph such that the total transmission energy consumed by the network is minimized. The existing solutions to this problem are impractical for deployment because they have not considered all of the following factors: 1) distributed implementation; 2) capacity and interference associated with the shared medium; and 3) realistic data correlation model. In this paper, we propose a new distributed framework to achieve minimum energy data gathering while considering these three factors. Based on a localized version of Slepian–Wolf coding, the problem is modeled as an optimization formulation with a distributed solution. The formulation is first relaxed with Lagrangian dualization and then solved with the subgradient algorithm. The algorithm is amenable to fully distributed implementations, which corresponds to the decentralized nature of sensor networks. To evaluate its effectiveness, we have conducted extensive simulations under a variety of network environments. The results indicate that the algorithm supports asynchronous network settings, sink mobility, and duty schedules.**

***Index Terms*—Correlated data gathering, data aggregation, distributed algorithm, mathematical optimization, wireless sensor networks.**

I. Introduction

RECENT technological advances have enabled the production of low-cost sensor nodes. These sensor nodes are small in size and equipped with limited sensing, processing, and transmission capabilities. They can be deployed in large numbers to construct a sensor network with the ability of distributed wireless sensing. The collaborative effort of these sensor nodes can achieve significant improvement over traditional sensors due to their improved accuracy and ease of deployment. In practice, the sensor nodes are densely deployed in an ad hoc fashion over the area of interest. After their deployment, the sensor nodes collect data from their surroundings, encode the data, and transmit them to the sink nodes via wireless channels. In addition to collecting data, the intermediate sensor nodes can be used as relays for other sensors distant from the sink nodes. Sink nodes are specialized nodes that are responsible for gathering collected data and serve as gateways between the sensor network and the wired or wireless backbone network.

Many applications for sensor networks such as target tracking [1] and habitat monitoring [2] involve monitoring a remote or hostile field. The sensor nodes are assumed to be inaccessible after deployment for such applications, and thus, their batteries are irreplaceable. Moreover, due to the small size of sensor nodes, they carry limited battery power. Thus, energy is a scarce resource that must be conserved to the extent possible in sensor networks.

A. Problem Description and Design Goals

In this context, the first objective of the correlated data-gathering problem is to find a rate allocation on the sensor nodes such that the aggregated data collected by the sink nodes can be decoded to reproduce the field of observation. The rate allocation assigns each sensor node an encoding rate, which is equivalent to its data transmission rate. If the data collected by the sensor nodes are statistically independent, then the rate allocation can be trivially determined—each sensor node can transmit at its data collection rate. However, the sensor nodes are densely deployed in sensor networks. Nearby sensor nodes have overlapping sensing ranges, and their collected data are either redundant or correlated. This data correlation can be exploited to reduce the amount of data transmitted in the network, which results in energy savings. To achieve minimum energy data gathering, the optimal rate allocation should minimize the encoding rates while ensuring the rates are sufficient to represent all the independent data generated by the sensor nodes.

The second objective of the correlated data-gathering problem is to find a transmission structure on the network graph such that the total energy consumed in transporting the collected data from the sensor nodes to the sink nodes is minimized. If the network has unconstrained bandwidth capacity, then this objective can be simply achieved—each sensor node can transmit its collected data via the minimum energy path. However, in any practical network, there are capacity limitations on the transmission medium and interference among the competing signals. In wireline networks, there is a time-dependent contention, where two signals compete with each other if they both arrived at the receiver at the same time. The effect of

Manuscript received April 20, 2006; revised October 11, 2006 and December 21, 2006. This work was supported in part by a grant from the Natural Sciences and Engineering Research Council of Canada and by Bell Canada through its Bell University Laboratories R&D program. The review of this paper was coordinated by Dr. S. Vishwanath.

K. Yuen was with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada. He is now with IBM Toronto Laboratories, Markham, ON L3R 9Z7, Canada (e-mail: yuenke@eecg.toronto.edu).

B. Liang and B. Li are with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada (e-mail: liang@comm.toronto.edu; bli@eecg.toronto.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2007.905243

interference in wireline networks is well studied, but they are not applicable in the context of sensor networks. As a variation of wireless ad hoc networks, sensor networks have the unique characteristic of location-dependent contention in addition to time-dependent contention. Signals will compete with each other if multiple sensor nodes in the nearby vicinity simultaneously access the wireless shared medium. To derive feasible solutions, the capacity and interference associated with the shared medium must be considered when constructing the optimal transmission structure.

It is shown in [3] that if the bandwidth capacity of the network is unconstrained, the two problem objectives can be independently achieved in two steps. First, according to the correlation model, the optimal rate allocation can be determined. Then the optimal transmission structure can be constructed by combining the minimum energy paths of the sensor nodes. However, when capacity constraints exist, the problem becomes complicated because the two objectives are dependent. Given a transmission structure, if the rate allocation is modified, then some of the links selected by the transmission structure may become congested due to the increased traffic flows. To alleviate this congestion, the transmission structure has to be adjusted. On the other hand, there are different coding schemes that exploit data correlation in the literature. They can be generally divided into two categories, i.e., distributed source coding and joint entropy coding with explicit communication. In practice, the coding schemes from both categories determine the rate allocation based on a given transmission structure. Consequently, the decision on the rate allocation affects the decision on the transmission structure, and vice versa. One of the highlights of this paper is to take this dependence into account and design an algorithm that jointly optimizes the rate allocation and the transmission structure while satisfying the capacity constraints.

In addition to the problem objectives, we have included several design goals when constructing the framework. The ultimate purpose of this paper is to create a solution to the correlated data-gathering problem that is practical for deployment. More importantly, the framework should be compatible, which allows other energy-saving mechanisms to be built on top of the framework to further extend the lifetime of data-gathering sensor networks.

- *Multisink support:* To facilitate efficient data gathering, it is envisioned that future sensor networks will consist of multiple sink nodes. By providing multisink support, the framework becomes feasible for deployment in large-scale sensor networks.
- *Distributed solutions:* With centralized solutions, the participating nodes need to repeatedly transmit detailed status information across the network to a central computation node. Although centralized approaches can generate results closer to the global optimum, they are generally not feasible in energy-constrained sensor networks.
- *Asynchronous network settings:* Due to the *ad hoc* infrastructure of sensor networks, it is expensive in terms of communication overhead to synchronize the nodes. If the framework is applicable in asynchronous network settings, it can avoid the scaling limitation posed by synchronous solutions.
- *Sink mobility:* Because of its multihop nature, the appearance of energy holes in static sensor networks seems unavoidable. Sensor nodes positioned around the sink nodes deplete their energy faster because they are frequently acting as relays. A natural way to counter energy holes is to introduce sink mobility, where sink nodes move within the network as they gather data from the sensor nodes.
- *Duty schedules:* To achieve maximum network lifetime, load balancing among the sensor nodes must be enforced. This can be accomplished with the introduction of duty schedules, where sensor nodes switch their operating status (on/off) to control and match their energy consumption rates. However, duty schedules give rise to network dynamics, since the sensor nodes may join and leave the network at run time.

B. Main Contributions

Data gathering with correlated sources in sensor networks and resource allocation with capacity constraints in wireless links were separately studied in the past literature. The main contribution of this paper is to propose a solution to the data-gathering problem that simultaneously considers both topics. The proposed solution copes with the dependence that exists between the two problem objectives as it jointly optimizes the rate allocation and the transmission structure. Furthermore, the optimization formulation is specifically designed to have a distributed solution.

Since the aim of the problem is to minimize energy consumption, it is a natural decision to employ optimization techniques. We model the problem as an exponential-constraint linear optimization formulation. According to the protocol model [4] of packet transmission in wireless networks, the formulation considers the capacity limitation of the network and the effect of location-dependent contention. As a result, our solution is guaranteed to be supported by the wireless shared medium. Since the exponential-constraint linear formulations are generally difficult to solve, we relax the formulation to become linear by adapting a localized version of Slepian–Wolf coding. Based on Lagrangian dualization, we utilize a price-based resource allocation strategy and solve the formulation with the subgradient algorithm. The price signals are communicated among the sensor nodes to reflect the congestion status of the network. The subgradient algorithm is amenable to distributed implementations, which makes it feasible for practical deployment. Moreover, we conduct extensive simulations to validate that our solution supports asynchronous network settings, sink mobility, and duty schedules. To the best of our knowledge, no previous works have addressed the correlated data-gathering problem considering all of the factors above.

C. Paper Organization

The remainder of this paper is organized as follows: In Section II, we present the exponential-constraint linear optimization formulation for the correlated data-gathering problem.

In Section III, we describe the localized Slepian–Wolf coding scheme and present the linear formulation. In Section IV, we construct an efficient distributed algorithm to solve the formulation with Lagrangian dualization and the subgradient algorithm. In Section V, we discuss implementation issues related to the algorithm. Numerical results from simulations are presented in Section VI. Finally, we discuss related work in Section VII and conclude this paper in Section VIII.

II. Problem Formulation

A. Network Model and Assumptions

The wireless sensor network is modeled as a directed graph $G = (V, E)$, where V is the set of nodes, and E is the set of directed wireless links. S_N denotes the set of sensor nodes, and S_K denotes the set of sink nodes. Then, $V = S_N \cup S_K$. The rate allocation assigns each sensor node $i \in S_N$ with R_i, which refers to a nonnegative data transmission rate. All the sensor nodes have a fixed transmission range r_{tx}. Let d_{ij} denote the distance between node i and node j. A directed link $(i, j) \in E$ exists if $d_{ij} \leq r_{\mathrm{tx}}$. Each link is associated with a weight $e_{ij} = d_{ij}^2$, which refers to the energy consumed per unit flow on link (i, j). All the links are assumed to be symmetrical, where $e_{ij} = e_{ji}$. Moreover, f_{ij} represents the flow rate of link (i, j). Here, the rate vector $[R_i]_{\forall i \in S_N}$ and the flow vector $[f_{ij}]_{\forall (i,j) \in E}$ are the variables that can be adjusted to minimize the optimization objective.

There are various models for sensor networks. In this paper, we mainly focus on a sensor network environment where we have the following.

- A spatial data correlation model [5] is assumed, where the sensor nodes can achieve various amounts of data aggregation based on their distance of separation. In contrast, a perfect data correlation model is assumed in [6]–[8], where intermediate sensor nodes can aggregate any number of incoming packets into a single packet. Although the perfect data correlation model can represent higher energy savings, it is generally not practical in most application scenarios.
- The transmission power is automatically managed by the sensor nodes. During a transmission, the sensor nodes have the ability to adjust their transmission power depending on the distance of transmission. Consequently, the energy consumed per unit flow on a link is a function of its distance. Moreover, the transmission power is assumed to be allocated in a specific way such that the capacity of the wireless shared medium is constant across the entire network. Power allocation is out of the scope of this paper and is left as a future research direction.
- Depending on the application of the sensor network, its data delivery model can be continuous, event driven, or query driven [9]. We have assumed a continuous data delivery model for illustration, where the sensor nodes periodically sense their surroundings and always have data to transmit in each round of communication. In the event-driven or query-driven delivery model, the data are transmitted to the sink nodes when the sensor node detects an event or receives a query. We emphasize that since our proposed solution supports duty schedules, it can be easily extended to accommodate these data delivery models.
- The objective of the correlated data-gathering problem is to minimize the total transmission energy consumed by the network. While this objective does not guarantee to maximize the lifetime of each individual sensor node, it can achieve a better energy efficiency, thus extending the network lifetime. In this paper, the sensor networks are assumed to have a high density of sensor nodes. This implies that the failure of an individual sensor node (possibility due to energy exhaustion) will not have a critical impact on the coverage or connectivity of the network. Moreover, our solution can be combined with load-balancing mechanisms to achieve fairness in energy consumption.

B. Data Correlation Model

Since the sensor nodes are usually continuous and not discrete sources, the theoretical tool to analyze the problem is the rate distortion theory [10], [11]. Let S be a vector of n samples of the measured random field returned by n sensor nodes. Let $\hat{S}$ be a representation of S and $d(S, \hat{S})$ be a distortion measure. With the mean square error as the distortion measure $d(S, \hat{S}) = \|S - \hat{S}\|^2$ and with the constraint

$$E\left(\|S - \hat{S}\|^2\right) < D \tag{1}$$

a Gaussian source is the worst case and needs the most bits to be represented compared with other sources [10]. For the purpose of illustration, we let S be a spatially correlated random Gaussian vector $\sim N(\mu, \Sigma)$. In this case, the rate distortion function of S is

$$R(\Sigma, D) = \sum_{n=1}^{N} \frac{1}{2} \log \frac{\lambda_n}{D_n} \tag{2}$$

where $\lambda_1 \geq \lambda_2 \cdots \geq \lambda_N$ are the ordered eigenvalues of the correlation matrix Σ

$$D_n = \min(K, \lambda_n) \tag{3}$$

and K is chosen such that

$$\sum_{n=1}^{N} D_n = D. \tag{4}$$

This is known as "reverse water filling" [11]. To formulate the optimization problem, we need to express the data correlations between the sensor nodes with a mathematical model. In sensor networks, sensor nodes generate data by detecting their surroundings; hence, it is a reasonable assumption that the data correlation between two nodes (node i and node j) can be expressed as a function of their spatial distance d_{ij}. Particularly, we let $\Sigma_{ij} = W^{d_{ij}^2}$, where W is a correlation parameter that represents the amount of correlation between spatial samples. W should be less than 1 such that Σ is a semi-positive definite matrix. Given any subset of nodes X and the distortion per node d, we can construct its correlation matrix Σ_X and calculate its rate distortion function $R(\Sigma_X, d \cdot |X|)$. We believe that $\Sigma_{ij} = W^{d_{ij}^2}$ is a practical model. Since W is always less than 1, the data correlation between two nodes exponentially decreases with increasing spatial distance. Moreover, our optimization framework can be applied with other data correlation models. Provided that the data correlation decreases with increasing spatial distance, the result of this paper should not be affected.

Depending on the application of the sensor network, its data source can be either continuous or discrete. Since the sources are continuous for most applications, we assume in this paper that the data generated by the sensor nodes can be represented by continuous random variables. For continuous random variables, their entropies are not significant as they always approach infinity. It is widely known that we can apply the rate distortion theory to find the minimum number of bits to represent continuous random variables given a distortion threshold. Recalling that entropy defines the minimum number of bits to represent a random variable, we can approximate the entropy of a continuous source with its rate distortion function $H(X) \approx R(\Sigma_X, d \cdot |X|)$.

Throughout this paper, the notation $H(X)$ is used to represent the entropy of the data generated by a set of sensor nodes X. Hence, while X represents a set of sensor nodes, it also stands for a random variable representing the data generated by X. This double representation of X applies whenever the notation $H(X)$ is used.

C. Optimization Objective

Given a rate allocation and a transmission structure, the flow rate on each wireless link, which is denoted by f_{ij}, can be determined, and the transmission energy consumed on each link is equal to $e_{ij} \cdot f_{ij}$. The objective of our optimization is to minimize the total transmission energy consumed in the network, i.e.,

$$\textbf{Minimize} \sum_{(i,j)\in E} e_{ij} \cdot f_{ij}. \tag{5}$$

In addition to transmission energy, the objective can be modified to optimize other metrics of interest with the structure [link cost] $\times$ [data size]. A similar optimization objective can be found in [3].

D. Flow Conservation Constraints

For each sensor node $i \in S_N$, the total outgoing traffic flows must equal to the sum of the incoming traffic flows and the nonnegative data transmission rate allocated to node R_i, i.e.,

$$\sum_{j:(i,j)\in E} f_{ij} - \sum_{j:(j,i)\in E} f_{ji} = R_i \qquad \forall i \in S_N. \tag{6}$$

These constraints enforce lossless transmission, which implies that no data packet will be discarded by any intermediate sensor node. In this paper, the sensor nodes utilize Slepian–Wolf coding to exploit data correlation. As a result, all the data packets generated by the sensor nodes contain independent data, and they must be received by the sink nodes.

E. Channel Contention Constraints

To generate solutions that are supported by the wireless shared medium, we introduce channel contention constraints in our formulation. The purpose of these constraints is to model the location-dependent contention that exists among the competing data flows. To accomplish this task, we need to identify when a transmission is successfully received by its intended recipient. In the literature, there exists two models for packet transmission in wireless networks [4]. They are generally referred to as the *protocol model* and the *physical model*, and they are presented as follows.

- *Protocol Model*: This model determines if a packet transmission is successful by considering the spatial location of the nodes. A packet transmission from node i to j is successful if, for all node k with $d_{kj} < (1+\triangle)d_{ij}$, node k is not transmitting. The quantity $\triangle > 0$ specifies a guard zone. In this paper, the interference range is assumed to be identical to the transmission range. Thus, $\triangle = 0$.
- *Physical Model*: This model is related to the physical layer and considers the signal power received at the receiver node. A packet transmission from node i to node j is successful if the signal-to-interference ratio (SIR) is greater than a threshold $\mathrm{SIR}_{ij} \geq \mathrm{SIR}_{\mathrm{thresh}}$.

In this paper, we focus on the protocol model of packet transmission. Based on the protocol model, any link originating from node k will interfere with link (i,j) if $d_{kj} < (1+\triangle)d_{ij}$. Utilizing this model, we derive Ψ_{ij} for each link $(i,j) \in E$ as the *cluster* of links that cannot transmit as long as link (i,j) is active. Here, the notation of cluster is treated as a basic resource unit as compared to individual links in traditional wireline networks. In wireline networks, data flows compete for the capacity of individual links. However, in the case of sensor networks, the capacity of a wireless link is interrelated with other wireless links in its vicinity. Consequently, data flows compete for the capacity of individual clusters, which is equivalent to the capacity of the wireless shared medium. A flow vector $[f_{ij}]_{\forall(i,j)\in E}$ is supported by the wireless shared medium if the following channel contention constraints hold [12]:

$$f_{ij} + \sum_{(p,q)\in\Psi_{ij}} f_{pq} \leq C \qquad \forall (i,j) \in E \tag{7}$$

where C is defined as the maximum rate supported by the wireless shared medium.

Note that in the equation above, f_{ij} and any instances of f_{pq} are not necessarily same-time flows. To illustrate this, let link (i,j) and link (p,q) be two interfering links. If f_{ij} and f_{pq} are both 10 kb/s, and the capacity of the shared medium is 20 kb/s, then the shared medium can support both f_{ij} and f_{pq} by transmitting the data in different time frames. Within a second, the shared medium can transmit at 20 kb/s on link (i,j) for 0.5 s and at 20 kb/s on link (p,q) for the other 0.5 s. In summary, (7) states that the combined flow rates of the interfering links cannot exceed the capacity of the shared medium, and it does not imply that the interfering links are generating same-time flows.

In practice, there are various methods that can be employed to construct the clusters [13]. For instance, if each node is provided with its own location information, in coordinates or in relation to the other nodes, then the clusters can be formed by considering the protocol model. An alternative is for a node to form local topology knowledge based on overheard transmissions in its surroundings. The exact method used to construct the clusters is beyond the scope of this paper.

In addition to the protocol model of packet transmission, the channel contention constraints can be simply tailored to adapt a particular Media Access Control (MAC) protocol by adjusting the derivation of the clusters Ψ_{ij}. For instance, in an IEEE 802.11 MAC protocol based network, if link (i, j) is active, then Ψ_{ij} should include all links that are originating from node k that satisfies $d_{kj} < (1 + \triangle)d_{ij}$ or $d_{ki} < (1 + \triangle)d_{ij}$. The sending node i is also required to be free of interference since it needs to receive the link layer acknowledgments from the receiving node j.

F. Rate Admissibility Constraints

Due to data correlation, the data collected by nearby sensor nodes are often redundant. Since transmitting redundant data across the network consumes unnecessary energy and decreases the useful throughput of the network, it is desirable to eliminate all redundancy. In the literature, there are many coding schemes that can be employed to exploit the data correlation. They can be generally divided into two categories, which are distributed source coding and joint entropy coding with explicit communication [3]. For coding with explicit communication, the sensor nodes aggregate their data with the side information received from other nodes. In this scenario, it is shown that the optimal rate allocation can be simply determined since it only relies on the side information, but building the optimal transmission structure becomes NP-hard. In contrast, distributed source coding allows each sensor node to generate independent data packets assuming that the sensors have knowledge of the global correlation structure. Although distributed source coding requires increased coding complexity and knowledge of the correlation structure, it is theoretically the most efficient coding scheme. It can achieve maximum energy savings for a lossless transmission since no redundant data are ever transmitted. Moreover, it can be implemented in distributed asynchronous network environments. Therefore, we employ distributed source coding to solve the correlated data-gathering problem.

We employ Slepian–Wolf coding as introduced in [14], which is a fundamental research study in distributed source coding. The Slepian–Wolf region specifies the minimum encoding rates that the sensor nodes must meet to transmit all independent data to the sink nodes. It is satisfied when every subset of sensor nodes encodes their collected data at a total rate exceeding their joint entropy. In mathematical terms, we have

$$\sum_{i \in \mathbf{Y}} R_i \geq H(\mathbf{Y}|\mathbf{Y}^C), \qquad \mathbf{Y} \subseteq S_N \tag{8}$$

where $\mathbf{Y}^C$ is the complement of $\mathbf{Y}$, $\mathbf{Y}^C = S_N - \mathbf{Y}$.

G. Exponential-Constraint Linear Programming Formulation

Combining the optimization objective with the introduced constraints, the correlated data-gathering problem can be modeled as an optimization problem, i.e.,

$$\textbf{Minimize} \quad \sum_{(i,j) \in E} e_{ij} \cdot f_{ij} \tag{9}$$

$$\textbf{Subject to}:$$

$$\sum_{j:(i,j) \in E} f_{ij} - \sum_{j:(j,i) \in E} f_{ji} = R_i \qquad \forall i \in S_N \tag{10}$$

$$f_{ij} + \sum_{(p,q) \in \Psi_{ij}} f_{pq} \leq C \qquad \forall (i,j) \in E \tag{11}$$

$$\sum_{i \in \mathbf{Y}} R_i \geq H(\mathbf{Y}|\mathbf{Y}^C), \qquad \mathbf{Y} \subseteq S_N \tag{12}$$

$$f_{ij} \geq 0 \qquad \forall (i,j) \in E \tag{13}$$

$$R_i \geq 0 \qquad \forall i \in S_N. \tag{14}$$

Since the rate admissibility constraints grow at an exponential rate in relation to the number of nodes, this is an exponential-constraint optimization formulation. In the following sections, we introduce a linear reformulation of this problem through localized Slepian–Wolf coding and further propose a price-based framework to provide a solution that is distributed among the individual sensor nodes.

III. Localized Slepian–Wolf Coding

The exponential-constraint linear optimization formulations are generally difficult to solve; hence, it is desirable to reduce the number of constraints from the formulation. Moreover, the rate admissibility constraints require each sensor node to have knowledge of the global correlation structure. This poses limitation on the scalability of our solution. In this paper, we adopt an approximated version of Slepian–Wolf coding from [15] to relax the rate admissibility constraints such that only the local correlation information is required at each sensor node. The approximation gives a definition for a neighborhood. For each sensor node, its neighborhood contains other sensors that are located in its surroundings. When a sensor node is determining its data transmission rate, it considers its data correlation with other sensors in its neighborhood instead of the entire network. Based on the spatial data correlation model, it is natural to assume that the sensors that are not in the neighborhood contribute very little or nothing in reducing the transmission rates. With a sufficient neighborhood size, this approximation should have a performance comparable to the global Slepian–Wolf coding. In this paper, we include the one-hop neighbors of the sensor nodes in their neighborhoods.

Extending from the approximation, we present a localized Slepian–Wolf coding scheme in Table I. This coding scheme supports sensor networks with multiple sinks, and it is amenable to distributed implementation. The localized Slepian–Wolf coding scheme specifies that each sensor node i should encode its data at a rate equal to the conditioned entropy. The conditioning is performed only on N_i, which is a subset of sensors within the neighborhood of sensor i that are closer

TABLE I
LOCALIZED SLEPIAN–WOLF CODING

1. Define a neighbourhood for each sensor node.
2. Find the nearest sink node for each sensor node using a distributed shortest path algorithm, such as the Bellman-Ford algorithm [16], [17]. Each sensor node refers to its nearest sink node as its destination sink node.
3. For each sensor node i:
1) Find within its neighbourhood, the set N_i of sensor nodes that have the same destination sink node as node i, and are closer to that destination sink node than node i.
2) The Slepian-Wolf region is satisfied when node i transmits at rate $R_i = H(i\|N_i)$.

to sensor i's destination sink node than sensor i itself. Instead of the global correlation structure, the sensor nodes using this coding scheme are only required to have knowledge of the correlation structure within their neighborhood. As a result, the localized Slepian–Wolf coding scheme overcomes the scalability limitation imposed by global Slepian–Wolf coding.

The performance of the localized Slepian–Wolf coding scheme depends on the transmission structure. The sensor nodes must realize their destination sink nodes before they can determine their achievable data transmission rates. If the capacity of the network is unconstrained, then the sensor nodes can simply determine their destination sink nodes based on relative spatial information. The sink node that is located closest to the sensor will be chosen as its destination sink node. However, when capacity constraints exist, a sensor node may not be able to transmit its collected data to its closest sink node due to data congestion. To avoid data congestion, our solution allows the sensor nodes to switch their destination sink nodes during run time. Hence, the transmission structure is dynamic as it is adjusted according to the rate allocation and the data congestion experienced by the wireless links. On the other hand, to accommodate the dynamic transmission structure, the localized Slepian–Wolf coding scheme dynamically determines the appropriate rate allocation during run time. Consequently, our solution jointly optimizes the rate allocation and the transmission structure, which are dependent upon each other. We believe that this approach will provide substantial improvements over the traditional approaches in solving the correlated data-gathering problem.

It is now possible to model the correlated data-gathering problem as a linear programming formulation. The rate admissibility constraints are relaxed, but the Slepian–Wolf region is still satisfied. The sensor nodes are required to transmit at the conditioned entropy specified by the localized Slepian–Wolf coding scheme. The linear programming formulation, which is also denoted as the *primal problem*, is expressed as follows:

$$\textbf{Minimize} \sum_{(i,j)\in E} e_{ij} \cdot f_{ij} \tag{15}$$

Subject to :

$$\sum_{j:(i,j)\in E} f_{ij} - \sum_{j:(j,i)\in E} f_{ji} = H(i|N_i) \qquad \forall i \in S_N \tag{16}$$

$$f_{ij} + \sum_{(p,q)\in\Psi_{ij}} f_{pq} \le C \qquad \forall (i,j) \in E \tag{17}$$

$$f_{ij} \ge 0 \qquad \forall (i,j) \in E. \tag{18}$$

IV. Distributed Solution: A Price-Based Framework

Many algorithms have been proposed in the past literature to solve linear optimization formulations, such as simplex, ellipsoid, and interior point methods. These algorithms are efficient in the sense that they can solve a large instance of optimization formulations in a few seconds. However, they have the disadvantage of being inherently centralized, which implies that they are not applicable for distributed implementations. In this section, we present our distributed solution to the proposed linear optimization formulation. The formulation is relaxed with Lagrangian dualization and then solved with the subgradient algorithm. In addition, we discuss the asynchronous network model that is utilized in this paper.

A. Lagrangian Dualization

With the localized Slepian–Wolf coding scheme, we are able to determine the optimal rate allocation. Our next step toward solving the linear programming formulation is to obtain the optimal transmission structure given the rate allocation. This part of the problem resembles a resource allocation problem, where the goal is to allocate the limited capacity of the wireless shared medium to the data flows originating from sensor nodes.

In the literature, Kelly *et al.* [18] and Low and Lapsley [19] have shown that the price-based resource allocation strategy is an efficient means to arbitrate resource allocation in wireline networks. With price-based strategy, the prices are computed as signals to indicate the relation between the supplies and demands of a resource. In these works, each wireless link is treated as a basic resource unit. A shadow price is associated with each wireless link to reflect the relation between the traffic load of the link and its bandwidth capacity. Based on the notation of maximal cliques, Xue *et al.* [20] extend the price-based resource allocation framework to respect the unique characteristic of location-dependent contention in wireless *ad hoc* networks.

In this paper, the notation of clusters as defined in Section II is utilized as the basic resource unit. Each cluster is associated with a shadow price, and the signals compete for the capacity of the clusters. The transmission structure is determined in response to the price signals such that the aggregated price paid by the data flows is minimized. It is shown from previous works that at equilibrium, such a price-based resource allocation strategy can achieve a global optimum, which leads to the optimal utilization of the resource. To solve the linear programming formulation with a price-based strategy, we first relax the channel contention constraints (7) with Lagrangian dualization and obtain the Lagrangian dual problem as

$$\textbf{Maximize } \mathrm{LS}(\beta), \quad \textbf{Subject to: } \quad \beta \geq 0. \tag{19}$$

By associating price signals or Lagrangian multipliers β_{ij} with the channel contention constraints, the Lagrangian dual problem is evaluated via the Lagrangian subproblem $\mathrm{LS}(\beta)$ as

Minimize

$$\sum_{(i,j)\in E} e_{ij} \cdot f_{ij} + \beta_{ij} \cdot \left(f_{ij} + \sum_{(p,q)\in\Psi_{ij}} f_{pq} - C \right) \tag{20}$$

Subject to :

$$\sum_{j:(i,j)\in E} f_{ij} - \sum_{j:(j,i)\in E} f_{ji} = H(i|N_i) \qquad \forall i \in S_N \tag{21}$$

$$f_{ij} \geq 0 \qquad \forall (i,j) \in E. \tag{22}$$

Here, we introduce a new notation Φ_{ij} as the set of clusters that link (i,j) belongs to. Recall that Ψ_{pq} refers to the cluster of links that cannot transmit when link (p,q) is active. For any link (i,j) that interferes with link (p,q), link (i,j) belongs to the cluster of link (p,q). Thus, for any link (i,j) and (p,q), $(p,q) \in \Phi_{ij}$ if and only if $(i,j) \in \Psi_{pq}$. Then the Lagrangian subproblem can be remodeled using this notation as

Minimize

$$\sum_{(i,j)\in E} f_{ij} \left(e_{ij} + \beta_{ij} + \sum_{(p,q)\in\Phi_{ij}} \beta_{pq} \right) - \beta_{ij} C \tag{23}$$

Subject to:

$$\sum_{j:(i,j)\in E} f_{ij} - \sum_{j:(j,i)\in E} f_{ji} = H(i|N_i) \qquad \forall i \in S_N \tag{24}$$

$$f_{ij} \geq 0 \qquad \forall (i,j) \in E. \tag{25}$$

The objective function of the remodeled Lagrangian subproblem specifies that the weight of each link is equal to the sum of its energy and capacity cost. Moreover, the capacity cost is equal to its Lagrangian multiplier of the link plus the sum of the Lagrangian multipliers in Φ_{ij}. This is intuitive since when link (i,j) is active, any link in the set Φ_{ij} cannot transmit due to interference. Thus, the actual cost for accessing link (i,j) should equal to the total cost for accessing link (i,j) and all the links in Φ_{ij}.

Since the capacity constraints are relaxed, we observe that the solution of the remodeled Lagrangian subproblem requires each sensor node to transmit its data along the shortest path that leads to its nearest sink node. As a result, the Lagrangian subproblem can be solved with any distributed shortest path algorithm, such as the well-known Bellman–Ford approach. Recall from the localized Slepian–Wolf coding scheme that a sensor node will coencode with another sensor node only if they have identical nearest sink node. Consequently, for any solution generated by the Lagrangian subproblem, data flows due to sensor nodes that have coencoded with each other will be absorbed by an identical sink node.

B. Subgradient Algorithm

We now describe the subgradient algorithm, which is an efficient iterative algorithm to solve the Lagrangian dual problem. The algorithm starts with a set of initial nonnegative Lagrangian multipliers $\beta_{ij}[0]$. Since the Lagrangian multipliers are price signals that reflect the congestion status of the clusters, a possible choice for the initial Lagrangian multipliers can be zeroes, assuming there is no data congestion in the network. In this case, the initial shortest paths chosen by the sensor nodes will be the minimum energy paths without any adjustments on the link weights.

During each iteration k, given current Lagrangian multiplier values $\beta_{ij}[k]$, we solve the Lagrangian subproblem by finding the shortest path from each sensor node to its nearest sink node, where the weight of each link is equal to the sum of its energy cost, its Lagrangian multiplier, and the Lagrangian multipliers of the clusters to which this link belongs. Using the new primal values $f_{ij}[k]$ obtained from the Lagrangian subproblem, we update the Lagrangian multipliers by

$$\beta_{ij}[k+1] = \max \left[0, \beta_{ij}[k] + \theta[k] \left(f_{ij}[k] + \sum_{(p,q)\in\Psi_{ij}} f_{pq}[k] - C \right) \right] \tag{26}$$

where θ is a prescribed sequence of step sizes. The equation above states that the Lagrangian multipliers vary depending on the value of $(f_{ij} + \sum_{(p,q)\in\Psi_{ij}} f_{pq} - C)$, which represents the amount of capacity violation within a cluster. When the violation of a cluster is positive, there are data flows traveling in the cluster that are not supported by the wireless shared medium. The Lagrangian multiplier for the cluster then increases according to the amount of violation to reflect the congestion. Conversely, when the violation for a cluster is negative, there is free bandwidth in the cluster that is not utilized by the data flows. Therefore, the Lagrangian multiplier for the cluster decreases to attract more data flows to occupy the free bandwidth.

The selection of step sizes plays an important role in the subgradient algorithm. If the step sizes are too small, then the algorithm has a slow convergence speed. If the step sizes are too large, then β_{ij} may oscillate around the optimal solution and fail to converge. The convergence is guaranteed when θ satisfied the following conditions [21], regardless of the values of the initial Lagrangian multipliers:

$$\theta[k] \geq 0, \qquad \lim_{k\to\infty} \theta[k] = 0, \quad \text{and} \quad \sum_{k=1}^{\infty} \theta[k] = \infty. \tag{27}$$

TABLE II
OPTIMIZATION PHASE

1) Choose initial Lagrangian multiplier values $\beta_{ij}[0]$, $\forall(i,j) \in E$.
2) For each cluster and link (i,j), repeat the following iteration until convergence (at times $t = 1, 2, \ldots$):
Cluster Price Update by cluster (i,j):
1) Receive flow rates $f_{pq}[t]$ from all links (p,q) in Ψ_{ij}.
2) Update Lagrangian multiplier $\beta_{ij}[t+1] = \max(0, \beta_{ij}[t] + \theta[t](f_{ij}[t] + \sum_{(p,q)\in\Psi_{ij}} f_{pq}[t] - C))$, where $\theta[t] = \frac{a}{(b+ct)}$.
3) Send $\beta_{ij}[t+1]$ to all links in Ψ_{ij}.
Link Rate Update by link (i,j):
1) Receive cluster prices $\beta_{pq}[t]$ from all clusters (p,q) in Φ_{ij}.
2) Determine the weight of link (i,j) as $(e_{ij} + \beta_{ij}[t] + \sum_{(p,q)\in\Phi_{ij}} \beta_{pq}[t])$.
3) If node i is a sensor node:
a) Compute its nearest sink node using a distributed shortest path algorithm, such as the Bellman-Ford algorithm. Sensor i refers to its nearest sink node as its destination sink node.
b) If modified, sensor i notifies other sensors in its neighbourhood about the identity and its distance to its new destination sink node.
c) Based on the information received, sensor i finds within its neighbourhood the set N_i. N_i consists of sensor nodes that have the same destination sink node as sensor i, and are closer to that destination sink node than sensor i.
d) Sensor i encodes its collected data at rate $R_i = H(i\|N_i)$, and transmits the encoded data to its destination sink node via the shortest path.
4) Measure and record the current rate on the link, denoted as $f_{ij}[t+1]$.
5) Send $f_{ij}[t+1]$ to all clusters in Φ_{ij}.

C. Distributed Algorithm

Based on the localized Slepian–Wolf coding scheme and the subgradient algorithm, we construct our distributed algorithm to solve the correlated data-gathering problem. Each cluster and wireless link is treated as an entity capable of processing, storing, and communicating information. In practice, each cluster and wireless link (i,j) is delegated to its sender node i, and all computations related to (i,j) will be executed on node i. The algorithm is summarized in Table II. In this algorithm, the price signals or the Lagrangian multipliers reflect the congestion status of the network. In addition, they act as a link of communication between the two problem objectives. When the algorithm converges, the generated solution will jointly optimize the rate allocation and the transmission structure.

Referring to Table II, the optimization algorithm requires the following control packets to be transmitted in each iteration.

1) Each cluster needs to have knowledge of the flow rates for all links within the cluster.
2) Each link needs to have knowledge of the prices for all clusters that are inherent to it.
3) Each sensor node needs to know, for the other sensor nodes within its neighborhood, the identities and their distances to their corresponding destination sink nodes.

Since the control packets introduced above are light weighted (with either rate/price/identity/distance information), and they are only communicated between local neighborhoods, the overhead introduced by these control packets should not be significant.

We now give an illustrative example to demonstrate the convergence of the distributed algorithm. Fig. 1 illustrates a random sensor network with 90 sensors and ten sinks, which are represented by asterisks and circles, respectively. All the nodes have a transmission range of 30 m, and the wireless links are represented by dotted lines. The distributed algorithm is executed for 500 iterations. The solid lines represent the links chosen by the obtained transmission structure. The thickness of each solid line indicates the amount of aggregated data traveling on the link, while the sensor nodes transmit according to the obtained rate allocation. Evidently, the distributed algorithm minimizes the energy consumption by exploiting data correlation. The sensor nodes that are distant from their corresponding sinks are assigned with lower transmission rates. For the duration of the experiment, the distributed algorithm generates a sequence of solutions. The total energy consumed by these solutions is recorded in Fig. 2. We observe from the figure that after an initial spike, the algorithm rapidly converges toward the optimal value within the first 50 iterations.

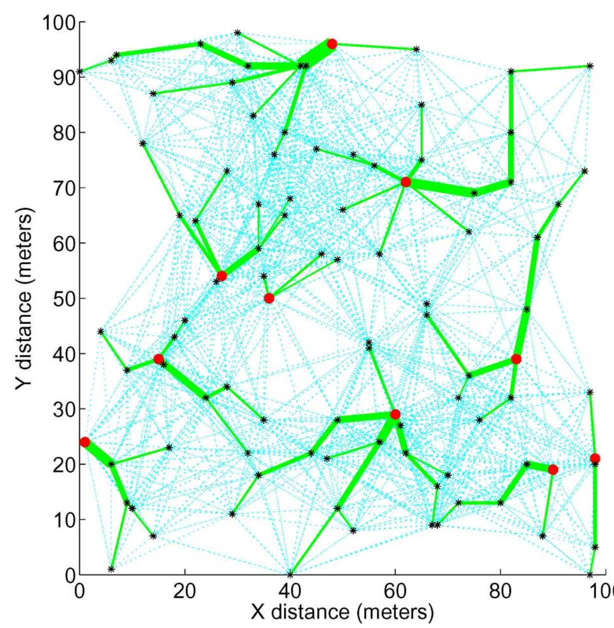


Fig. 1. Random topology with 100 nodes.

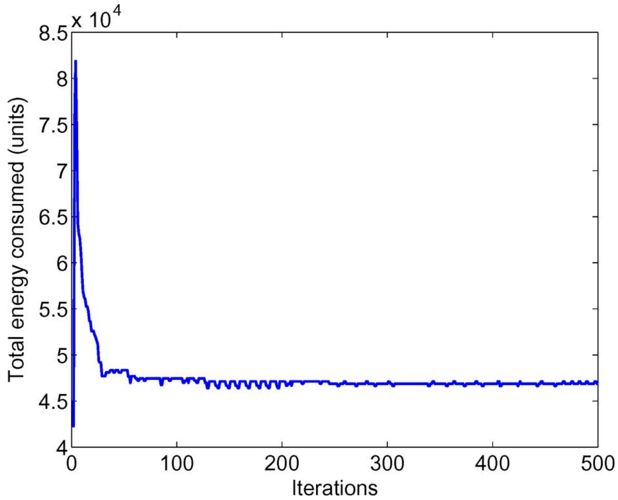


Fig. 2. Convergence behavior of the distributed algorithm.

D. Asynchronous Network Model

Until now, we have assumed a synchronous implementation for the iterative subgradient algorithm. In this case, the local clocks on the nodes are synchronized such that all of the nodes will simultaneously execute an iteration of the algorithm at every time instance $(t = 1, 2, 3, \ldots)$. A bounded communication delay is assumed where price and rate updates will arrive at their destinations before the next time instance. As a result, each node is able to execute the algorithm based on the most recent price and rate information. However, in realistic *ad hoc* network environments, it is expensive in terms of communication overhead to synchronize local clocks across the entire network.

In asynchronous network environments, nodes with different computation speeds will execute the iterative algorithm at varying paces. Consequently, the nodes may not always have the most recent price or rate information due to delayed or out-of-order updates. To accommodate these asynchronous updates, we introduce the *partial asynchronism model* that will be assumed in the practical implementation of our algorithm. The partial asynchronism model makes the following assumption.

There exists a positive integer B such that, for every cluster and wireless link (i, j), the time between consecutive updates is bounded by B for both price and rate updates, and the one-way communication delays between any two nodes are at most B time instances.

The partial asynchronism model is first discussed in [17]. Later, it is adopted by Low and Lapsley [19] in wireline networks and Xue *et al.* [20] in wireless networks. In [20], a technique is proposed to improve the price-based resource allocation strategy to accommodate the partial asynchronous model. At time instance t, instead of the most recent information, a node may only received a sequence of recent updates. The concept of this technique is for the nodes to estimate the most recent price and rate information by computing the average of the sequence received from time $t - B$ to t. To improve the accuracy of the estimation, a moving average can be utilized with a heavier weight assigned to the more recent updates. From their work, it is shown that such a strategy will converge the fastest when all the weight is assigned to the most recently received update. Moreover, with a sufficiently small step size θ, the strategy will converge to the global optimum in asynchronous network environments. Since our optimization formulation is solved with a price-based strategy, it is natural for us to adapt this technique. In our implementation, each node estimates the price and rate information based on the most recently received update. In Section VI, we validate via simulations that our algorithm converges in asynchronous network environments.

V. Implementation Issues

The subgradient algorithm provides an efficient tool in obtaining a lower bound on linear programming formulations (such as our primal problem) via solutions to the Lagrangian dual problem. However, it has the disadvantage that an optimal solution or even a feasible solution to the primal problem may not be found. With such a subgradient optimization approach, methods such as primal penalty functions and tangential approximation schemes have been proposed for directly obtaining the primal solutions. These methods are not suitable for our purpose because they either require the optimization to be conducted in the joint primal-dual space or introduce a significant additional computation overhead. In this section, we propose two implementations of our distributed algorithm that are aiming to overcome this problem. Moreover, we discuss how the distributed algorithm can be extended to handle network dynamics.

A. Implementation I: Primal Recovery

In [22], Sherali and Choi introduce a primal recovery algorithm. The algorithm directly recovers the primal solutions from the solutions to the Lagrangian dual problem generated by the subgradient algorithm. We adapt this algorithm in our first implementation. The primal recovery algorithm restricts the step size strategy and specifies that the primal solutions should equal to the convex combination of the solutions generated by the Lagrangian subproblem. At iteration k of the subgradient algorithm, we compose a primal solution $f^*_{ij}[k]$ via

$$f^*_{ij}[k] = \sum_{m=1}^{k} \lambda^k_m f_{ij}[m] \tag{28}$$

where $\theta[k]$'s are the step sizes, and λ^k_m's are the convex combination weights given by

$$\theta[k] = \frac{a}{b + ck} \qquad \forall k, \quad \lambda^k_m = \frac{1}{k} \qquad \forall m = 1, \ldots, k \qquad \forall k \tag{29}$$

where a, b, and c are positive constants. This step size strategy also satisfies condition (27); hence, the convergence of the subgradient algorithm is guaranteed. In the kth iteration, we can calculate the adjusted flow vector $f^*_{ij}[k]$ by

$$f^*_{ij}[k] = \frac{k-1}{k} f^*_{ij}[k-1] + \frac{1}{k} f_{ij}[k]. \tag{30}$$

TABLE III
STATISTICS ON CAPACITY VIOLATION IN BITS

Maximum	Minimum	Mean	Standard deviation
21.78944	0	8.23610	4.87849

It is proven in [22] that when the conditions in (29) are met, any accumulation point of the sequence $f^*_{ij}[k]$ generated via (28) is feasible to the primal problem.

Although the primal recovery algorithm guarantees to generate feasible primal solutions, it has a major disadvantage. Since the generated primal solution depends on the previous solutions, the network must remain static. Any dynamics introduced to the network such as sink mobility and duty schedules may introduce obsolete links during the execution of the distributed algorithm. If any one of the previous solutions contains the obsolete links, then the generated primal solution becomes invalid. To accommodate these dynamics, we propose a heuristic approach in our second implementation.

B. Implementation II: Capacity Reservation

Recall that the subgradient algorithm provides quick lower bounds to linear programming formulations. In the context of this paper, as the subgradient algorithm converges, it generates a sequence of rate allocations and transmission structures as solutions to the Lagrangian dual problem. However, some of these solutions often violate the channel contention constraints, which are imposed by the primal problem but relaxed in the dual problem. To analyze this behavior, we conduct a simulation study on sensor networks with 90 sensors and ten sinks. The capacity of the wireless shared medium is set to 150 bits. The distributed algorithm is executed for 1000 iterations on 50 random topologies. For each random topology, we record the amount of capacity violation induced by the last solution generated by the algorithm, and the statistics is presented in Table III.

Evidently, the subgradient algorithm generates tight lower bounds since the mean capacity violation is only a small fraction of the capacity offered by the shared medium. Based on this behavior, we observe that with high probability, the distributed algorithm can generate primal feasible solutions by reserving a suitable amount of capacity in advance. To reserve capacity, the distributed algorithm can be executed with the knowledge that the shared medium can only support a fraction (e.g., 90%) of its actual capacity. Although this implementation does not guarantee primal feasible solutions, it does not introduce any additional computational complexity into the algorithm.

C. Handling Network Dynamics

With energy-saving mechanisms such as sink mobility and duty schedules, the topology of a sensor network is inherently dynamic. Wireless links maybe added or removed from the topology. Moreover, since the energy cost of a wireless link is a function of its distance, it can vary with the node movement. One of the main design goals for the distributed algorithm is to be compatible with these mechanisms and variations to achieve higher energy savings. To this end, we propose an extension to the algorithm for handling network dynamics.

In the distributed algorithm, each wireless link is treated as two separate entities, i.e., a link and a cluster. For each wireless link (i, j), its sender node maintains two lists, i.e., Ψ_{ij} and Φ_{ij}. The first list Ψ_{ij} contains the identities and rates of the links that belong to cluster (i, j). In addition, the second list Φ_{ij} contains the identities and prices of the clusters to which link (i, j) belongs. To handle network dynamics, these lists must be updated as the topology changes.

We assume that the nodes are able to retrieve up-to-date topology information within their transmission range. At the beginning of each iteration, each participating node initiates the distributed algorithm by determining if it has new, obsolete, or modified links originating from itself. Afterwards, the nodes execute the maintenance phase given in Table IV. Finally, the nodes complete the iteration by executing the price and rate updates given in Table II. The purpose of the maintenance phase is to update the lists for the wireless links. It introduces several light-weighted control packets, and they are exchanged between the nodes and their local neighborhoods.

VI. PERFORMANCE EVALUATION

A. Simulation Environments

With the C++ programming language, we have implemented the proposed distributed algorithm for solving the correlated data-gathering problem. Our implementation includes both the optimization phase and the maintenance phase presented in Tables II and IV. The data packets, control packets, and update packets are communicated between the participating nodes with a round-robin scheduling algorithm. In practice, we expect that the data packets can be scheduled with a weighted fair queueing algorithm [23]. As a result, the sensors can achieve guaranteed data transmission rates specified by the rate allocation.

In this section, we evaluate both implementations proposed in Section V with extensive experimental results. The experiments are conducted on a high-performance cluster with 50 dual-processor servers. Unless stated, the experiments are performed on the random topology with 100 nodes presented in Fig. 1. The transmission range and the interference range are set to 30 m. The capacity of the wireless shared medium is set to 150 bits. The correlation parameter W and the per node distortion d is set to 0.99 and 0.01, respectively.

We study the distributed algorithm in three different simulation environments. In the *independent* environment, we neglect the effect of data correlation by substituting the localized Slepian–Wolf coding scheme with an independent coding scheme. In the *synchronous* environment, the participating nodes simultaneously execute an iteration of the algorithm at every time instance. The *asynchronous* environment is based on the partial asynchronism model, which assumes the existence of an integer B that bounds the time between consecutive updates. To implement this environment, each sensor node maintains a timer with a random integer value between 0 and B. The timer decreases itself by 1 at every time instance. When the timer reaches 0, the sensor node executes an iteration of the algorithm

TABLE IV
MAINTENANCE PHASE

1) For each wireless link (i, j) delegated to node i:

• If wireless link (i, j) is new or modified, node i sends a NOTIFY packet to node j. The NOTIFY packet contains the current price for cluster (i, j).

• If wireless link (i, j) is obsolete, node i sends a DELLINK packet to all nodes p for the clusters $(p, q) \in \Phi_{ij}$. In addition, node i sends a DELCLUSTER packet to all nodes p for the links $(p, q) \in \Psi_{ij}$.

2) Upon receiving the following packets:

• **NOTIFY**: Node j sends a CHECK packet to all potential nodes k that can be interfering with wireless link (i, j). Since the interference range is assumed to be equivalent to the transmission range in this paper, the potential nodes are limited to the nodes that are within node j's transmission range. The CHECK packet contains the price for cluster (i, j) extracted from the NOTIFY packet.

• **CHECK**: Based on local information exchanged between the nodes, node k determines if its outgoing wireless links (k, l) are interfering with wireless link (i, j) according to the protocol model of packet transmission.

- If wireless link (k, l) is interfering with wireless link (i, j). Check if cluster $(i, j) \in \Phi_{kl}$. If NO, add the identity and the price of cluster (i, j) to Φ_{kl}. Then, node k sends a ADDLINK packet to node i with the identity and rate of link (k, l).

- If wireless link (k, l) is NOT interfering with wireless link (i, j). Check if cluster $(i, j) \in \Phi_{kl}$. If YES, remove cluster (i, j) from Φ_{kl}. Then, node k sends a DELLINK packet to node i with the identity of link (k, l).

• **ADDLINK**: Cluster (i, j) adds the identity and the rate of link (k, l) to Ψ_{ij}.

• **DELLINK**: If DELLINK packet is generated by node i, cluster (p, q) removes link (i, j) from Ψ_{pq}. If DELLINK packet is generated by node k, cluster (i, j) removes link (k, l) from Ψ_{ij}.

• **DELCLUSTER**: Link (p, q) removes cluster (i, j) from Φ_{pq}.

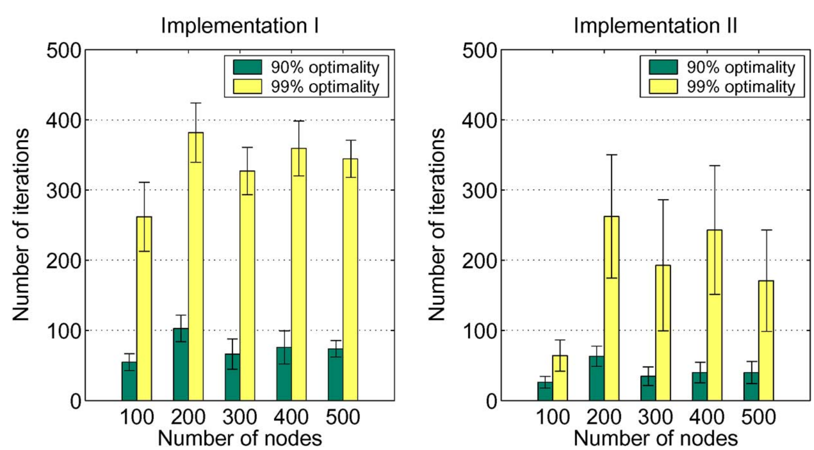


Fig. 3. Convergence speed in static networks. For each experiment, the horizontal bars indicate one standard deviation below and above the mean.

before resetting the timer. For experiments involving network dynamics, we make a conservative estimation that the sensor network is capable of executing two iterations of the algorithm per second. This implies that the duration of a time instance is equal to half of a second.

B. Convergence Speed

In our first study, we observe the convergence speed of the algorithm with different numbers of participating nodes. To this end, we generate five random sensor fields ranging from 100 to 500 nodes in increments of 100 nodes with 10% of the nodes randomly chosen as sink nodes. The sensor field with 100 nodes has an area of 100 m $\times$ 100 m. We maintain a constant node density by scaling the area of the other sensor fields. This eliminates the effect caused by varying node density and allows us to focus on the scalability of our algorithm. To attain convergence, we let the algorithm run for 500 iterations, and the optimum is taken as the minimum total energy consumption achieved. For each experiment, the algorithm is executed in the *synchronous* simulation environment on ten random topologies. For both implementations, we plot the mean number of iterations required to achieve 90% and 99% optimality in Fig. 3. The horizontal bars indicate one standard deviation below and above each mean. Supposing that the numbers approximately follow a Gaussian distribution given by the Central Limit Theorem, each interval includes about 70% of the observations.

The figure reveals that the convergence behaviors of the two implementations are different. On average, the primal recovery algorithm increases the convergence time by 50%, but the standard deviations on the convergence time are smaller when compared with the capacity reservation scheme. This is an expected result since the primal recovery algorithm generates a solution by averaging all the previous solutions obtained in the subgradient algorithm. In contrast, the capacity reservation scheme always utilizes the most current solution obtained in the subgradient algorithm. As a result, it has a shorter convergence time, but it is also heavily influenced by the fluctuations introduced by the subgradient algorithm, which leads to larger deviations. These convergence behaviors can be verified in Fig. 4, where we plot the sequences of solutions generated by the two implementations.

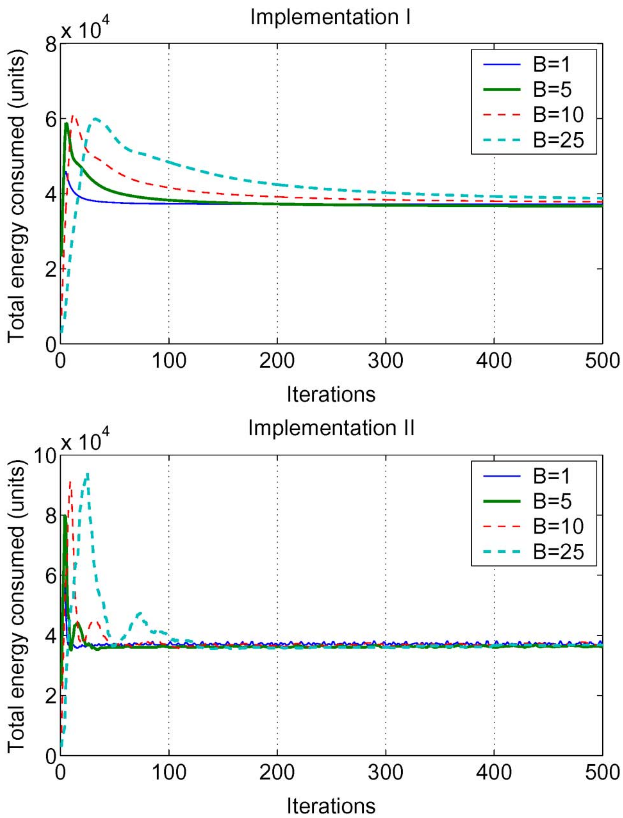


Fig. 4. Convergence behavior in asynchronous network settings.

In general, we observe that the time needed to achieve 99% optimality remains almost the same for networks with 200 to 500 nodes. Moreover, we notice that the algorithm can achieve 90% optimality in less than half of the time needed to achieve 99% optimality. Recall that for both implementations, the solution generated in each iteration is primal feasible. Therefore, when it is not necessary to achieve the true optimum, we can obtain a near-optimal solution in a much shorter time. These results exhibit the excellent scalability of our algorithm as the network size increases.

C. Impact of Asynchronous Network Settings

With the *asynchronous* simulation environment, we evaluate the convergence behavior of the distributed algorithm in asynchronous network settings. The algorithm is executed for 500 iterations with different time bounds $B = 1, 5, 10, 25$. For both implementations, the total energy consumption attained at each iteration is recorded in Fig. 4. In all the experiments, the algorithm converges toward an identical optimal solution. This result indicates that our algorithm is able to achieve convergence in asynchronous network settings. Moreover, we conclude that the convergence speed of the algorithm is associated with the time bound B since a longer convergence time is required when B is large.

D. Effect of Data Correlation

We investigate the effect of data correlation by comparing the *synchronous* simulation environment against the *independent*

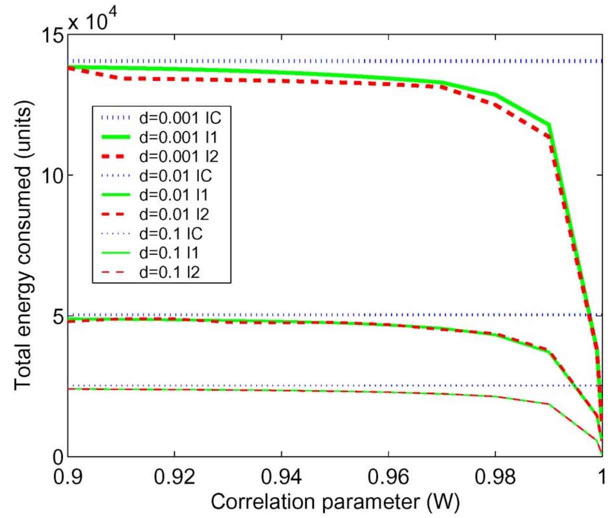


Fig. 5. Localized Slepian–Wolf coding versus independent coding. IC: independent coding, I1: implementation I, and I2: implementation II.

simulation environment. For each simulation environment, we execute the algorithm under three per node distortion values $d = 0.001$, 0.01, and 0.1. As the correlation parameter W varies from 0.9 to 0.9999, the minimum total energy consumption achieved by the different simulation environments is recorded in Fig. 5. Intuitively, the energy consumed at high correlation $(W = 0.9999)$ is much lower compared with the energy consumed at low correlation $(W = 0.9)$. Overall, the two implementations achieve similar results, and the localized Slepian–Wolf coding scheme outperforms the independent coding scheme in all the experiments. These results imply that although the proposed algorithm utilizes only local information, it can achieve significant energy savings for a wide range of data correlation and distortion levels.

E. Adaptation to Sink Mobility

In this section, we study the impact of sink mobility over random networks. Our aim is to seek the mobility threshold such that the algorithm is not fast enough to remain at convergence. Based on the random topology with 100 nodes, we introduce sink mobility by simultaneously moving the ten sink nodes. With periods of 50 and 100 s, the sink nodes move and remain static between alternating periods. For each mobile period, the sink nodes move in random directions with a specified average speed. The algorithm is executed in the *synchronous* simulation environment for 500 s. Fig. 6 plots the convergence behavior of the algorithm for the different scenarios. From this figure, we observe that the algorithm can achieve new convergence after the network topology is modified. The results indicate that the algorithm converges sufficiently well when the sink nodes move at 0.1 m/s without pause. When the node speed increases to 0.5 and 1 m/s, there are larger fluctuations in the attained total energy consumptions. A further increase in node speed may result in insufficient convergence time. In addition, we observe that the algorithm rapidly achieves and stays in convergence once the topology remains static. Obviously, the algorithm can support higher node speeds when the pause time increases.

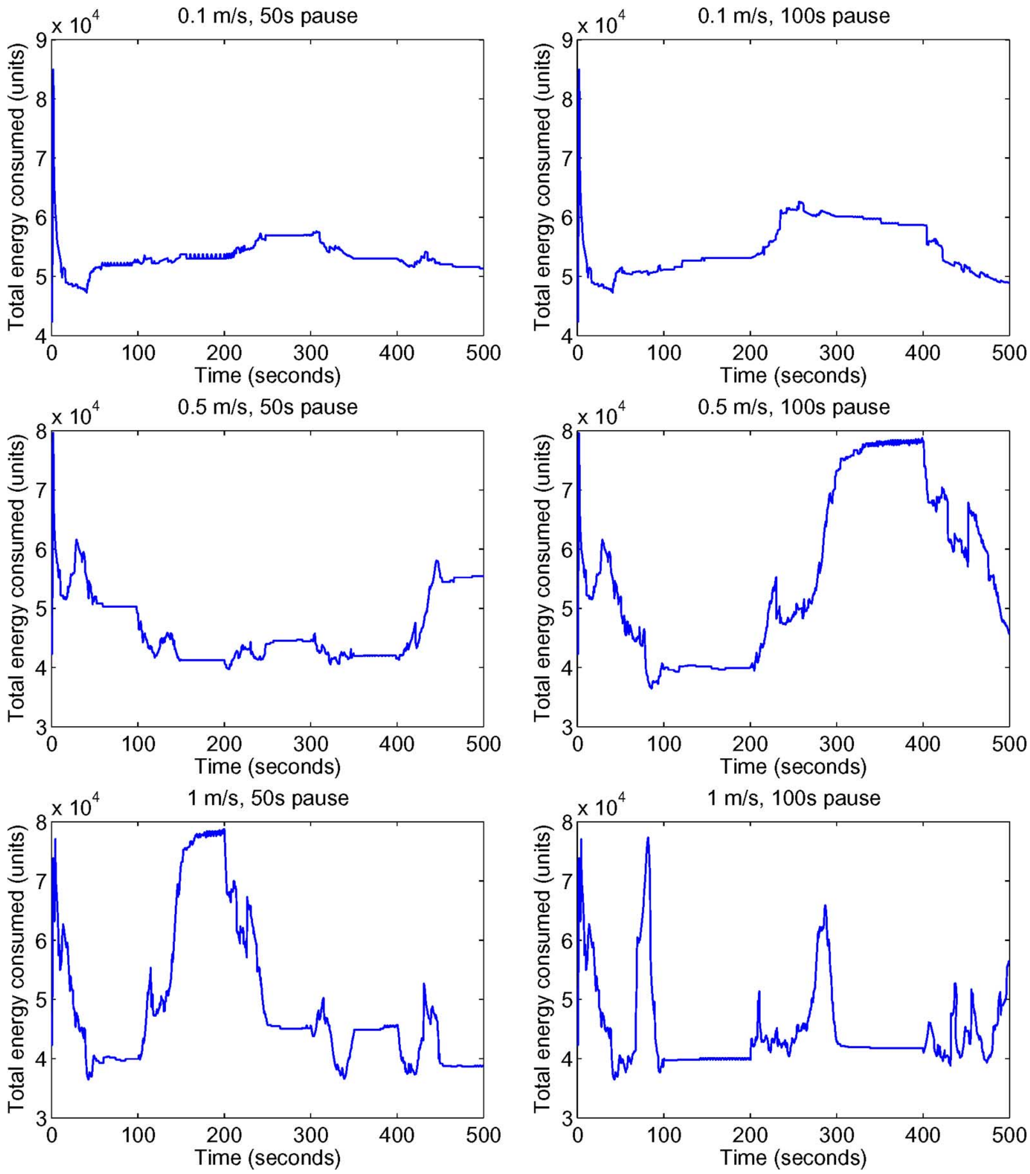


Fig. 6. Experiments with varying sink speeds and pause times.

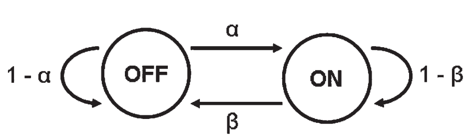


Fig. 7. Two-state Markov chain.

F. Adaptation to Duty Schedules

To extend the network lifetime, it is essential to establish the load balancing between sensor nodes with mechanisms such as duty schedules. In our final study, we are interested in examining the dynamic behavior of the distributed algorithm triggered by sensor joins and departures. We model the duty schedules as a two-state Markov chain, as shown in Fig. 7. The state transition probabilities α and β are adjusted to emulate different duty schedules. The experiments are performed in the *synchronous* simulation environment for 300 s. In the first 100 s, all the sensor nodes remain active. Afterwards, the sensors switch their operating status based on the introduced duty schedules.

The results of the experiments are summarized in Figs. 8 and 9. In Fig. 8, we adjust the summation of α and β with a fixed transition ratio α/β of 5. The summation represents the frequency of state transitions experienced by the network. Note that the summation cannot be greater than 2 since each of the transition probabilities cannot exceed 1. We observe that as the frequency of state transitions increases, the topology of the network changes more rapidly, which leads to larger fluctuations in the attained total energy consumptions. Fig. 9 illustrates the performance of the algorithm under different transition ratios with a fixed summation of 0.01. We have avoided combinations of transition ratio and summation that may lead to network partition. For example, if the transition ratio is less than 1, then the active sensor nodes are more likely to shut themselves off than inactive sensor nodes turning themselves on. As the number of inactive sensor nodes increases, a partition in the sensor network would eventually occur. Moreover, we notice

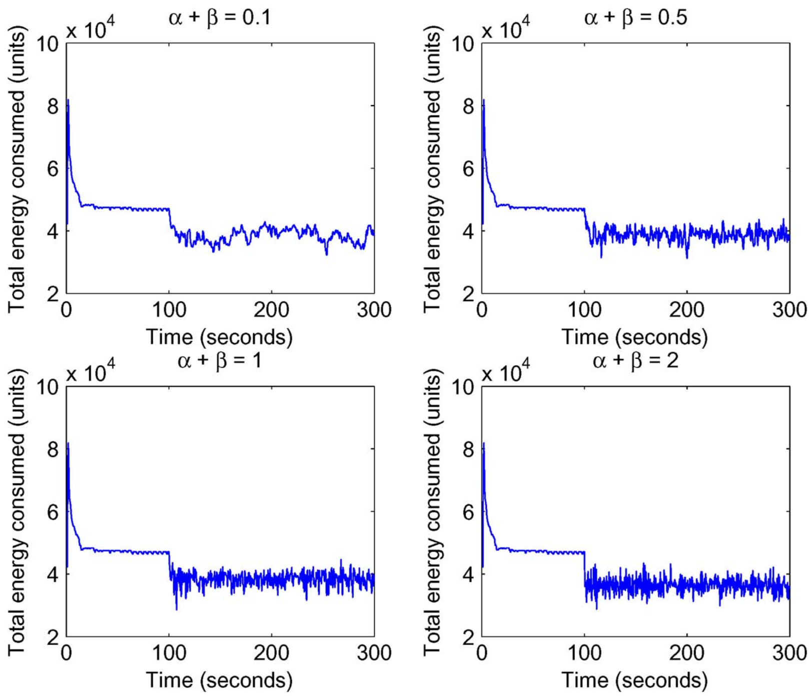


Fig. 8. Experiments with varying amount of state transitions.

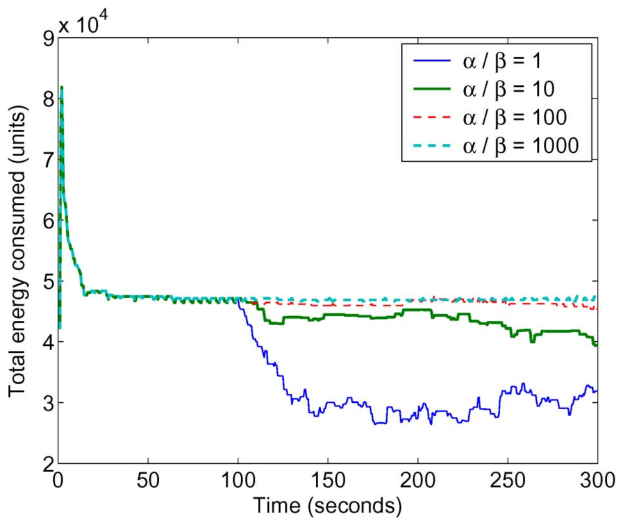


Fig. 9. Experiments with different transition ratios.

that with a higher transition ratio, the network consumes more energy since more sensor nodes are active.

VII. Related Work

The problem of energy-efficient routing in sensor networks has been investigated with mathematical optimization techniques in research studies including [24]–[27]. Chang and Tassiulas [24] have formulated a flow-based linear programming formulation to maximize the network lifetime. In [25], the optimization model minimizes the energy consumption and takes into account the channel contention constraints associated with the wireless shared medium. Ordonez and Krishnamachari [26] propose another optimization formulation to maximize the raw data arriving at the sink nodes subject to flow, fairness, energy, and capacity constraints. Johansson *et al.* [27] study the simultaneous routing and power allocation problem in wireless data networks using optimization techniques. In [26] and [27], the optimization problems utilize the physical model [4] of packet transmission in wireless networks to model the channel contention constraints. However, the resulting channel contention constraints are nonconvex, which can lead to extremely difficult optimization problems. In this paper, we represent the channel contention as linear constraints based on the protocol model [4]. More importantly, although all of the above existing works generally save energy, they do not consider the additional energy savings that can be achieved by exploiting the data correlation among the sensor nodes.

The data aggregation was introduced by Krishnamachari *et al.* [28] as an essential paradigm for wireless routing in sensor networks. The concept is to exploit the data correlation among the sensor nodes by eliminating redundancy. Consequently, there are fewer transmissions in the network, which thus save energy. In [7], Kalpakis *et al.* have formulated the maximum-lifetime data-gathering problem as a linear programming formulation by taking data aggregation into consideration and presented a polynomial-time algorithm to solve the problem. Although this optimization framework yields satisfactory performance, it makes the simplistic assumption of perfect data correlation, where intermediate sensor nodes can aggregate any number of incoming packets into a single packet. A perfect data correlation can also be found in [6], which analyzes the performance of data-centric routing schemes with in-network aggregation. In [8], Goel and Estrin consider the joint treatment of data aggregation and transmission structure. The problem of data gathering is addressed by using concave nondecreasing cost functions to model the aggregation function utilized by the intermediate nodes. However, it also makes the assumption of perfect data correlation. The aggregation performance of a node only depends on the number of nodes providing incoming data, regardless of the correlation structure. The assumption of perfect data correlation is not made in this paper since it is not applicable in most application scenarios.

While this paper exploits the data correlation with Slepian–Wolf coding, there are alternative approaches to take advantage of the correlation structure. In [3] and [29], the correlated data-gathering problem is considered with single-input coding schemes. With single-input coding, the data compression ratio at an intermediate node only depends on the side information provided by one other node. Cristescu *et al.* [3] prove that this optimization problem is NP-hard even in a simplified network setting, where the data compression ratio at the nodes does not depend on the quantity of side information but only on its availability. Since single-input coding schemes only consider data correlation among pairs of nodes, they will not perform as well as source coding schemes, which consider the joint data correlation of multiple nodes.

The multi-input coding schemes are often employed by routing schemes embedded with data aggregation, such as directed diffusion [30], LEACH [31], and PEGASIS [32]. Directed diffusion is a routing driven algorithm that emphasizes source compression at each individual node, and data aggregation opportunistically occurs when the routes intersect. In the model of LEACH, the nodes are chosen as cluster heads, which are then responsible for aggregating all the data generated in their corresponding cluster into a single packet. Instead of clusters, the PEGASIS algorithm finds chains of nodes, and the head node of each chain aggregates data from other nodes in the chain. Although the multi-input coding schemes can exploit data correlation among multiple nodes, they require the participating nodes to explicitly communicate with each other. In contrast, the Slepian–Wolf coding schemes do not require any explicit communication; hence, they can be applied in asynchronous network settings where no timing assumptions are made. In addition, these routing schemes do not incorporate the effect of wireless interference in their design.

Other closely related works are the ones involving Slepian–Wolf source coding. In [33], Barros and Servetto introduce the sensor reach-back problem, which requires one of the nodes in the network to receive enough information to reproduce the entire field of observation. The Slepian–Wolf coding is employed to meet the above requirement. This paper inspires us to apply Slepian–Wolf coding in the correlated data-gathering problem; hence, the sink nodes will be able to receive all independent data from the sensor nodes. In [15], Cristescu *et al.* address the correlated data-gathering problem with Slepian–Wolf coding. However, since their formulation does not consider the capacity and interference associated with the wireless channels, their solution may not be supported by the shared medium.

VIII. Conclusion

With the ability of distributed wireless sensing, the sensor networks can be applied to a vast number of applications. However, before we can recognize the full potential of sensor networks, the problem of correlated data gathering must be solved under realistic assumptions. We conclude this paper with the belief that our proposed framework is an efficient means to accomplish this task. In this paper, we have shown that in the presence of capacity constraints, finding the optimal rate allocation and finding the optimal transmission structure are two dependent problems. By jointly optimizing both problems, our approach minimizes the total transmission energy consumed by the network. Furthermore, it exploits data correlation among the sensor nodes and accounts for the effect of location-dependent contention in the wireless channels. To ensure scalability, our algorithm is amenable to distributed implementations, is applicable in asynchronous network settings, and provides support for multisink sensor networks. To the best of our knowledge, there does not exist any previous work that has simultaneously considered the correlated data-gathering problem with data aggregation and wireless channel interference, especially when a price-based strategy is employed to obtain a distributed algorithm to solve the problem.

References

[1] T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan, and K. K. Saluja, “Sensor deployment strategy for detection of targets traversing a region,” *ACM Mobile Netw. Appl.*, vol. 8, no. 4, pp. 453–461, Aug. 2003.

[2] A. Mainwaring, J. Polastre, R. Szewczyk, and D. Culler, “Wireless sensor networks for habitat monitoring,” in *Proc. 1st ACM Int. Workshop Wireless Sens. Netw. Appl.*, Sep. 2002, pp. 88–97.

[3] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, “On network correlated data gathering,” in *Proc. IEEE INFOCOM*, 2004, pp. 2571–2582.

[4] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[5] M. Lotfinezhad and B. Liang, “Effect of partially correlated data on clustering in wireless sensor networks,” in *Proc. IEEE Int. Conf. SECON*, Oct. 2004, pp. 172–181.

[6] B. Krishnamachari, D. Estrin, and S. Wicker, “Modelling data-centric routing in wireless sensor networks,” Univ. Southern California Comput. Eng., 2002, Tech. Rep. CENG 02-14.

[7] K. Kalpakis, K. Dasgupta, and P. Namjoshi, “Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks,” *Comput. Netw. J.*, vol. 42, no. 6, pp. 697–716, Aug. 2003.

[8] A. Goel and D. Estrin, “Simultaneous optimization for concave costs: Single sink aggregation or single source buy-at-bulk,” in *Proc. 14th SODA*, 2003, pp. 499–505.

[9] S. Tilak, N. Abu-Ghazaleh, and W. Heinzelman, “A taxonomy of wireless microsensor network models,” *ACM Mobile Comput. Commun. Rev.*, vol. 6, no. 2, pp. 28–36, Apr. 2002.

[10] T. Berger, *Rate Distortion Theory: A Mathematical Basis for Data Compression*, ser. Prentice-Hall Series in Information and System Science. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[11] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[12] B. Hamdaoui and P. Ramanathan, “Rate feasibility under medium access contention constraints,” in *Proc. IEEE GLOBECOM*, Dec. 2003, pp. 3020–3024.

[13] B. Liang and Z. J. Haas, “Virtual backbone generation and maintenance in ad hoc network mobility management,” in *Proc. IEEE INFOCOM*, Mar. 2000, pp. 1293–1302.

[14] D. Slepian and J. K. Wolf, “Noiseless coding of correlated information sources,” *IEEE Trans. Inf. Theory*, vol. IT-19, no. 4, pp. 471–480, Jul. 1973.

[15] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, “Networked Slepian–Wolf: Theory and algorithms,” in *Proc. Eur. Workshop Wireless Sens. Netw.*, 2004, pp. 44–59.

[16] D. P. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1992.

[17] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[18] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, “Rate control for communication networks: Shadow prices, proportional fairness and stability,” *J. Oper. Res. Soc.*, vol. 49, no. 3, pp. 237–252, Mar. 1998.

[19] S. H. Low and D. E. Lapsley, "Optimization flow control: Basic algorithm and convergence," *IEEE/ACM Trans. Netw.*, vol. 7, no. 6, pp. 861–874, Dec. 1999.

[20] Y. Xue, B. Li, and K. Nahrstedt, "Optimal resource allocation in wireless ad hoc networks: A price-based approach," *IEEE Trans. Mobile Comput.*, vol. 5, no. 4, pp. 347–364, Apr. 2006.

[21] N. Z. Shor, *Minimization Methods for Non-Differentiable Functions*. New York: Springer-Verlag, 1985.

[22] H. D. Sherali and G. Choi, "Recovery of primal solutions when using subgradient optimization methods to solve Lagrangian duals of linear programs," *Oper. Res. Lett.*, vol. 19, no. 3, pp. 105–113, Sep. 1996.

[23] H. Luo, S. Lu, and V. Bharghavan, "A new model for packet scheduling in multihop wireless networks," in *Procs. ACM MOBICOM*, 2000, pp. 76–86.

[24] J. H. Chang and L. Tassiulas, "Maximum lifetime routing in wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 12, no. 4, pp. 609–619, Aug. 2004.

[25] B. Hamdaoui and P. Ramanathan, "Energy-efficient and MAC-aware routing for data aggregation in sensor networks," in *Proc. Sens. Netw. Oper.*, 2004.

[26] F. Ordonez and B. Krishnamachari, "Optimal information extraction in energy-limited wireless sensor networks," *IEEE J. Sel. Areas Commun.—Special Issue Fundamental Performance Limits Wireless Sensor Networks*, vol. 22, no. 6, pp. 1121–1129, Aug. 2004.

[27] M. Johansson, L. Xiao, and S. Boyd, "Simultaneous routing and resource allocation in CDMA wireless data networks," in *Proc. IEEE Int. Conf. Commun.*, May 2003, vol. 1, pp. 51–55.

[28] B. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data correlation in wireless sensor networks," in *Proc. 22nd Int. Conf. Distrib. Comput. Syst.*, 2002, pp. 575–578.

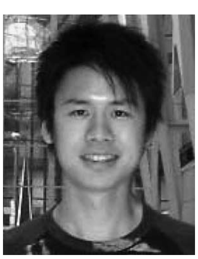
[29] P. V. Rickenbach and R. Wattenhofer, "Gathering correlated data in sensor networks," in *Proc. DIALM-POMC Joint Workshop Found. Mobile Comput.*, 2004, pp. 60–66.

[30] 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.

[31] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. Hawaii Int. Conf. Syst. Sci.*, Jan. 2000, pp. 3005–3014.

[32] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power efficient gathering in sensor information systems," in *Proc. IEEE Aerosp. Conf.*, Mar. 2002, pp. 1125–1130.

[33] J. Barros and S. D. Servetto, "Network information flow with correlated sources," *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 155–170, Jan. 2006.



Kevin Yuen (S'06) received the B.A.Sc. and M.A.Sc. degrees from the University of Toronto, Toronto, ON, Canada, in 2004 and 2006, respectively.

He is currently a Software Developer with IBM Toronto Laboratories, Markham, ON. He was interested in applying optimization techniques to achieve energy-efficient data gathering in sensor networks.

Ben Liang (S'94–M'01–SM'06) received the B.Sc. (Valedictorian) and M.Sc. degrees in electrical engineering from the Polytechnic University, Brooklyn, NY, in 1997 and the Ph.D. degree in electrical engineering with a computer science minor from Cornell University, Ithaca, NY, in 2001.

From 2001 to 2002, he was a Visiting Lecturer and a Postdoctoral Research Associate with Cornell University. Since 2002, he has been with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, where he is currently an Associate Professor. His current research interests are in mobile networking and multimedia systems.

Dr. Liang is a member of the Association for Computing Machinery and Tau Beta Pi. He received an Intel Foundation Graduate Fellowship in 2000 toward the completion of his Ph.D. dissertation, the Best Paper Award at the International Federation of Information Processing Networking Conference in 2005, and the Runner-up Best Paper Award at the International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks in 2006.

Baochun Li (M'00–SM'05) received the B.Engr. degree from Tsinghua University, Beijing, China, in 1995 and the M.S. and Ph.D. degrees from the University of Illinois at Urbana-Champaign, Urbana, in 1997 and 2000, respectively.

Since 2000, he has been with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, where he is currently an Associate Professor. He has been the Nortel Networks Junior Chair in network architecture and services since October 2003 and the Bell University Laboratories Chair in computer engineering since July 2005. His research interests include application-level quality-of-service provisioning and wireless and overlay networks.

Dr. Li is a member of the Association for Computing Machinery. He was the recipient of the IEEE Communications Society Leonard G. Abraham Award in the field of communications systems in 2000.